Doc Code: PA..
Document Description: Power of Attorney

PTO/AIA/82A (07-13)
Approved for use through 03/31/2021. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FOR POWER OF ATTORNEY TO ONE OR MORE REGISTERED PRACTITIONERS

NOTE: This form is to be submitted with the Power of Attorney by Applicant form (PTO/AIA/82B) to identify the application to which the Power of Attorney is directed, in accordance with 37 CFR 1.5, unless the application number and filing date are identified in the Power of Attorney by Applicant form. If neither form PTO/AIA/82A nor form PTO/AIA82B identifies the application to which the Power of Attorney is directed, the Power of Attorney will not be recognized in the application.

Application Number	
Filing Date	
First Named Inventor	Anurag Agarwal
Title	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
Art Unit	
Examiner Name	
Attorney Docket Number	ThreatM010

SIGNATURE of Applicant or Patent Practitioner

Signature	/Paul B. Johnson/	Date (Optional)	2019-06-29
Name	Paul. B. Johnson	Registration Number	59375
Title (if Applicant is a juristic entity)			
Applicant Name (if Applicant is a juristic entity)			

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. If more than one applicant, use multiple forms.

☑ *Total of 1 forms are submitted.

This collection of information is required by 37 CFR 1.131, 1.32, and 1.33. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 3 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Doc Code: PA..
Document Description: Power of Attorney

PTO/AIA/82B (07-13)
Approved for use through 03/31/2021. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number

POWER OF ATTORNEY BY APPLICANT

I hereby revoke all previous powers of attorney given in the application identified in either the attached transmittal letter or the boxes below.

Application Number	Filing Date

(Note: The boxes above may be left blank if information is provided on form PTO/AIA/82A.)

☑ I hereby appoint the Patent Practitioner(s) associated with the following Customer Number as my/our attorney(s) or agent(s), and to transact all business in the United States Patent and Trademark Office connected therewith for the application referenced in the attached transmittal letter (form PTO/AIA/82A) or identified above: 145042

OR

☐ I hereby appoint Practitioner(s) named in the attached list (form PTO/AIA/82C) as my/our attorney(s) or agent(s), and to transact all business in the United States Patent and Trademark Office connected therewith for the patent application referenced in the attached transmittal letter (form PTO/AIA/82A) or identified above. (Note: Complete form PTO/AIA/82C.)

Please recognize or change the correspondence address for the application identified in the attached transmittal letter or the boxes above to:

☑ The address associated with the above-mentioned Customer Number

OR

☐ The address associated with Customer Number:

OR

☐ Firm or Individual Name					
Address					
City		State		Zip	
Country					
Telephone		Email			

I am the Applicant (if the Applicant is a juristic entity, list the Applicant name in the box):

ThreatModeler Software Inc.

☐ Inventor or Joint Inventor (title not required below)

☐ Legal Representative of a Deceased or Legally Incapacitated Inventor (title not required below)

☑ Assignee or Person to Whom the Inventor is Under an Obligation to Assign (provide signer's title if applicant is a juristic entity)

☐ Person Who Otherwise Shows Sufficient Proprietary Interest (e.g., a petition under 37 CFR 1.46(b)(2) was granted in the application or is concurrently being filed with this document) (provide signer's title if applicant is a juristic entity)

SIGNATURE of Applicant for Patent

The undersigned (whose title is supplied below) is authorized to act on behalf of the applicant (e.g., where the applicant is a juristic entity).

Signature	Anurag	Date (Optional)	
Name	Anurag Agarwal		
Title	CEO of ThreatModeler Software Inc. of NJ		

NOTE: Signature - This form must be signed by the applicant in accordance with 37 CFR 1.33. See 37 CFR 1.4 for signature requirements and certifications. If more than one applicant, use multiple forms.

☑ Total of 1 forms are submitted.

This collection of information is required by 37 CFR 1.131, 1.32, and 1.33. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 3 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

SYSTEMS AND METHODS FOR AUTOMATED THREAT MODEL GENERATION FROM THIRD PARTY DIAGRAM FILES

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This document is a continuation-in-part application of U.S. Patent App. Ser. No. 16/228,738 titled "Threat Model Chaining and Attack Simulation Systems and Related Methods," naming as first inventor Anurag Agarwal, filed Dec. 20, 2018, which in turn is a continuation-in-part application of U.S. Patent App. Ser. No. 15/922,856 titled "Threat Model Chaining and Attack Simulation Systems and Methods," naming as first inventor Anurag Agarwal, filed Mar. 15, 2018, issued as U.S. Pat. No. 10,200,399 on Feb. 5, 2019, which in turn is a continuation-in-part application of U.S. Patent App. Ser. No. 15/888,021 titled "Threat Modeling Systems and Related Methods Including Compensating Controls," naming as first inventor Anurag Agarwal, filed Feb. 3, 2018, issued as U.S. Pat. No. 10,255,439 (hereinafter "the '021 Application") on Apr. 9, 2019, which '021 Application in turn claims the benefit of the filing date of U.S. Provisional Pat. App. Ser. No. 62/507,691 titled "System and Method of Including Compensating Controls in a Threat Modeling Process," naming as first inventor Anurag Agarwal, filed May 17, 2017, and which '021 Application also claims the benefit of the filing date of U.S. Provisional Pat. App. Ser. No. 62/527,671 titled "System and Method for Identifying and Analyzing the Potential Attack Surface of a Complex System," naming as first inventor Anurag Agarwal, filed June 30, 2017, and which '021 Application also claims the benefit of the filing date of U.S. Provisional Pat. App. Ser. No. 62/530,295 titled "Method and Apparatus for Early Implementation of Enterprise DevSecOps," naming as first inventor Anurag Agarwal, filed July 10, 2017, and which '021 Application also claims the benefit of the filing date of U.S. Provisional Pat. App. Ser. No. 62/520,954 titled "System and Method for Identifying Potential Threats to a Complex System," naming as first inventor Anurag Agarwal, filed June 16, 2017, the disclosures of each of which are hereby incorporated entirely herein by reference.

BACKGROUND

1. Technical Field

[0002] Aspects of this document relate generally to threat modeling processes and systems.

2. Background Art

[0003] Threat modeling is a process by which vulnerabilities of a system or process may be detailed and prioritized. One example of an existing threat modeler is a modeler marketed

IriusRisk
Ex. 1002 - Page 3

under the name THREAT MODELING TOOL by Microsoft Corporation of Redmond, WA. Threat modeling allows a user to analyze potential attack vectors and prioritize vulnerabilities. While some threat modeling involves threats related to computing networks and systems, threat modeling in general encompasses a broader scope and may involve modeling threats in non-computer-related systems and processes. Some commercial threat modeling tools utilize a dynamic question and answer user interface which generally relies on an underlying data flow diagram (DFD) or process flow diagram (PFD) conceptual basis.

[0004] With regards to modeling computing networks, traditional threat modeling systems and methods exist for assessing risk from potential cyber threats at an individual application level without fully accounting for application-application interactions, thereby miscalculating organizational risk. Other systems and methods exist for discovering vulnerabilities to applications deployed in cyber environments by surveying applications. These rely on recognizing previously identified and cataloged vulnerability signatures and are not designed to survey non-application items included in an organization's cyber environment, nor can these systems recognize which threats newly introduced to a cyber system are relevant to security, nor can these systems perform "what-if" scenarios as part of an organization's risk management analysis. Existing methodologies also do not provide for communicating the risks associated with identified threats in a way that non-security experts easily appreciate or understand.

[0005] Accordingly, traditional threat modeling methodologies have the following weaknesses: (1) they are effective in analyzing only single applications operating in isolation on a predetermined infrastructure; (2) they require security subject-matter experts for their creation, use, and maintenance; (3) they are resource-intensive to build and maintain, and; (4) they cannot be effectively used to scale a threat modeling practice to meet the needs of enterprises generating tens of software applications per year. Traditional threat modeling methodologies also cannot be easily integrated into existing agile software development approaches or with DevOps practices, and so are often rejected by agile developers and operational teams as non-productive.

[0006] Existing threat modeling methodologies also do not help organizations understand: (1) the nature of application interactions; (2) the user of shared components; (3) the effect of including third party elements within the IT ecosystem; (4) the downstream impact should potential threats be realized; (5) the nature of the organization's comprehensive attack surface; (6) the explicit nature of the organization's attacker population; (7) the effectiveness of deployed or contemplated compensating controls; or (8) a means to communicate the potential

IriusRisk
Ex. 1002 - Page 4

impact should threats be realized without communicating confidential information or security details which could further expose the organization to risk from potential threats.

[0007] Software exists for converting a diagram file of one type to another type of diagram file, though not into a threat model. For example, LUCIDCHART, SMARTDRAW, and EDRAWSOFT are programs which will convert a MICROSOFT VISIO file into another file type (for example importing the visual elements and retaining some diagram features of the original VISIO file) but without creating an associated threat model.

IriusRisk
Ex. 1002 - Page 5

[0008] Embodiments of threat modeling methods (methods) may include: providing one or more databases, the one or more databases including: a plurality of threat model components stored therein; and a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases; providing one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components"); and in response to receiving one or more user inputs, using one or more user interfaces displayed on one or more computing devices communicatively coupled with the one or more databases: using the one or more computing devices, reading a data file generated by the third party software application, the data file including a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components; determining using the one or more computing devices, for the subset of third party diagram components, correlated threat model components as defined in the one or more mapping files; displaying on the one or more user interfaces a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, the relational diagram defining a threat model; and generating, using the one or more computing devices, and displaying, on the one or more user interfaces, a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

[0009] Embodiments of threat model methods (methods) may include one or more or all of the following:

[0010] The one or more mapping files may not be stored in the one or more databases but may be stored on the one or more computing devices.

[0011] The one or more mapping files may include one or more tables stored within the one or more databases.

[0012] The data file may be a VSD file, a VSDX file, a TMT File, an XML file, a JSON file, a PNG file, and/or a JPEG file.

[0013] Determining, for the subset of third party diagram components, correlated threat model components, may include first determining whether a user-defined name for one of the subset of third party diagram components has a match in the one or more mapping files and, if no

IriusRisk
Ex. 1002 - Page 6

match is found, then determining whether a default title for the third party diagram component, as defined by the third party software application, has a match in the one or more mapping files.

[0014] The method may further include, if no match is found for the default title, matching the third party diagram component with a placeholder component and displaying the placeholder component on the relational diagram to indicate that no match was found.

[0015] The method may further include determining, for the one or more relationships defined by the data file, correlated threat model components as defined in the one or more mapping files.

[0016] Embodiments of threat modeling systems (systems) may include: one or more computing devices communicatively coupled with one or more databases, the one or more databases including: a plurality of threat model components stored therein; and a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases; one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components"); an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by one of the one or more computing devices, the data file generated by the third party software application, the data file including a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components; an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of third party diagram components in the one or more mapping files; a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, the relational diagram defining a threat model; and a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

[0017] Embodiments of threat modeling systems (systems) may include one or more or all of the following:

IriusRisk
Ex. 1002 - Page 7

[0018] The one or more mapping files may not be stored in the one or more databases but may be stored on the one or more computing devices.

[0019] The one or more mapping files may include one or more tables stored within the one or more databases.

[0020] The data file may be a VSD file, a VSDX file, a TMT File, an XML file, a JSON file, a PNG file, and/or a JPEG file.

[0021] The one or more mapping files may include a file correlating user-defined names of the third party diagram components with the threat model components.

[0022] The one or more mapping files may include a file correlating default titles of the third party diagram components with the threat model components.

[0023] For each of the subset of third party diagram components for which no correlated threat model component is found in the one or more mapping files, the diagram interface may display a placeholder component indicative of a match not being found.

[0024] One or more mapping interfaces may be displayed on the one or more computing devices and may be configured to, in response to receiving user input, modify the one or more mapping files.

[0025] Embodiments of threat modeling systems (systems) may include: one or more computing devices communicatively coupled with one or more databases, the one or more databases including: a plurality of threat model components stored therein; and a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases; one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with VISIO diagram components; an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by the one or more computing devices, the data file including a subset of the VISIO diagram components, the data file defining one or more relationships between the subset of VISIO diagram components; an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of VISIO diagram components in the one or more mapping files; a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of a computing network, using visual representations of the threat model components correlated with the subset of VISIO diagram components, the relational diagram defining a threat model; and a threat report interface, displayed on the one

IriusRisk
Ex. 1002 - Page 8

or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

[0026] Embodiments of threat modeling systems (systems) may include one or more or all of the following:

[0027] The one or more mapping files may not be stored in the one or more databases but may be stored on the one or more computing devices.

[0028] The one or more mapping files may include one or more tables stored within the one or more databases.

[0029] The one or more mapping files may include a file correlating user-defined names of the third party diagram components with the threat model components.

[0030] The one or more mapping files may include a file correlating default titles of the third party diagram components with the threat model components.

[0031] General details of the above-described embodiments, and other embodiments, are given below in the DESCRIPTION, the DRAWINGS, and the CLAIMS.

IriusRisk
Ex. 1002 - Page 9

BRIEF DESCRIPTION OF THE DRAWINGS

[0032] Embodiments will be discussed hereafter using reference to the included drawings, briefly described below, wherein like designations refer to like elements:

[0033] FIG. 1 is a diagram representatively illustrating an implementation of a threat modeling system (hereinafter at times called a "threat model chaining system," "attack simulation system," "system for automated threat model generation from third party diagram files," and "system");

[0034] FIG. 2 is a block diagram representatively illustrating an implementation of a threat modeling method;

[0035] FIG. 3 is a block diagram representatively illustrating an implementation of a threat modeling method;

[0036] FIG. 4 is a flowchart representatively illustrating an implementation of a threat modeling method;

[0037] FIG. 5 is a flowchart representatively illustrating an implementation of a threat modeling method;

[0038] FIG. 6 is an implementation of an interface of the system of FIG. 1;

[0039] FIG. 7 is an implementation of an interface of the system of FIG. 1;

[0040] FIG. 8 is an implementation of an interface of the system of FIG. 1;

[0041] FIG. 9 is an implementation of an interface of the system of FIG. 1;

[0042] FIG. 10 is an implementation of an interface of the system of FIG. 1;

[0043] FIG. 11 is an implementation of an interface of the system of FIG. 1;

[0044] FIG. 12 is an implementation of an interface of the system of FIG. 1;

[0045] FIG. 13 is an implementation of an interface of the system of FIG. 1;

[0046] FIG. 14 is a block diagram representatively illustrating an implementation of an attack simulation method;

[0047] FIG. 15 is an implementation of an interface of the system of FIG. 1;

[0048] FIG. 16 is a flowchart representatively illustrating an implementation of a threat modeling method;

[0049] FIG. 17 is an attack tree diagram representatively illustrating an attack path of a threat of a threat model;

[0050] FIG. 18 is a block diagram representatively illustrating attack paths of multiple threats of a threat model;

[0051] FIG. 19 is the block diagram of FIG. 18 with compensated controls added to the diagram;

IriusRisk
Ex. 1002 - Page 10

[0052] FIG. 20 is a block diagram of an asset classification method of an attack simulation method;

[0053] FIG. 21 is an implementation of an interface of the system of FIG. 1;

[0054] FIG. 22 is an implementation of an interface of the system of FIG. 1;

[0055] FIG. 23 is an implementation of an interface of the system of FIG. 1;

[0056] FIG. 24 is an implementation of an interface of the system of FIG. 1;

[0057] FIG. 25 is an implementation of an interface of the system of FIG. 1;

[0058] FIG. 26 is an implementation of an interface of the system of FIG. 1;

[0059] FIG. 27 is an implementation of an interface of the system of FIG. 1;

[0060] FIG. 28 is an implementation of an interface of the system of FIG. 1;

[0061] FIG. 29 is an implementation of an interface of the system of FIG. 1; and

[0062] FIG. 30 is a flowchart representatively illustrating a method of threat model generation using the system of FIG. 1.

IriusRisk
Ex. 1002 - Page 11

DESCRIPTION

[0063] Implementations/embodiments disclosed herein (including those not expressly discussed in detail) are not limited to the particular components or procedures described herein. Additional or alternative components, assembly procedures, and/or methods of use consistent with the intended systems and methods for automated threat model generation from third party diagram files (hereinafter called "threat model system(s)" and "threat model method(s)" respectively) may be utilized in any implementation. This may include any materials, components, sub-components, methods, sub-methods, steps, and so forth.

[0064] As used herein, the term "input field" includes a "selector." For example, a button or space on a user interface in which a user may move a cursor to and click to make a selection, and a checkbox field, and other similar fields, as well as alphanumeric input fields, are all "input fields" as used herein.

[0065] The term "compensating control" in implementations herein may be an alternative mechanism to a security requirement or standard issued by a standards-issuing body that is allowed by the standards-issuing body when the security requirement or standard as stated cannot be met by a party due to legitimate technical or documented business constraints.

[0066] In the payment card industry (PCI), as a non-limiting example, compensating controls were introduced in Payment Card Industry Data Security Standard (PCI DSS) 1.0 to give organizations an alternative to security requirements that could not be met due to legitimate technological or business constraints. According to the PCI Council, which was the standards-issuing body issuing the standard (jointly created by the four major credit-card companies VISA, MASTERCARD, DISCOVER, and AMERICAN EXPRESS), compensatory controls in that industry must: (1) meet the intent and rigor of the original stated requirement; (2) provide a similar level of defense as the original stated requirement; (3) be "above and beyond" other PCI DSS requirements (not simply in compliance with other PCI DSS requirements); and (4) be commensurate with the additional risk imposed by not adhering to the original stated requirement. Examples of compensating controls for information technology may include: using audit trails and logs for payroll management instead of segregation of duties (having two individuals responsible for separate parts of payroll management); the use of database security applications and services, network access control (NAC), data leak prevention strategies, and e-mail encryption in lieu of comprehensive encryption (i.e., in lieu of converting all electronic data into ciphertext and changing cryptographic keys periodically); two-factor authentication with a change of password every 60 days in lieu of long complex passwords; and so forth.

IriusRisk
Ex. 1002 - Page 12

[0067] Notwithstanding the above, in implementations herein "compensating control" may have a more general definition. For example, in implementations a "security requirement" may be defined as a mechanism for stopping or closing a threat at the source of the threat, and a "compensating control" may be defined as a mechanism for stopping or closing a threat not at the source, but between the source and some protected component (i.e., preventing a threat emanating from a source from reaching a protected component). In other implementations a "security requirement" may be defined as a hardware solution and a "compensating control" may be defined as a software solution, such as a change in the code or software added to a system. In still other implementations a "security requirement" may be generally defined as any solution which is costlier or more time consuming and a "compensating control" may be defined as a solution which is not as secure or sure-proof as the relevant security requirement but which is less cost-prohibitive or time-prohibitive.

[0068] The threat modeling system and related methods discussed herein are implemented using computing devices and/or networks. Referring to FIG. 1, an implementation of a threat modeling system (threat model chaining system) (attack simulation system) (system) 100 is shown. FIG. 1 only shows a representative example, and there are many other contemplated systems that could be used to implement the threat modeling processes. System 100 includes a computing device 102 having a display 104. While the computing device is drawn as a desktop computer it could be a laptop, a mobile phone or tablet, or any other type of computing device. The same goes for all other computing devices shown in the drawings.

[0069] Device 102 is shown communicatively coupled with server 106 which is communicatively coupled with a database (DB) 108. The coupling may be direct, such as through a wired connection, or through a local area network (LAN), or remotely through telecommunication network 110 (which may be the Internet). In some systems the server and database could be housed on the same machine as the computing device 102 using virtualization. In implementations device 102 could be accessed by an administrator of the system to choose settings, add or remove users, add or remove items from the database, and so forth. System 100 only shows one computing device 102, though in implementations the number of computing devices 102 may be scaled up to any number. Likewise, only one server and database are shown, but these also may be scaled up to any number as needed.

[0070] Other computing devices may be included in system 100. Computing device 112 includes display 114 and is an example of a computing device which is communicatively coupled with device 102 both directly (such as through a hardwired or wireless LAN), and coupled

IriusRisk
Ex. 1002 - Page 13

directly with the server (such as through a hardwired or wireless LAN), and also may be coupled with the server and/or the device 102 through telecommunication network 110. System 100 is shown with only one device 112 but in implementations it could be scaled up to any number of devices 112.

[0071] Computing device (device) 116 is an example of a computing device that is not directly coupled with either device 102 or the server but is only coupled thereto through the telecommunications network 110. Nevertheless, device 116 may access the server and database through the telecommunications network. Although only one device 116 is shown, this may be scaled up to any number. Device 116 has a display 118, as shown.

[0072] Also shown are a web server 120 and a remote server (server) 122, each of which may be included in implementations of system 100. By non-limiting example, device 116 may access the server 106 and database through the web server 120, such as by navigating to a uniform resource locator (URL) and providing login credentials. Computing devices 102 and 112 could do the same. Although only one web server is shown, this may be scaled up to any number as needed.

[0073] None of the computing devices shown in FIG. 1 are directly coupled with remote server 122, which may by non-limiting example be a third party server, or multiple servers (such as a portion of a server rack) or any portion thereof. System 100 could, for example, exclude server 106, and utilize only remote servers 122 which have access to the database 108 (which may be stored on the remote servers), and each of the computing devices may access the database through the remote servers and through one or more described web servers such as through one or more user interfaces displayed on the displays of the computing devices when accessing correlated URLs.

[0074] In other implementations one or more application servers could be included in the system, the application server(s) positioned relationship-wise between an end-user device and the database(s) to facilitate operation of the methods that will be later described.

[0075] As indicated, these are only examples of how to implement a threat modeling system, and many other layouts are possible. System 100 may also include many other elements which are not shown for brevity. In a small business or organization wherein only one computing device may be needed to do threat modeling, system 100 could be implemented using a single computing device 102 with a database 108 stored thereon, or with one computing device 102 coupled with a server 106 and database 108 through a local connection (wired or hardwired), or using a cloud-stored database that the users access through user interfaces through remote

servers 122 and/or web servers 120. In an organization in which multiple computing devices may need to do threat modeling the system 100 could be implemented using a computing device 102 having the database stored thereon, or coupled through a local or Internet connection to a database 108 stored elsewhere within the organization's computing devices such as on a server 106 or remotely on remote servers 122 accessed via a web server 120, with other computing devices 112 and/or 116 coupled either directly with device 102 and/or 106 and/or through the telecommunication network 110. In implementations in which remote servers are utilized these may be scaled up to any needed number.

[0076] The threat modeling system and methods include the modeling of threats utilizing software which users access and interact with through a variety of user interfaces, some examples of which will be described hereafter, but a brief description of the processes facilitated by the software will now be discussed.

[0077] Referring to FIG. 2, a representative example of a threat modeling process (process) (method) 200 includes generating a threat model (model) 208 for any application, process, or system under consideration. By non-limiting example, this could include modeling the possible threats to commuting to work safely, modeling the possible threats to preventing the spread of an infectious disease, or modeling the possible attacks on a computer network (cybersecurity). Model 208 is used to generate an original threat report (report) 214 which in implementations includes identified threats, the status of identified threats (threat status), and the source(s) of identified threats, among other things.

[0078] As illustrated in FIG. 2, process 200 may include storing a plurality of threats 202 and threat model components (components) 204 in a data store 206. This may include, by non-limiting example, storing titles, definitions or descriptions, and/or associated images in the database 108 for each component and/or threat. The threats and components are used to create threat model 208.

[0079] Process 200 in implementations includes a user selecting from among the available components those components which are relevant to any given system, process or application. This is represented by the arrow between "components" and "relevant sources." The relevant sources 212 are relevant sources of threats that are determined by the system 100 either after, or while, the user is selecting the components that are involved in the specific application, system or process and defining relationships between and among the relevant components. The system also retrieves from the database relevant threats 210 that were previously correlated with the chosen components or combinations of components through the database,

IriusRisk
Ex. 1002 - Page 15

to form the threat model 208 (this is representatively illustrated by the arrow between "threats" and "relevant threats"). The threat model thus includes relevant threats and the relevant sources of those threats. The threat model is used to generate a threat report 214.

[0080] There may be some components with which no threat is associated (and they are therefore not relevant sources of threats), and there may be some threats that are dependent on one or more relationships between components. For example, when modeling a computing network some components may communicate with one another using a hypertext transfer protocol secure (HTTPS) protocol or instead with a transmission control protocol (TCP), and this relationship may determine whether there is a relevant threat to include in the model (or which relevant threat to include in the model). Although these relationships between components are communicative couplings and/or protocols in some instances, the relationships themselves may be considered "components" in a broad sense so that, in FIG. 2, the potential relationship types between components would themselves be considered components 204 that are stored in the data store and which the user utilizes to build a diagram of the system, application or process. Communication protocols may use conductive wires (electric signals), optic fibers (optic signals), wireless technologies (electromagnetic signals), and so forth.

[0081] FIG. 2 is a simplified diagram. Each component and each plurality of components is a potential relevant source for one or more threats. For example, one threat may be "Bluejacking" and one component, which would be correlated to this threat through the database, could be "BLUETOOTH port." Accordingly, if a user includes a BLUETOOTH port in a diagram of a computing system, the system 100 will identify that port as a relevant source for bluejacking in the associated threat model and threat report. The component in this example is a physical component of a computing device or system/network. In other implementations components/sources may not be physical components. For example, if one is modeling the threats involved in commuting to work safely one threat may be "freeway collision" and one component, which would be correlated with this threat through the database, would be "merging onto freeway." Thus, in this latter example "merging onto freeway" would be a relevant source for "freeway collision." In this example the component (and relevant source) is defined as an action or step, and not as a physical component.

[0082] Regardless of whether the threats and components/sources are physical elements or steps, actions, etc., the database allows for the correlation of any threat to any number of components and likewise, the correlation of any component to any number of threats. Accordingly, using the above example, the BLUETOOTH port may be defined as a relevant

source for a Bluejacking threat, but it may not be the only relevant source (another source may be a BLUETOOTH device wirelessly connected with the system being analyzed). Similarly, Bluejacking may be defined as one threat that may occur due to vulnerabilities of the BLUETOOTH port, but there may be other vulnerabilities or threats that are present in the system due to the BLUETOOTH port.

[0083] Referring to FIG. 3, another implementation of a threat modeling process (process) (method) 300 is shown. Method 300 includes the steps described above with respect to FIG. 2 but also includes additional steps. A modified data store 302 includes data store 206 but also includes compensating controls 304 stored in the database. The stored compensating controls include, by non-limiting example, a title, definition, image, and/or other items for each compensating control. Each compensating control may be associated with one or more threats and/or with one or more components and/or with one or more security requirements through the database (security requirements may in turn be associated with one or more components and/or one or more threats through the database). Method 300 includes user selection of one or more compensating controls (relevant compensating controls 308) from among all compensating controls 304 stored in the database, and the relevant compensating controls together with the threat model 208 previously discussed (in other words the relevant threats 210 and relevant sources 212) are included in the modified threat model 306. Modified threat model is used to generate modified threat report 310.

[0084] Referring now to FIG. 4, method 300 may include other steps and may be shown by threat modeling process (process) (method) 400 which includes creating a threat model of the application, system or process of interest (step 402), generating a threat report based on the threat model identifying relevant threats (step 404), generating a modified threat model with the status of one or more threats changed to "mitigated" using one or more of the relevant compensating controls (step 406), and generating a modified threat report and/or a change log including the threats designated as "mitigated" by the one or more relevant compensating controls (step 408).

[0085] Referring now to FIG. 5, another implementation of a threat modeling process (process) (method) 500 is representatively illustrated. Method 500 includes cycling through the process of selecting/deselecting a compensating control (i.e., adding it to the threat model, or removing it) as often as the user chooses. After the threat model is initially built and the initial threat report is generated (step 502) the user may define a compensating control (step 504) (this step may also be done before the threat model has been built or the threat report generated). The

IriusRisk
Ex. 1002 - Page 17

user may select the threats mapped to the compensating control and/or deselect threats mapped to the compensating control (step 506).

[0086] The user may add the compensating control to the threat model and thereby modify the model (step 508). System 100 matches threats identified in the original threat model with those mapped to the selected compensating control (step 510). The user at this point may be able to further modify the threats mapped to the compensating control. The status of every matched threat may be changed to "mitigated" or some other status by the user (step 512). A modified threat report may then be generated (step 514). The user is then brought to a point at which he/she may select whether to revert the model to its previous state (step 516). If the user selects to not revert the model, the process skips to step 522, otherwise the compensating control is removed from the model (step 518) and the status of matched threats are reverted to their pre-matched values (step 520). The user then determines whether to add another compensating control (step 522) and if so the process cycles to step 504 again, otherwise the process ends.

[0087] Reference will now be made to several example user interfaces which may be utilized to accomplish the above general processes and other processes as will be described. It should be stressed that these are only examples, and that other user interfaces could be used to accomplish the methods. Similarly, although specific user interfaces are described with respect to specific functionalities (dropdown menus, buttons, fields, tags, text prediction, etc.), the practitioner of ordinary skill in the art will be able to mix and match these functionalities and/or use other functionalities with the user interfaces to make the user experience intuitive and easy. For example, in instances where a dropdown menu is present this could be replaced by a search field, or a radio button selection, and so forth. Nevertheless, the user interface implementations as depicted in the drawings are useful for a variety of reasons, as will be understood.

[0088] Referring now to FIG. 6, a representative example of a user interface (interface) 600 is shown. This interface is titled the "knowledge base" page and may be displayed on any of the displays of system 100 shown in FIG. 1, depending on the particular system setup. Interface 600 has a number of selectors or menu items near the top of the screen such as, by non-limiting example, a three-bar menu selector, a "select all" selector, a "new" selector, an "edit" selector, a "delete" selector, a "copy to library" selector, a "select library" dropdown selector, a "home" selector, a "knowledge base" selector, a "templates" selector, a "profile" selector, a "change password" selector, and a "sign out" selector. In implementations some of these selectors may

IriusRisk
Ex. 1002 - Page 18

be further organized into dropdowns so as to take up less space on the interface. Additionally, in implementations of system 100 some or all of the menu items may be present on other interfaces, such that they are permanent menu items at the top of those other interfaces as the user navigates from one interface/page to another.

[0089] This "knowledge base" interface is an interface where the user may view and edit "components" that are stored in the database. The title "components" is thus present at the left near the top of the screen, and below this are three columns for "name," "type," and "labels." Below the column titles are search boxes where a user may begin typing and predictive text will bring up a list of items that begin with the letters typed so far. Each row shows one component type, name, and labels (if any). The list is scrollable as indicated by the scroll bar, and only a small number of components are listed here. For example, the topmost item is a WiFi router (named WIFI RTR in shorthand, and this and any other text used in the system may be in some type of shorthand or may be spelled out in its entirety in various implementations), its type is "Device," and no labels are applied. Shown in FIG. 6 are the following types: device, IoT device (which represents "internet of things" device), communication protocols (such as HTTPS), generic data store, database, application component, and deployment component. Other available types not shown may include, by non-limiting example: cloud, external system, FTP server, generic external interactor, generic process, generic trust boundary, Modbus, perimeter control, third party software, third party library, transaction, WINDOWS application, WINDOWS OS, and so forth. These are only representative examples, and the user may define new types as desired and apply those types to a new component (or existing component), which will associate the type with the component name through the database. The rows may be organized in alphabetical order by any of the columns (name, type, labels) by clicking on the title of the column and may be organized in reverse alphabetical order by again clicking on the same column header (and toggled with continued clicking). The labels column may be useful for, among other things, searching for components that have a label applied. For example, if the user is looking for embedded system components, the user may search in the labels column for "embedded system" (or a shortened version if the system uses a shortened version) to find all components that have the embedded system tag applied.

[0090] A user may select any row by clicking anywhere in that row. The selected row will be highlighted, as representatively illustrated in FIG. 6 wherein the "login" row is highlighted. Once a row is highlighted any data associated with the component through the database is

shown to the right. For example, in the middle of the screen the component name "login" and its type "application component" are shown. Below this the tags applied to this component are shown, which in this case include: web, application, feature, authentication, password. A description can be associated with the component through the database – in this case no description has been provided. If a user desires to edit the name, type, labels, or description the above "edit" selector may be selected and the user will be able to input new information for any of these fields, which upon pressing a "cancel" button will revert to the prior data or upon pressing a "save" button will write the new data to the database.

[0091] Interface 600 also shows a "properties" section to the right, which shows the threats and security requirements associated with this component through the database. The user may press the plus icons to add new threats or security requirements, as desired, or may use the "X" icons to remove threats and/or security requirements associated with the component through the database. In this example the login component has the following threats associated with it: inducing account lockout; session hijacking; and clickjacking. The security requirements associated with the component in this example are: parameterized queries – dynamic SQL; breach – randomizing secrets per request; and encryption algorithm (the first two only show a short description on the interface screen for ease of viewing).

[0092] The threats and security requirements associated with the component through the database will at least partially facilitate the later threat modeling when a user is depicting a specific system layout. For example, if a user is depicting a computer system layout and adds the "login" component to the system layout then one or more or all of the associated threats may become "relevant threats" to the overall system as previously described. In some systems 100 all the threats associated with this element would become "relevant threats" to the specific application, process, or system being modeled, and in other systems 100 some of these threats may be automatically dropped from "relevant threats" for the specific application, process or system being modeled if other components added to the system would inherently mitigate those threats (such as an HTTPS protocol instead of HTTP protocol being used between two components).

[0093] With regards to the other selectors, a user may select the "select all" option to select all components then listed (if the user has done a search this would limit to selecting all those components populated by the search), and the user may then press the delete selector to delete all of these components and their relationships (associated threats, security requirements, tags, descriptions, etc.) from the database. The user may instead, once all are

IriusRisk
Ex. 1002 - Page 20

selected, press the "copy to library" selector to copy the selected items to a default library set up by the user previously through another interface. The user may similarly select only one or a few components to delete or copy to the default library. The "select library" selector allows a user to display various libraries, here the "ThreatModeler" library is displayed but the user may create his/her own libraries, may view other built-in libraries or third party libraries accessible through the interface through a web server or other server which communicates with another third party database, and so forth.

[0094] The user may select the "new" icon to bring up a popup interface in which the user may define a new component, giving it a name, selecting the component type from a dropdown menu, adding an image to be displayed to represent the component (such as a lock for a security feature, a USB indicator for a USB device, etc.), labels, and a description, and may then select a "cancel" selector to cancel the new component or may select a "save" selector to write the new component information and relationships to the database, and selecting either save or cancel will also remove the popup interface and revert back to interface 600.

[0095] The home menu item at the top right of the screen will navigate the user to a home screen. The knowledge base icon will navigate to interface 600. The templates selector navigates to an interface similar to interface 1000 of FIG. 10 (hereinafter described) but with a blank canvas 1002 for creating a new template which may be used later for creating other relational diagrams (diagrams) 1004 (also described hereafter). The profile selector brings up a popup window which displays the current user's name, email, department, role (admin, etc.), and last login date/time as stored in the database, and a close button to close the window. The change password selector brings up a popup window to change the user's password with a "close" selector to cancel the change and a "save" selector to write the new data to the database. In implementations the home, templates, and knowledge base icons are permanently displayed at the top of all interfaces of system 100 and the profile, change password, and sign out selectors are available from a dropdown icon which is also displayed at the top of all interfaces of system 100. The sign out selector allows the user to log out of the system. Interface 600, as well as other interfaces of the system, may also include a refresh selector to refresh the page and a help icon to bring up a help menu. An icon displaying a title and/or logo of the software may also be displayed at the top of the screen (such as the top left) which when selected may navigate to the home screen.

[0096] When the triple-bar menu icon is selected it brings up a popup expandable menu which, when all items are fully expanded, appears as interface 700 of FIG. 7. The "threat library" menu

IriusRisk
Ex. 1002 - Page 21

item includes the sub-menu items "threats," "test cases," and "threat agents," the "security requirements" menu item includes the sub-menu items shown, the "threat patterns" menu item includes the sub-menu items shown, and the "enterprise" menu item includes the sub-menu item "users." When any of the sub-menu items are selected an interface very similar to interface 600 is displayed.

[0097] For example, FIG. 8 shows interface 800 which is accessed by selecting the "threats" sub-menu item. The title "threats" is shown near the top of a list of threats which are organized into rows and columns showing each threat's name, risk level, and attached labels. Several threats are shown, and any column may be searched using the search box directly below the name, risk, or labels headers, which behaves similarly as described for FIG. 6, and the data may be organized alphabetically (or reverse alphabetically) by any column as described with respect to interface 600. The selected threat is highlighted, and its information is displayed to the right including the name, risk level, labels, and a description, which in this case includes a URL. The properties section includes threat agents which may be added, though none are added to this threat—but examples include things like "insider," "black hat," "hacktivist," "cyber terrorists," "authorized external user," and so forth which are threat agents that had previously been identified by the user. Threat agents may be added and removed. Test cases may also be added and removed; this threat has three test cases added which were previously entered into the database by the user. Security requirements may also be added by the user and, in this way, compensating controls may be associated with specific threats through the database—this is one of the ways the system facilitates step 506 of FIG. 5. Security requirements may be added to the database originally through another interface, described hereafter, and identified there as a compensating control, then added to a specific threat from interface 800. Such an association through the database facilitates the system displaying various compensating controls in a diagrammed system, method or application and the threats those compensating controls can mitigate.

[0098] The menu items at the top of FIG. 8 are the same commands/links as those shown in FIG. 6 but are shown here in icon format to give another representative example, where the select all selector is represented by a checkmark, the new selector is represented by a plus icon, the edit selector is represented by a pencil icon, the delete selector is represented by an "X" icon, the copy to library selector is represented by an icon of two documents, the home selector is represented by a home icon, the knowledge base selector is represented by a file folder icon, the templates selector is represented by a document icon, and the profile, change password,

IriusRisk
Ex. 1002 - Page 22

and sign out selectors are available by selecting a dropdown menu represented by a downward facing triangle icon.

[0099] Referring back to FIG. 7, if the "test cases" sub-menu item is selected an interface similar to interfaces 600/800 is displayed, but showing test cases, and allowing the user to add, edit, delete, copy, add labels, and so forth, similarly as previously described with respect to the "threats" sub-menu item. The threat agents, security requirements, code snippets, code reviews, components (already described with respect to interface 600 since the components interface is set as the default interface when the "knowledge base" selector is selected), data elements, roles, widgets, component types, attributes, and user sub-menu items all have similar functionality and bring up similar interfaces when selected. Data elements, which represent data elements that may be captured by any diagrammed system, application or process (such as credit card numbers, billing addresses, pins, phone numbers, email addresses, order history, birth date, medical history, insurance history, and so forth) may be associated with specific threats and with specific security requirements in the same way as those associations may be made for components.

[0100] If a user selects the security requirements sub-menu item an interface similar to interfaces 600/800 will be shown similar to interface 900 of FIG. 9, which allows the user to order security requirements alphabetically (or reverse) by name or label and search for specific security requirements. Selecting any security requirement will highlight it and display its name, description, labels, an indicator to indicate whether the security requirement is a compensating control, and any code snippets or code reviews (under a "properties" header) that have been associated with the security requirement (which may be added or removed from that interface similar to what is described with respect to the "properties" elements of interfaces 600/800).

[0101] Interface 900 is actually the interface visible when the user selects the "new" selector to create a new security requirement. From this window the user may add a name, add a description, check a box (or leave it blank) to indicate whether the security requirement is a compensating control, add any labels (which are previously input into the database by the user), and press cancel to abort the addition or press save to write the data to the database.

[0102] From any of the previously mentioned interfaces, if the user presses the home selector an interface similar to interface 1300 of FIG. 13 will be displayed, which lists all previously stored threat models by name and showing their version. Either column may be organized alphabetically or in reverse, and the PDF button will export the list to PDF. If any specific threat model is selected it will be highlighted and its associated threat report (threat report interface)

1302 will be displayed, which will be discussed hereafter. From the top menu items the user may select the new selector to create a new threat model, the edit selector to edit the name, version, risk level, an "internal" toggle, and labels associated with the selected threat model, a delete selector to delete the selected threat model, a diagram selector to view the diagram for the selected threat model, a report selector to export to PDF the threat report (which shows for each threat the threat name, source, risk level, status, and creation date), a threat tree selector to view a diagrammed threat tree, showing threats of the threat model, and other selectors already described.

[0103] If the threat tree selector is selected a threat tree is displayed in which threats are organized as sub-elements of data elements and/or components, and also displayed are mitigating security requirements or compensating controls that may be implemented to mitigate the threats and an indicator of whether each threat has been mitigated. Portions of the tree may be collapsed, expanded, or viewed in vertical tree or horizontal tree format. The interface showing the tree diagram has selectors to zoom in, zoom out, revert to 100% zoom, toggle on/off a thumbnail overview image in a corner of the display, save an image of the tree, open a legends window which shows the colors in which various elements are diagrammed (threat agent, widget, component, role, data element, security requirement, threat, protocol, node, and project), and a filter selector which allows the user to remove and add back in any of the aforementioned items from the tree. The tree may be dragged and moved in any direction for viewing any portion of it easier.

[0104] Continuing with FIG. 13, if the new selector is selected (which, as with all other selectors, could be implemented as an icon or logo absent any wording), a popup window appears allowing the user to enter a name, version, toggle an "internal" indicator, select a risk level from a dropdown (from among previously determined risk levels stored in the database), apply labels (which allow the user to type and populate a list of already entered labels stored in the database to select one, or to add a new label by typing a new item and selecting "add new"), or the user may select a "new from template" button to begin a new diagram from an existing diagram template, or a "new from features" button to begin a new diagram from user-selected features, or the user may select "import" to begin a diagram from a previously stored diagram (such as from an older version of the software), or the user may select "empty" to begin a diagram with a blank canvas.

[0105] If the user selects "new from features" the diagram interface of FIG. 15 displays with a blank canvas 1002, and a popup selector (not shown) allowing the user to select "business

IriusRisk
Ex. 1002 - Page 24

requirement" features which are desired to be included in the model such as, by non-limiting example, authentication, funds transfer, credit verification, bill pay, or other features already stored in the database previously by the user, and after selecting the desired features, the blank canvas will populate with a node for each feature. The user can then modify and/or add components to the diagram as desired. In implementations each of these added "business requirements" will add a "component group" to the diagram and threat model, as will be described hereafter.

[0106] If the user selects "empty" the diagram interface (interface) 1000 of FIG. 15 will appear, showing the blank canvas 1002 as shown. To the left of the canvas is a toolbox module which may be minimized using the left facing double arrows and re-expanded by selecting the same arrows which will then be right facing. To the right of the canvas is a properties module which also may be minimized and expanded in similar fashion, along with sub-menu items "general," "components," "data elements," "roles," and "widgets" each of which may be expanded downward or minimized upward using the double arrows (in FIG. 15 they are all minimized).

[0107] The toolbox module is searchable, with the user able to begin typing and the system suggesting through a list populated just below the search box components which begin with the letters (or include the letters) typed by the user. The dropdown to the right of the search box may be used to allow the user to search from among all components or a subset such as one or more specific component libraries as previously discussed, with ThreatModeler being an example component library) and/or one or more specific component types as stored in the database (and previously discussed with respect to FIG. 6). The user may also scroll down through the components using the scrollbar function.

[0108] In the example of FIG. 10 the components in the toolbox are of varying sizes, but in implementations they may all be of similar sizes and shapes (or identical sizes and shapes). In implementations in which the system is deployed within a single organization with only a local library of components the components listed in the toolbox will be those that have been input by the user or that were previously loaded into the database during software installation. In implementations in which the system includes some remote communication with other libraries the toolbox may display components available from other parties, such as through a cloud computing services, e.g., MICROSOFT AZURE or the like (in implementations the entire system and methods could be implemented using cloud computing in instances where a local software installation or local database are not desired).

IriusRisk
Ex. 1002 - Page 25

[0109] The toolbox, as can be seen, includes components that were previously entered into the database through interface 600 as previously described. From interface 1000 the user may select a desired component and drag it to a desired location on the canvas, and by dragging multiple items the user may begin to add the components of a system, application or process to be modeled. By non-limiting example, FIG. 10 shows interface 10 with a relational diagram (diagram) 1004 already fully created by the user. As can be seen, the user has dragged multiple components onto the canvas 1002 and has defined various relationships between them. The user dragging components to the canvas is one way in which the user selects relevant components or relevant sources as described with respect to FIG. 2, and it is from these relevant components (relevant sources) that the system identifies relevant threats by identifying the threats that are associated through the database with those components.

[0110] In implementations a communication protocol (protocol) can be defined between components by clicking on a component and dragging the cursor to a second component. This will create an arrow, such as those shown on FIG. 10 between components on the canvas, and will display a default protocol (here the default protocol is HTTPS), but a user may right click the arrow to display a popup selector allowing the user to deselect HTTPS and/or select one or more other protocols (such as TCP, SMB, WiFi, 3G, 4G, AJAX, binary, BLUETOOTH, FTP, FTPS, HTTP, IMAP, MAPIRPC, SMTP, SMTPS, USB, etc.), and when any protocol is selected that protocol name will be displayed (or if multiple are selected then "multiple" or some term/icon meaning "multiple" will be displayed - in the image this is shown as "MULT" for short)—these protocol arrows will also be color coded by type (such as green for HTTPS, red for multiple, and so forth). Protocol arrows may also be deleted as desired. The alphanumeric descriptions are shown here next to the link arrows, but in implementations the wording may be superimposed over the link arrows. They are shown here next to the arrows for easier readability.

[0111] The components displayed in the toolbox of FIG. 10 are generally shown without icons. The USB Port, SSO (single sign on), BLUETOOTH PORT, and BLUETOOTH PROXY components are shown with icons as an example that all components may be shown in the toolbox with an icon if desired. It may also be seen that the user may input components that constitute branded software elements, such as PAY PAL, AKAMAI DNS, SKYPE, etc. In implementations all compensating controls will be displayed in the toolbox with a similar icon, such as the padlock icon shown for the SSO component of FIG. 10.

[0112] As a user adds components and links them together, the threat model includes the threats that are associated with the components/protocols. As previously described, each

IriusRisk
Ex. 1002 - Page 26

component, each protocol, and each data element may be directly associated with one or more threats through the database. Accordingly, each time a component is added to the diagram, or a new link drawn between components and protocol selected, if there are any threats associated through the database with the component and/or protocol those threats are then included in the threat model. A user may at any time right click on a blank area of the canvas when no component/link is selected and a "threat report" item will be selectable which, when clicked, will generate a popup threat which will include a list of all threats. In implementations this will look much like the threat report 1302 shown in FIG. 13, and will include a first "threats" tab list of all threats (and the number of threats), identifying the risk level of each threat, showing the threat status (open/mitigated), identifying the source of the threat (which is a component on the diagram), and having an actions column from which the user may display a description of the threat (previously associated with the threat through the database) (the description selector here shown with the letter "D" though the selector could instead show "description" or the like) and leave a comment to store to the database (the comment selector having the phrase "note" though in other implementations it could say "notes" or "comment" or the like). The user may organize the table by any column (alphabetically or reverse) and may also organize by column headers by dragging the column headers to the location indicated. For example, the user could drag the "status" header to the identified location and then drag the "risk" header to the right of it and the table would then be organized first by status and then by risk level. When this "dragging" is done a "ghost header" is dragged up to the identified location in the sense that the existing header remains in place. The organization may be reverted to an earlier state by deleting the ghost headers using an "x" icon.

[0113] The threat report displayed also includes a security requirement tab (showing number of security requirements) and displaying a table similar to the threats table. The security requirements table lists the security requirements that are associated with one or more of the identified threats in one column, lists a source in another column (which is the component associated with the threat), includes an "implemented" column indicating whether the security requirement has been implemented, an "optional" column indicating whether the security requirement is optional, and a similar "actions" column to the previously described actions column, this time the description giving a description of the security requirement and allowing the user to record a written note. This table may be organized similar to the previously described table.

IriusRisk
Ex. 1002 - Page 27

[0114] With respect to the "implemented" column, in implementations this is a checkbox for each security requirement indicating whether the security requirement has been implemented. The user may manually check those security requirements that have been implemented and manually uncheck those that have not. This column may also have been previously populated with some "implemented" checkmarks based on mitigations that the user has selected for specific compensating controls, which will be discussed later with respect to FIG. 11. Further, the "optional" column may also include checkboxes which a user may manually toggle to indicate which security requirements are optional or not. This column also may be populated by the user selecting certain mitigations as will later be discussed with respect to FIG. 11—for example if there are two security requirements associated through the database with a specific threat, but only one needs to be implemented, then if the user selects one of those security measures as mitigating the threat the other security measure may populate on this list as being checked "optional."

[0115] The threat report further includes a "test cases" tab (showing the number of test cases) and displays a table listing test cases associated with the threats through the database. As previously described with respect to FIG. 8, each threat may have test cases associated with it. An example test case would be, for example, a "dictionary-based password attack" for a password-related threat. The test case tab would list this test case and have an "actions" column which, when an icon is selected, pops up a description, which in this case states "use a password cracking tool what will leverage the dictionary to feed passwords to the system and see if they work" and further list techniques like setting the test to try all words in the dictionary, common misspellings, and combinations of words and common misspellings of combinations. The test case tab/table thus offers tests that the user may try to test against the threats both before and after security requirements are implemented. This table may be organized and sorted similar to the previously described tables.

[0116] The threat report further includes a "code reviews" tab (showing the number of code reviews) and displays a table listing the code reviews that are associated with the security requirements through the database (as previously shown with respect to FIG. 9). An example code review would be, for example, an "Authentication: Weak Password" code review. The code review table would list this code review and have an "actions" column which, when an icon is selected, pops up a description, which in this case states "Password strength should be enforced upon a user setting/selecting one's password" and gives examples of code that may be used to set up such an enforcement. The code review tab thus offers sample code that the user

IriusRisk
Ex. 1002 - Page 28

may use to implement measures against threats which may complement or work together with implemented security requirements. This table may be organized and sorted similar to the previously described tables.

[0117] Multiple selectors are shown at the top of interface 1000 in addition to the permanent selectors that are available on several interfaces. The "select all" selector selects all components on the canvas (the user may then deselect some, if desired, for example for grouping some or all the components as later described). The "copy" selector copies selected components and the "paste" selector pastes the copy onto the canvas (this may also be done with keyboard shortcuts, and shortcuts may in implementations be used for all other selectors described herein for system 100). The "delete" selector deletes the selected components from the canvas and the "clear canvas" selector deletes all components from the canvas. The "zoom in" and "zoom out" and "100%" zoom in and out of the canvas and revert to a 100% zoom, respectively. The "comment" selector populates an empty comment box on the canvas in which the user may type a comment which will remain on the canvas until later removed (though it may appear in minimized or icon format until hovered over or opened). The "overview" selector displays a thumbnail overview of the entire diagram in a corner of the canvas.

[0118] The "undo" selector undoes the last action (or multiple last actions if selected multiple times) and the "redo" selector does the opposite. The leftmost "templates" selector pops up a menu from which one or more previously saved business requirement "features" may be selected to add to the canvas (these may be nodes and/or component groups, for example, and may be the same nodes/component groups that are described above when the user uses the "new from features" function when generating a new model). The "save as image" selector has a dropdown and allows the user to save an image of the diagram as then displayed in one of various formats such as PNG, SVG, etc., or to take a "snapshot" which saves the diagram at the present state to the database. The "PDF" selector generates a PDF of the threat report based on the current configuration of the diagram components (the threat report described to some extent above). The "settings" dropdown selector allows the user to toggle between straight or curved link lines (protocol lines/arrows), showing or hiding the link wording (in FIG. 10 the lines are curved and the link wording is shown), making the diagram public or non-public, and showing or hiding comments (the comment function described above).

[0119] The "save as template" selector allows a user to save the entire diagram as either a threat model or a business requirement "feature" (in other words saving it as a component

IriusRisk
Ex. 1002 - Page 29

group) and in either case the user may give the feature a name, add labels, and select the type as either threat model or business requirement, then press cancel to cancel or save to store the new template to the database. The "snapshots" selector populates a list of previously saved snapshots, any one of which may be opened from the list or deleted.

[0120] The "group" and "ungroup" selectors allow the user to create a group containing multiple components or to delete a group (but not the included components). As seen in FIG. 10 for example, there is a "WINDOWS 7" group which includes file system, PDF client, SKYPE, OUTLOOK 2010, MCAFEE HIPS, MCAFEE AV VSE, IE11, BIT LOCKER, MCAFEE DLP, and OFFICE 2010 components. There is also a Laptop group containing the WINDOWS 7 group and further containing WiFi port, USB port, ethernet port, HDMI port, and BLUETHOOTH port components. Finally, there is an Office Network group which includes the Laptop group and also includes file server, SHAREPOINT, printer, WiFi access point, IRONPORT, email server, BLUECOAT proxy, and SSO (single sign on) components. Then there are other components (external email gateway, AKAMAI DNS) which are not part of any group. A group may be formed from any one or more components, and the AZURE group is seen containing only a single component: ONEDRIVE.

[0121] When a group is formed the user may, using the right-side "general" dropdown, add a custom display name, select a container type (from container, trust boundary, collection, or some other option stored in the database previously by the user), select a component (for example for the WINDOWS 7 the component "WINDOWS machine" is selected, which shows that some components stored in the database may include other components), select other display choices such as title and background/border color, and select a Common Platform Enumeration (CPE) identification from among a list previously stored in the database (for example in this case ID a version of WINDOWS 7 is selected) (the list of CPE IDs may be searched using filters to easily find the appropriate one), and the user may also add notes.

[0122] As further examples, the WINDOWS 7 group is identified as a container group, a WINDOWS machine, and a specific WINDOWS 7 operating system version is identified. The Laptop group shown in FIG. 10 is listed as a container group and no CPE ID is selected. The Office Network group is identified as a trust boundary group type and no CPE ID is selectable. The AZURE group is identified as a trust boundary group type and no CPE ID is selectable.

[0123] Each grouping of components, however, could be diagrammed separately as an independent threat model and then saved as a component so that it may be imported into another threat model/diagram. When a user adds any of these component groups to a blank or existing diagram/threat model the threat model of the component group is added to (and/or

nested within) the threat model of the existing diagram/threat model. In this way the user can modify a threat model by incorporating previously defined threat models. This ability is generally termed "threat model chaining" herein and is a useful mechanism for allowing a user to diagram complex systems/processes without having to repeatedly build common elements among the systems/processes.

[0124] Each component group may thus be redefined as a discrete "component" and may then be included as a single icon in the toolbox menu. By non-limiting example, referring to FIG. 10, the WINDOWS 7 component group could be defined as a component, then the user could, in another diagram (or the same diagram), select and add a WINDOWS 7 component to the diagram to import into the diagram and associated threat model the threats associated with the WINDOWS 7 threat model. That same could be done for the LAPTOP component group. Accordingly, a component group and associated threats added to a diagram may in turn already include other nested/chained threat models therein, so for example if a user defined a laptop component group such as that in FIG. 10 as a "WINDOWS LAPTOP" component then, when a user later adds a WINDOWS LAPTOP element to a diagram/threat model by selecting a WINDOWS LAPTOP component from the toolbox and dragging it onto the diagram, the threats associated with the laptop itself, as well as the nested/chained threats associated with the included WINDOWS 7 threat model, are automatically included in the threat model for the then displayed diagram.

[0125] Referring back to FIG. 10, the "compensating controls" selector may be selected to display a popup similar to compensating control report (report) 1200 of FIG. 12. This report is populated from previous relations stored in the relational database. As indicated previously with respect to FIG. 6, each component may be associated with one or more threats and one or more security requirements through the database using interface 600, and as described with respect to FIG. 8 each threat may be associated with one or more security requirements through the database. As further indicated with respect to FIG. 9, each security requirement may be identified through the database as a compensating control. Accordingly, based on these relationships and selections the list populated in FIG. 12 shows all possible compensating controls that could be implemented to mitigate threats that are present in the diagrammed system, application or process.

[0126] By non-limiting example, the threat of "physical theft" is associated with the component "laptop" through the database, but no security requirements are directly associated with the laptop component. Nevertheless, the "physical theft" threat is associated with the security

requirements "MCAFEE FRP" and "BITLOCKER" through the database, both of which security requirements are identified through the database as compensating controls (MCAFEE FRP being a USB encryption tool and BITLOCKER being a disk encryption tool). Accordingly, when a user adds the laptop component to a diagram, if the user selects the "compensating controls" selector this list will then include both MCAFEE FRP and BITLOCKER as compensating controls for the physical theft threat. If the user removes the laptop component from the diagram (deletes it) then the MCAFEE FRP and BITLOCKER compensating controls will no longer appear in the list (unless they are also associated as compensating controls for some other threat present in the diagram). In implementations the list populated will also show compensating controls which are directly associated with a component that is added to the diagram. In other words, in the above example there are no security requirements directly associated with the laptop component, but if a third security requirement were directly associated with the component through interface 600, and if the third security requirement was identified through the database as a compensating control, then that third compensating control would also populate in the list of interface 1200.

[0127] Referring again to FIG. 10, it may be seen that this diagram includes the MCAFEE DLP, BITLOCKER, MCAFEE HIPS, and MCAFEE AV VSE components grouped in the WINDOWS 7 group. The user may, for example have modeled the WINDOWS 7 group, then used the compensating controls selector at the top of interface 1000 to identify that there are certain threats that these components would mitigate. The user may then add those components to the WINDOWS 7 group by dragging them from the toolbox to a location within the group outline to add them to that group. Then, upon right clicking on any specific security requirement component, the user may select a "mitigations" selector which pops up mitigations interface (interface) 1100 as shown in FIG. 11. The user could add one of these components at a time and model the mitigations, generate a new threat report, then either remove that component or add another compensating control, and in this way test out various compensating controls. This is one way in which the system facilitates steps 512, 514, 516, 518, 520, and 522, among other steps, of FIG. 5.

[0128] Interface 1100 displays a table which lists all threats and their associated sources (component associated with the threat), risk levels, and status, and highlights the listed threats which the security requirement is configured to mitigate as identified previously through the database through interface 800 (these threats are "mitigatable" by the selected security requirement). The user may deselect highlighted threats and/or may highlight other threats,

then may select "close" to cancel or may select "mitigate" to toggle those threats to mitigated through the database. Once this is done, for example, the threat report as seen in FIG. 13 (or popup shown from the diagram interface) will list the mitigated threats as "mitigated" and the compensating controls list shown in FIG. 12 will also show the mitigated risks as mitigated.

[0129] The threat report of FIG. 13 allows a user to manually change the risk level and threat status directly from the home menu using the "change risk" and "change threat status" drop downs once a threat is selected by clicking on any row. Changing the risk level of a threat from this interface will only change the risk level for this specific threat model. Threat statuses that may be used include, by non-limiting examples: open, closed, mitigated, fixed, not applicable, needs more details, not tested, secure usage practice, and so forth.

[0130] The systems and methods described herein may also be used for analyzing an attack surface of a complex system or process, which will be described now. In implementations the attack surface of a system or process is comprised of the sum of all open and unmitigated potential threats to an "asset" identified through threat modeling.

[0131] As described herein, a modeled system or process may include a modeled deployed or contemplated computing network, which could include by non-limiting example: one or more applications; one or more on-premises infrastructures; one or more cloud-based infrastructures; one or more hybrid infrastructures; serverless architectures; microservices; one or more embedded devices; one or more IoT devices; one or more mobile devices; one or more Industrial Control Systems (ICS); one or more cyber-physical systems (CPS); one or more third party systems; one or more organizational networks or intranets; non-cyber elements interacting with a cyber system such as, by non-limiting example tangible assets, intangible assets, property(ies), plant(s), equipment, liquid assets, brands, reputation, residential structures, real property, utility services, unattached removable items and assets; an infrastructure system, such as a transportation infrastructure, an electrical grid, a telecommunications network, and so forth, all in implementations accessible and utilized through a highly interconnected network of intranets, wireless networks, and the Internet.

[0132] All such components of a modeled process or system may include potential threats which, if discovered and exploited by an adversary, may yield one or more attack vectors to one or more assets of the system or process. The "asset" may be any item selected by a user to be identified as an asset. The modeled "attackers" included in any threat model may be actually modeled as a person in implementations, though in other implementations may be simply

IriusRisk
Ex. 1002 - Page 33

modeled as an entry point or device included in the process or system which an attacking person may use to cause harm.

[0133] It may be pointed out that increased interconnectivity of a computing system with other systems (such as the Internet, third party systems, end user systems, etc.) may increase economic value and efficiency though these may also increase organizational risk due to the increase in adversarial actors and a constantly evolving threat landscape. The threat modeling chaining and attack simulation systems and methods described herein allow organizations to manage threats at a comprehensive organizational level notwithstanding an ever-changing threat landscape.

[0134] A modeled "attack" as used herein is a modeled attempt by an adversarial entity to traverse the modeled system or process from an attack surface to one or more assets within the system or process. The routes from all potential attackers to a selected asset are the "attack vectors" or attack paths to that asset.

[0135] Referring now to FIG. 14, a block flow diagram illustrates steps that may be included in a general attack surface analysis. Step 1402 includes user generation of a diagram of a system or process, as has been described previously with respect to FIG. 10. Step 1404 includes system generation of a threat report for each component or component group and step 1406 includes system combination of the individual threat reports into a comprehensive threat report for the overall diagrammed system/process (this would include, for example, including threat report elements for nested or chained threat models, as has been explained above), and then steps 1408-1414 include steps which may occur in any order. In step 1408, once a user has selected an asset to analyze, the system summarizes data to show all attack vectors associated with threats which may compromise that asset. At step 1410 the user analyzes the various attack vectors to determine what compensating controls may be included to protect the asset. At step 1412 the user adds or removes compensating controls to the diagram and/or toggles compensating controls between ON/OFF states. At step 1414 the user determines the effectiveness of the compensating controls or other risk management methods (such as changing communication protocols, changing the relative location of the asset within the modeled environment, adding non-compensating control elements between the asset and attack locations, and so forth).

[0136] FIG. 16 further breaks down step 1402. At step 1602 the user identifies the components of the system or process and chooses their relative placement within the diagram, at step 1604 the user identifies the type and placement of any communication protocols between

IriusRisk
Ex. 1002 - Page 34

components, and at step 1606 any other attributes of components (including protocols) may be further designated (this step may be excluded in some methods, and these steps may be done in any order in implementations).

[0137] FIG. 17 shows a tree diagram that representatively illustrates a method of determining an attack vector for a selected asset. After a system or process is modeled using the diagram interface, described above, then in the diagram interface or in another interface either an asset is selected or an associated component is selected and relevant threat attack paths illustrated. For example, in the topmost node of FIG. 17 an asset 1702 is shown, and the attack tree or threat tree shows that the asset is associated through the database with four components. Components 1704, 1706, 1708, and 1710 are each associated with threats (i.e., the threats that are associated with each component through the database). Accordingly, threats 1712, 1714, 1716, 1718, 1720, 1722, and 1724 are shown. It may be seen that the system then determines that threat 1722, through component 1708, is a threat to asset 1702, and thus attack vector 1726 (shown by lines of heavier weight) is shown between threat 1722 to asset 1702. The diagram of FIG. 17 may be shown on an interface of the system, though in implementations FIG. 17 simply models the method the system is utilizing behind the scenes to show the attack vector(s) in the diagram interface, as in some implementations (such as those shown in the drawings) the threats are not explicitly shown on the diagram interface(s). FIG. 17 shows an attack vector by which an adversarial entity may target the asset, i.e., by exploiting threat 1722 of component 1708. It should be noted that the asset may be some sub-component within a diagrammed component, or it may be some element that is somehow associated with the component (for example the component may be a database, and the asset may be sensitive client information stored within the database—in that instance the asset could be defined broadly as the database, or it could be defined narrowly as the sensitive information within the database, other information in the database not being included in the definition).

[0138] It is noted that each asset could have any number of threats that could threaten to compromise it. Accordingly, the system identifies all threats which may compromise the asset in order to visually diagram attack vectors for the user. Some assets may be threatened by only a single threat, some may be threatened by two or three threats, or more, and so forth. Along these same lines, it is pointed out (as has been discussed to some extent above) that each individual component (or grouped set of components) of a threat model could, itself, be associated with its own threat model through the database. Because of this, the overall threat model that is shown (for instance in FIG. 13) in implementations could be called a threat model

IriusRisk
Ex. 1002 - Page 35

portfolio as it includes all sub-threat models and nested threat models. For example, an overall threat model A could include components B, C, and D. Component B could be a single component, component C could be a previously modeled group of components having its own threat model, and component D could be a previously modeled group of components having its own threat model that also includes therein a nested threat model for a component group E. Accordingly, the threat model A would include all threat models associated with components and component groups B, C, D, and E, including all nested threat models. As described above, this "threat model chaining" may allow for quick and simple building of process/system models without having to recreate commonly included system/process elements.

[0139] In implementations one or more interfaces of system 100 may be utilized to list the top potential threats (such as the top ten potential threats), the top most vulnerable components, the composition of threats by risk type, the composition of threats by status (mitigated, unmitigated, open, etc.), and so forth. The data may of course be organized and displayed in many other ways. In implementations an interface could list each threat, list each threat risk level, and list the source(s) for each threat. In implementations each threat source listing may include a component and an underlying source, so for example one component may be associated with multiple sources of a threat. An example would be an online banking interface which is shown on a diagrammed threat model as a component, and a threat may be a virus being introduced to the bank's environment through a breach in the banking interface, and one source of the threat may be a legitimate user who unwarily exposes the ATM to a virus on his/her computer, and another source of the threat may be an individual hacker seeking to introduce the virus to the banking environment, and another source of the threat may be a state-sponsored entity with high end mechanisms to seek to introduce the virus to the banking environment.

[0140] In implementations an interface of the system may list the most vulnerable components in ascending/descending order, along with the risk level for each listed component, the number of total threats to that component, and the number of open or unmitigated threats for that component. In implementations an interface of the system may list multiple data elements associated with components (for example user names, social security numbers, financial info, credit card numbers, employee ages, etc.) (in this case the component likely being a database housing the data), a data classification for each data element (such as confidential, restricted, public, etc.) and an exposure level for each data element (such as very high, high, low, etc.). In implementations an interface of the system may break down the risk levels by percentage, such

as 45% very high risks, 25% high risks, 15% medium risks, and so forth. In implementations an interface of the system may break down risk status by percentage, such as 45% open threats, 25% closed threats, 30% mitigated threats, etc. In implementations an interface of the system may visually illustrate the number of threat introductions plotted versus time, such as the number of threat introductions by month.

[0141] Any of the interfaces discussed herein may, as would be expected, change (and in implementations may change dynamically) as edits are made to the diagram (for example when a compensating control is toggled between ON/OFF states, this may change many or all of the interface displays accordingly by varying percentages, mitigating threats, etc. Thus when a compensating control is introduced and/or turned on or off, the threat model, threat report, mitigations interface, compensating control report, and so forth may all change accordingly to reflect the updated configuration.

[0142] It is pointed out that any threat model of the system may be subjected to an attack vector analysis. So, for example, an attack vector analysis may be performed on a simple two-component diagram, or an attack vector analysis may be performed on a hundred-component diagram that includes the two-component diagram as a nested sub-threat model.

[0143] In implementations an interface of the system may show a display similar to FIG. 17 except mapping all components/assets vulnerable from a single threat. In other implementations an interface of the system may show a display of all threats affecting a single component or asset. In the latter case the visual display may show attack vectors branching outwards from the asset, tracing communication protocols and extending through multiple components as far back as the locations/sources of the underlying threats. This is representatively illustrated in FIG. 18, which shows a type of display that may be shown on an interface of the system.

[0144] FIG. 18 shows components A-G. Component A has been selected as an asset, and the attack vectors of all threats threatening the asset (component A) are diagrammed using arrows. The arrows in FIG. 18 are shown in different formats so that they can be distinguished from one another. In other implementations they may all have the same appearance. The arrows are also shown pointing away from the asset, but in other implementations this may be reversed so that arrows point towards the asset from the underlying threats. It may be seen that a threat emanating from component G threatens to compromise component A (or the asset within or associated with component A), the attack vector 1802 passing through component F and component B to reach component A. A threat emanating from component E also threatens to

IriusRisk
Ex. 1002 - Page 37

compromise component A, the attack vector 1806 passing through component B to reach component A. A threat emanating from component D also threatens to compromise the asset, the attack vector 1804 passing through component C and component B to reach component A.

[0145] The attack vector displays may be modified based on contemplated compensating controls. FIG. 19, for example, shows the diagram of FIG. 18 but modified to include six compensating controls (each labeled "CC"). These compensating controls may mitigate different threats emanating from different components. Compensating control 3 (CC3) may mitigate one or more threats from component D. CC2 may mitigate one or more threats from components C and/or D. CC5 and CC6 may each mitigate one or more threats from component E. CC4 may mitigate one or more threats from components F and/or G. CC1 may mitigate one or more threats from components B, C, D, E, F, and/or G. Using such a modeled attack vector diagram, the user may be able to alter compensating controls and make decisions about where in the system to deploy compensating controls. The user may determine, for example, that CC1 may mitigate many threats more efficiently than several compensating controls deployed elsewhere in the system. Or it may be determined that the asset is better protected by some other combination of compensating controls.

[0146] The elimination of any threat of course protects downstream components. Accordingly, for example, if CC3 eliminates or mitigates one threat emanating from component D, then the downstream effects of that threat to components C, B, and A are all mitigated. On the other hand, if CC1 eliminates or mitigates the same threat, it would only eliminate or mitigate the threat for component A, not for components B or C. Accordingly, there are security advantages to having compensating controls further upstream.

[0147] Changing the compensating controls in such a diagram could also change the other displays on other interfaces, for example the listing of top threats, listing of top ten most vulnerable components, other top ten listings, composition of threats by risk, composition of threats by status, composition of threats by source, composition of threats by other criteria, the overall threat model, the mitigations interface, the compensating control report, and so forth. This dynamic change of all interfaces allows the user to do "what if" analyses by adding/removing any compensating control and/or toggling any compensating control between ON/OFF states and then seeing how all the aforementioned displays/reports etc. change accordingly. The change may, for example, illustrate one or more threats as mitigated or closed instead of open or unmitigated, reduce the total number of open or unmitigated threats, reduce exposure levels of data elements, alter the percentages of threats by risk level (very

IriusRisk
Ex. 1002 - Page 38

high, high, medium, low, very low, etc.), alter the percentages of threats by status (open, closed, unmitigated, mitigated), or the reverse of all these, on any of the interfaces or reports of the system.

[0148] The term "asset" as it is used herein may refer to anything that has value to a user or organization, and therefore requires security measures to protect it from theft, manipulation, destruction, compromise, or other forms of abuse. Attackers are an active population of computer environment users and vary in the skills, toolsets, opportunities, financial backing, and other attributes required for a successful attack on an asset. Most security measures to protect a computer system-related asset may be categorized into one of two categories: (1) defensive and/or monitoring mechanisms to inhibit attacks in progress or mitigate the effects thereafter, and; (2) reducing the number of potential threats through which attacks may be initiated to begin with. The latter can in many cases be less costly overall, and the ability of the systems and methods described herein to test and retest various configurations, including modeling the use of compensating controls, allows users to focus efforts on this method.

[0149] In implementations threat modeling and/or attack simulation methods may include the following steps: (1) generally defining a threat model (including model type, outputs to be generated, and overall objectives for different stakeholders); (2) visually diagramming a system/process using database-stored components (including communication protocols and compensating controls); (3) classifying one or more data elements within a component (or associated with a component), and/or one or more components themselves, as assets; (4) identifying and enumerating potential threats to the assets based on the diagrammed components; (5) analyzing the identified threats (threat analysis); (6) identifying and analyzing potential attackers based on the threat analysis; (7) analyze the effectiveness of deployed or contemplated compensating controls to mitigate the potential threats; (8) analyze a threat model and attack surface as modified by compensating controls; and (9) measure and quantify the effectiveness of the model and/or method relative to the desired outputs and objectives previously determined. This is just one example of steps included in such methods, and other methods may exclude some of these steps, or include other steps, and so forth. In an implementation of the above steps, step (6) interrupts step (5), and in an alternative implementation step (5) is allowed to be completed without interruption. Furthermore, in implementations steps (7) and (8) may be iteratively bounced between during any given implementation, so that a user adds a compensating control then checks the outcome, toggles the configuration of the compensating control then checks the outcome, removes a

compensating control then checks the outcome, moves a compensating control to a new location then checks the outcome, etc., before moving to step (9).

[0150] Referring to step (1), different stakeholders may have different threat model output needs, so different model types can be built using system 100 and the methods described herein. For example, software application developers, operations teams, embedded system teams, IoT device teams, cloud deployment and architecture teams, ICS system or cyber-related system teams, and so forth, may create different threat model types. Architects may model threats to a residential structure, structural engineers may model threats to occupants, neighborhood planners may model supporting infrastructures needed for developments, and so forth. Home decorators may model occupant flow and so determine the most secure places for expensive home décor. In non-computing systems/process that are modeled, the protocols may represent things other than communication – for example in the home décor example the protocols may represent methods of moving an occupant between rooms. In implementations a user may select one of many model types to begin with (to populate the proper predefined components with which to diagram the model). This could be selecting, by example, from among examples such as the following: application threat model; operational threat model; cloud threat model; embedded threat model; IoT threat model; cyber-physical system threat model; and so forth. A user may still access all components in the diagram interface by searching for them but selecting the proper model may allow the most relevant components to populate at the top of the list.

[0151] Still referring to step (1), a user may in implementations also select role-based threat model outputs (such as with yes/no selectors) such as: secure coding requirements; operational checklists; executive metrics; executive reporting; security metrics; security reporting, and so forth, and these may also affect the position of components within the toolbox list on the diagram interface. In other implementations these selections (and the selections in the above paragraph) may only be used for record-keeping purposes. The input may also include the name of the model, the version, the name of the model creator, the date created, the update person, the update date, comments, desired objectives by development/operations/security/executives/business leaders, and others, and notes or toggle elements to indicate whether those objectives are met, in progress, or in some other state (and any or all of these features may be included on one or more of the already-described interfaces).

IriusRisk
Ex. 1002 - Page 40

[0152] A representative example of the aforementioned step (3) is illustrated in FIG. 20, which may involve the classification of assets which may be associated with, located nearby or relevant to, or utilized by diagrammed components. In implementations the asset classification for each asset includes: an asset class; an asset, and; an associated component. In implementations the asset classification may be used as an input for analyzing potential attackers by providing insight to possible purposes attackers may have in targeting the asset. As an example, in implementations of cybersecurity modeling the assets may be classified as: non-confidential data; confidential consumer data; confidential corporate data; system capabilities (such as electronic funds transfers); non-digital assets (such as infrastructure or IT-system controlled equipment, and so forth. In a residential security model the classification could be: lightweight valuables; heavy weight valuables; personal information; life and health of occupants; residential structure/utilities; home security (entry keys/codes), etc.

[0153] Step (5) has already been described to some extent previously with respect to FIG. 17 with the creation of an attack tree (a partial attack tree is shown in FIG. 17 – a full attack tree would include all assets as topmost nodes, and in implementations the attack tree may be similar or identical to the "threat tree" shown in FIG. 14 of the '021 Application). Referring to FIG. 17, the attack vector connects the asset with the component from which the threat emanates, and this association was determined by the previous classification step. In implementations the attack/threat tree may be simplified by only showing one topmost node (one asset) and only its associated components and threats. In implementations the attack tree only shows three levels – the topmost level being the assets, the middle level being components, and the bottom level being threats. This model can be simplified further to have only two levels – the topmost level includes asset/component pairings, and the bottom level shows the threats. In this simplified model the attack tree may show the asset identified as being nested within a component or otherwise attached or coupled with it (in implementations the topmost level would then show, for example, component 1 including asset 1 in level 1, component 2 including asset 1 in level 1, component 3 including asset 1 in level 1, etc., with level two (the bottom level) showing the threats coupled with the components). As a representative example, the asset could be jewelry, the components could be a front door, a back door, and a window, and a threat to the front door could be a bump-key, and since an attacker may use a bump-key to enter the front door, the bump-key is a threat to the asset.

[0154] Step (5) includes a second part, which is analyzing the identified threats. In implementations this involves various stages. One stage includes creating a profile for each

IriusRisk
Ex. 1002 - Page 41

threat. This may include utilizing real-world threat intelligence to provide attributes to said threats, and the attributes may include defined technical means to exploit the threat. A numerical value may be assigned to this means attribute (a high value indicating a higher level of technical means required to exploit the threat. This means attribute may also include the requisite skill, toolset, social network, financial backing, etc. the attacking entity would need to utilize the threat and traverse the corresponding attack vector. As examples, the technical means for the aforementioned bump-key against the front door is relatively low, whereas the technical means to alternatively burrow through the home's foundation to bypass other security controls is relatively high.

[0155] Another threat attribute is the access required to carry out the attack. A number is assigned to the access attribute, with a high number indicating a higher required level of access. A low access attribute may for example be attached to a lower level employee's alphanumeric password (which may be known by an immediate supervisor), while a high access attribute may be attached to biometric signatures for key personnel (which may require high access levels such as a privileged system administrator).

[0156] Another threat attribute is a defensive distance, or number of security controls and their effectiveness, between a threat and the asset. A high numerical value indicates a greater number of effective security measures in between. Another threat attribute is the potential impact if the threat is realized – a high value indicates a high technological or business impact in the event the threat is realized.

[0157] The second stage involves analyzing the attacker population. Two methods are disclosed herein. In one method (the preferred method), potential attackers are analyzed based on the identified threats and identified associated assets. The other method allows users to adopt existing threat intelligence about known attackers who have utilized existing threats (for example data of a 30% chance of a home invasion in any given year compared with a 1 in 20 million chance of a terrorist attack at the home). In the preferred method objective attributes and characteristics of attackers are provided even where such intelligence is not known.

[0158] The third stage involves determining a likelihood of a threat being realized. When the above preferred method is utilized this includes three sub steps of determining a subset of the attacker population with the sufficient means, sufficient motivation, and sufficient opportunity to exploit the threat, to determine the probability of an attacker subset coming from the attacker population (this discards attackers who would be unable to exploit the threat for whatever reason) (if the other method is used the statistically-provided probability is used). A

determination is then made of whether the asset is attractive to the attacker, if not the likelihood of threat exploit is set to a minimum value, but if it is attractive the likelihood of threat exploit is the product of an attacker coming from the attacker population and the level of attraction. The next step calculates the product of the impact should the identified threats be realized and the likelihood of the threats being realized for each threat. Each threat is then prioritized according to some scheme (e.g., critical, high, moderate, etc.).

[0159] Step (6) involves analyzing attackers using means, motive, and opportunity (as identified above, in the preferred embodiment this occurs partway through step (5) – but if real world intelligence is used this step is obviated and skipped). If the attacker lacks either sufficient means, or sufficient motivation, or sufficient opportunity to conduct an attack, the attacker is disregarded, whereas if the attacker has all three the attacker is included in the organizational attacker population. These determinations could be done, for example, by any known or discovered method to assign a meaningful and relevant value to each category, and if the value of the attacker's technical means meets or exceeds the above set technical means value (and similarly for a motivation value compared with the above identified access level value and an opportunity value compared with the above identified defensive distance value) then the attacker is included in the organizational attacker population, otherwise not.

[0160] Step (7) has previously been described with respect to FIG. 5.

[0161] Step (8) includes reviewing the threat report after all of the above steps have been done. Step (9) involves comparing the outputs obtained from the threat model and process relative to the desired outputs and desired objectives indicated in step (1). In implementations steps (1) and (9) may be done through system interfaces. In other implementations they may simply be done manually. In still other implementations they may both be excluded.

[0162] Referring now to FIG. 21, a representative example of attack simulation methods is shown. FIG. 21 shows the methods illustrated using the diagram interface 1000, though in other implementations the methods may be implemented using a separate interface that takes the components diagrammed in any given threat model and displays them in a separate interface just for the purpose of illustrating attack vectors. In FIG. 21 it is to be understood that the user has previously identified, through the database, the ONE DRIVE element as an asset. In other implementations the asset could be linked to the ONE DRIVE component but could be displayed as its own separate component – for example a "client info database file" shown next to or within the ONE DRIVE component (though in such a case the "asset" itself would be a

IriusRisk
Ex. 1002 - Page 43

"component" as that term is used herein). In the shown implementation the ONE DRIVE component itself is identified as the asset.

[0163] A number of dashed protocol lines are shown in FIG. 21. These are shown in this fashion after the user has selected the ONE DRIVE component and made a selection to show all threats that could compromise the component. For example, this could be done in implementations by right-clicking on the ONE DRIVE component and selecting a selector such as "Show Attack Paths," though the attack paths may be shown using other methods that the practitioner of ordinary skill in the art may determine. Once the selection is made, all threats reaching the ONE DRIVE component are highlighted in some manner. In the representative example they are highlighted by showing dashed and "moving" arrows, meaning an animation which shows the dashed portions of the lines as moving in one direction or another. In some implementations this animation could show the dashes moving in the direction towards the asset, which may assist the user to trace attack paths back to the outermost threats. In other implementations the arrows could be reoriented when the attack paths are shown so that the arrow tips all point downstream towards the asset, which may also assist the user to trace attack paths back to outermost threats. In implementations the attack paths are shown in a different color and/or using a greater line thickness than other protocol lines. In implementations the attack paths are shown in bright red and using thick lines while non-compromised paths are shown in thin black lines.

[0164] It can be seen that there are threats associated with the WIFI ACCESS POINT and with the SHARE POINT components, and that these threats reach the ONE DRIVE component through OFFICE 2010 and IE11 components. There are a number of potential compensating controls in the diagrammed system which may be utilized to mitigate these threats. For example, the laptop component group is shown including the MCAFEE HIPS, MCAFEE AV VSE, BIT LOCKER, and MCAFEE DLP compensating controls, and there is also an SSO compensating control diagrammed between the IE11 and ONE DRIVE components. All compensating controls in this diagram are shown as having an icon of a padlock on them so that the user can easily view the compensating controls to modify the threat model and the attack simulation (the compensating controls in implementations do not introduce any new threats but only mitigate threats).

[0165] Referring now to FIG. 22, in this screen the user has toggled the MCAFEE AV VSE and SSO elements to an ON configuration. In implementations this is done by right-clicking each component and selecting "BLOCK" from a menu, which then changes all attack paths that are mitigatable by the selected compensating control to a mitigated display state (in some

IriusRisk
Ex. 1002 - Page 44

implementations the user may be able to select which attack paths are mitigatable from this interface). In FIG. 22 these compensating controls are highlighted by outlining them in a thicker line to show that they are turned on (though this display may be excluded in other implementations). The attack paths that are mitigated by the compensating controls that have been toggled to an ON state (or to a blocking state) are now shown as solid lines again, while those that are not mitigated are still shown in dashed format. In implementations the mitigated attack paths are shown in a thick green line. The user may reselect the compensating controls and select unblock to revert the threat model to the previous state. The user may also delete any given compensating control from the diagram altogether, and this will have the effect of changing any attack paths that were mitigated by that compensating control to an unmitigated display state.

[0166] While the mitigated display state for attack paths are described herein as green and bold and the unmitigated display state for attack paths are described as dashed, red, bold and animated, the practitioner of ordinary skill in the art could select other visual techniques to differentiate attack paths from non-attack paths and mitigated attack paths from non-mitigated attack paths. When a user toggles a compensating control to an ON state this may have the effect of dynamically changing the threat report, the compensating control report, the mitigations interface, and so forth by toggling relevant threats to mitigated (and toggling the same compensating control to the OFF state or removing it altogether from the diagram may have the opposite effect). In implementations one or more of the user interfaces (such as those of FIGS. 21-22) may allow user selection of certain threat sources and deselection of others, so as to only show attack paths of threats associated with a selected subset of the diagrammed components. The threats are shown in FIGS. 21-22 as emanating from outermost components (and the system/method may default to checking for threats at all outermost nodes), but in implementations they could emanate from internal sources (such as a USB port in an internal system administrator computer being a threat source for a virus). An attacker could be visually displayed at each threatened component, for example an "attacker" component could be placed on the diagram coupled with the WIFI ACCESS POINT and SHAREPOINT components to indicate the presence of an attacker. In other implementations the attacker need not be displayed in this way, but the user may understand that an attacker is reaching for the asset through the end points from where the threats are shown emanating on the diagram. From these end point components, or from an actual "attacker" component, the user may edit an attack profile (such as to indicate along a spectrum from a low funded individual hacker all the

IriusRisk
Ex. 1002 - Page 45

way to a state sponsored team of skilled and well-funded hackers) and this may modify the attack paths. For example, if the latter profile is identified then certain compensating controls may not mitigate the threats (for example the SSO element may not be effective, and another compensating control would be needed instead), and accordingly when the profile is changed one or more of the attack paths may change to a mitigated or unmitigated display state in response. The identification of hacker type may be done, for example, through dropdown lists, checkmark items, and so forth that had been previously set through the database by the user, and the ability of a compensating control to mitigate threats by lower-end attackers but not higher-end attackers may also be defined/set through the database in the input interfaces previously described (though using selectors not shown—but which the practitioner of ordinary skill in the art will know how to implement).

[0167] In implementations another interface may allow the user to get a list of all threats which may compromise the asset and all attack paths associated with those threats. The asset has been described above as being anything of value to the user, it could for example be a database, a financial asset, a trade secret (or files related to trade secrets), and so forth.

[0168] One more comment is made here with respect to threat model chaining. The user may select a sub-component or component group (for example the user could right click the WINDOWS 7 component in FIG. 22) and select an "open in new window" selector to open a new diagram interface showing the diagram just for that component group. The user may then view just the threats associated with the component group, using previously identified methods, and/or may identify assets within the component group (for example OUTLOOK 2010) to show attack paths related just to this component. In implementations a user may right click or otherwise select a component group on an existing threat model by selecting a box outlining the component group and then select a "show threats" or other menu item to treat the entire component group as an asset and to show all attack paths thereto.

[0169] In implementations the designation of a component as an asset is done as simply as clicking on any component and selecting a "show threats" or similar menu item from a list. In other implementations a component may be designated as an asset by another mechanism and/or using another interface. The asset may be selected and its features edited, as well, as detailed to some extent above (for example some of the asset classification may be done here, though this may alternatively be done using one or more other interfaces). The attack simulation systems and methods herein allow a user to obtain a thorough understanding of

IriusRisk
Ex. 1002 - Page 46

threats and risks to multi-faceted systems/processes, and the impact to relevant assets, should potential threats be realized.

[0170] Although the examples of computing system models have been described in detail, system 100 may be utilized to model other threats. For example, referring to a disease epidemic threat model, the threats may be sources or ways the disease may spread, the security requirements may be methods or elements to reduce the effects of those sources, such as sterilizing surgery instruments, washing hands, using air filters, breathing masks, gloves, and the components may be steps or interactions that occur with respect to persons who may be infected, such as admitting them, dealing with infected clothing, performing medical operations on them, etc. In such a scenario the threat model may have no components, threats, or security requirements that relate directly to computing devices or systems, though the threat may be modeled on system 100 which utilizes computing devices. The same may be said for the aforementioned threat model of commuting to work. Numerous other threat models, involving any of myriad types of threats, could be modeled using system 100. Other types of threat models could include: modeling physical and technology threats to airplanes, smart homes, smart TVs, an electric grid, and so forth.

[0171] Accordingly, the system and interfaces described herein allow a user to model a variety of systems, applications, and processes to determine threats and to mitigate those threats. The user may, at any time, delete any security requirement component (which may be a compensating control component) from a diagram and then generate the threat report again to revert to the threat statuses prior to mitigation by any particular component. This is one way in which step 518 of FIG. 5 may be accomplished, since removing the mitigating component will revert the mitigated threats back to the open status (except for threats which are also mitigated by another mitigating component). In implementations there may be other ways to "roll back" the threat report to an earlier status, such as with one or more selectors on the threat report itself to revert to an earlier configuration, or one or more "back" selectors on the diagram interface, and so forth.

[0172] Utilizing the methods and systems described herein an organization or individual can quickly and efficiently model deployed or contemplated systems, applications and/or processes and assess relevant threats to determine a mitigation strategy. The user may assess new threats to the entire attack surface of a system, application or process as the new threats arise and accordingly develop modified mitigation strategies with minimal disruption to existing operations. Additionally, the methods and systems described herein allow users to study the

IriusRisk
Ex. 1002 - Page 47

ability of a compensating control to block new or existing threats without penetration testing (pen-testing), which can be resource-intensive, slow, and not as thorough as desired. The systems and methods also allow assessment of future compensating controls which have not yet been implemented (and accordingly where penetration testing is not yet an option). Any of the threat modeling described herein may utilize existing threat modeling methodologies including STRIDE, P.A.S.T.A., TRIKE, OCTAVE, and/or VAST methodologies, which may include modeling the system, application or process as a data flow diagram (DFD) or a process flow diagram (PFD). The methods and systems herein integrate well with agile and DevOps development environments, and allow a threat modeling process which is simple enough to allow non-security experts to participate in the threat modeling process and to benefit from the outputs of the system/method.

[0173] In implementations a compensating control record in the database may include a definition or details which include: an articulation of the rigor and intent of a prescribed security requirement which the compensating control is meant to satisfy; a defensive level of the compensating control; security and defensive capabilities of the compensating control, and; additional risk imposed by not adhering to the prescribed security requirement.

[0174] Systems and methods described herein may have other features that have not been described in detail. Users may be able to access the interfaces of the system through a web page, as described, and may be able to register using user accounts, with login credentials, functionality for retrieving lost passwords etc. Individual users may be able to import custom libraries and other elements into the database including custom components, threats, threat agents, test cases, security requirements, code snippets, code reviews, data elements, roles (developer, customer, user, admin, etc.), widgets, component types, attributes, templates, threat models, and the like. In implementations widgets may be a subset of components and may include reusable modular components that may be included in some components. Some examples of widgets may be, by non-limiting example: forms, cookies, banners, embedded video, RSS feeds, SMS functionality, SILVERLIGHT, secure hardware token, file upload, XML parser, and so forth. Widgets may be associated with threats through the database, using an interface similar to interface 600 (but accessible by selecting “widgets” from interface 700), and may be added to any diagram in the same way that components are added, which may include defining communication protocols between the widget and other components/widgets. Widgets may in implementations comprise identifying information about a component, user

IriusRisk
Ex. 1002 - Page 48

roles which can interact with a component, and the means by which the component receives, transmits, or stores data elements.

[0175] In implementations a software installation on one or more computing devices to implement the systems and methods may include storing in the database default libraries, such as over 650 threats, and a number of standard or default definitions for components, default risk levels to choose from, though the user may set up custom threats and associations through the database (which essentially sets up custom risk algorithms), custom risk levels to choose from, and so forth, as described above. The ability of the user to diagram a system, application or process visually allows for non-security experts to analyze threats using a threat model. In implementations, previously generated diagrams may be able to be imported into the system using a PDF, XML, TMT, JSON, PNG, JPEG, or VISIO drawing (VSD, VSDX).

[0176] In implementations the system is configured to interface with third party software applications and libraries (including software as a service (SAAS)) services, by communicating with the third party software/library using its application programming interface (API). This may be used, for example, to store standards from standards-issuing bodies in the database and to associate them with security requirements and/or compensating controls through the database. This may assist a user to determine when a component added to a diagrammed threat model would qualify as a compensating control as defined by the standards issuing body. The system may be said to include a "contextual threat engine," which is the underlying database relations allowing the system to dynamically update the threat model each time a component is mapped onto the canvas, or connected with another component using a communication protocol, or when a protocol is changed or deleted, or a component removed, etc. The systems and methods described herein facilitate comprehensive enterprise-level attack surface analysis. Different users may use different interfaces within an organization in some implementations. Information technology (IT) professionals may define the actors, threats, components, etc. specific to an organization, non-IT professionals may map out systems using the canvas, executives may review the high-level threat details included in the home page or threat model, and so forth (though in other organizations any user may use any interface).

[0177] In implementations one or more user inputs may be used using one or more user interfaces to import a threat model from a third party software or database into the system. When this is done an administrator may map components or elements of the imported threat model to stored components of the system so that the relevant threats and etc. will be included in the imported threat model. Along these lines, it is pointed out here that both end users and

IriusRisk
Ex. 1002 - Page 49

administrators are "users" of the system as that term is used herein. Accordingly, when the claims refer to "one or more user inputs" these inputs may be carried out by an end user and/or they may be carried out by an administrator, such as an administrator maintaining a cloud database which several different end users are accessing to implement different deployments of the method/system.

[0178] Referring now to FIGS. 23-29, various representative examples of systems and methods for automated threat model generation from third party diagram files will be described. As far as systems for automated threat model generation from third party diagram files, the system 100 previously described may be used for this, with some additional user interfaces that will be described shortly. FIG. 23 shows a dashboard interface (interface) 2300 which may be displayed on one of the computing devices of system 100 (such as an end-user computer) and may allow a user to view information and/or navigate to different interfaces or windows as desired by the user. For example, the dashboard interface shows a THREAT TRENDS area, a THREAT TRACEABILITY MATRIX area, a TOP 10 THREATS area, and a THREATS BY RISK area. These may show information and/or may be selected to navigate to an interface/window with more information, and are only representative examples that will not be described in great detail. A filter icon is also seen at the top left which may allow a user to filter the items shown on the screen in various ways. At the top right a threat model selector 2302 may be selected to navigate to a threat model interface (interface) 2400, a representative example of which is shown in FIG. 24. Interface 2300 could be a home page for the software.

[0179] Interface 2400 is seen to list threat models by name and to include a version number (1.1, 1.0, etc.). These may be all threat models that have been created by a specific organization or end user of the system and methods. Also listed are a percentage (indicating a completion percentage of the threat model), an overall threat risk of the model (Medium, Very High, etc.) as assigned by a user using interface 2500 (described later), who created or last worked on the model, and a trash icon for deleting the model. A filter selector at the top left allows a user to apply various filters to search for a specific threat model. On the right are shown THREAT TRACEABILITY MATRIX and TASKS areas which show further information and which may operate as selectors to navigate to further information. The user is seen to have selected the END POINT SEC. threat model, so that it is highlighted. Additional selectors include a new selector (selector) 2402 which may be used to begin a new threat model, an edit selector which may be used to edit a threat model or the aforementioned descriptive information, a number indicator (48) indicating the total number of threat models in the one or more databases (or the total number

response to the filter request), and other selectors. In implementations a user could double-click on any given threat model to navigate to another interface, such as the diagram interface of FIG. 21 or the threat report interface of FIG. 13.

[0180] In the representative example the user may select the new selector 2402 to bring up import interface (interface) 2500 as shown in FIG. 25. This interface includes a number of fields for the user to fill out, including a threat model name (here the user has input "Test Proj 1"), a version number (the user has input "1.0"), and a selector to select a type of threat model (here the user has selected "AWS Cloud App" from a dropdown menu which was previously populated in the database, in other implementations other options could include, by non-limiting example, AZURE Cloud Application, Mobile Application, ICS Application, Web Application, Web Service, Computing Network, Banking Network, Communications Network, and so forth). Interface 2500 also includes a field for the user to add tags or labels to the model, here the user has added the AWS tag to this model. These tags may be used for more easily finding this threat model later such as when using filters on the aforementioned interfaces. Other representative examples of labels could be, by non-limiting examples: Hardware, Firmware, JAVA, JIRA, INI, Certificates, PII, Trial, IoT, BLUETOOTH, Banking, SSO, Privacy, Input Validation, Password Policy, CAPEC, Log Files, OWASP, Embedded System, MIDAS, CAPEC-100, CAPEC-14, CAPEC-24, CAPEC-433, CAPEC-45, CAPEC-47, CAPEC-8, CAPEC-9, CAPEC-101, WASC-36, OWASP A1-Injection, OWASP A2 – Broken Authentication and Session Management, and so forth.

[0181] A dropdown menu also allows a user to select an initial threat level for the model, here the user has selected a "High" threat level. A checkbox indicator allows the user to select whether the model is of a system, application, etc. external or internal to the organization. Any of the text fields of interface 2500 (as with all other interfaces of the systems and methods described herein) may include predictive text and/or most-used suggestions so that text often used or predicted by the system as being likely to be used in this instance is suggested to the user, with the user being able to select the text to insert it into the field instead of typing it out in full.

[0182] On the left of interface 2500 are selectors. An EMPTY selector allows the user to begin a new threat model from scratch. The TEMPLATE selector allows a user to begin a new threat model by starting with one or more templates previously stored in the one or more databases. The WIZARD selector allows the user to answer questions and indicate some parameters to begin a threat model. The IMPORT selector allows a user to use a preexisting data file from a

IriusRisk
Ex. 1002 - Page 51

third party software program to generate a threat model. By non-limiting example, in implementations a user may import a MICROSOFT VISIO file in VSD and/or VSDX format, a file in PDF format, an XML file, a GLIFFY file, TMT file, JSON file, PNG file, JPEG file, or a number of other file formats. In the representative example shown in FIG. 25, the system is configured only to import VSD and VSDX file formats. The VISIO file imported by the user would be a VISIO diagram file on which the user has previously diagrammed a system, application, or process and now wishes to generate a threat model using that same diagram.

[0183] On the preexisting diagram the user will have used visual diagram components (hereinafter "third party diagram components" or "VISIO diagram components" in the case of a VISIO file) to diagram the system, application or process. These third party diagram components may have default titles for various components. For example, VISIO may have a component titled "INTERNET GATEWAY" which the user could add to a VISIO diagram. However, the user may also add a custom name (user-defined name) to a component in the third party diagramming software. Or, in implementations, some software may allow a user to replace a default title with a user-defined name for any given component. In VISIO a user may select an element or component that has been placed on a diagram and type text to give the component a user-defined name that is then displayed on the diagram and associated with the component. Either or both of the user-defined name and the default title may be used by the system to determine how to generate a threat model using the third party diagram.

[0184] Referring to FIGS. 28 and 29, interfaces 2800 and 2900 show comma separated value (CSV) files (mapping files) that may be used by the system to correlate the third party diagram components with threat model components previously stored in the one or more databases. Interface 2800 shows a CSV file which includes a table listing, for each listed item, a name given by an administrator, a value (which is the default title assigned to a component in the third party diagramming software), a comment (if any), and a global unique identifier (GUID) which is associated with a specific threat model component through the one or more databases. In this way any component of the third party diagramming software may be mapped to (correlated with) a corresponding threat model component stored in the one or more databases. So, for example, if the user's VISIO file includes a component that has the default title in VISIO of INTERNET GATEWAY, when the VSD or VSDX file is imported using interface 2500 the system 100 may include a corresponding threat model component (i.e., an "INTERNET GATEWAY" threat model component previously defined in the system and stored in the database(s)), so that in a threat model diagrammed on the diagram interface 1000 after importing the file an

IriusRisk
Ex. 1002 - Page 52

INTERNET GATEWAY threat model component is located exactly or approximately where the INTERNET GATEWAY third party diagram component was located on the VISIO diagram. Additionally, the INTERNET GATEWAY threat model component would have previously been configured to match the details/properties of the INTERNET GATEWAY third party diagram component. The step of defining component mapping, as described above, is representatively illustrated by step 3002 of flowchart 3000 in FIG. 30. This may be done by an administrator or, in implementations, by an end user. FIG. 30 also representatively illustrates step 3004 of selecting a file for import, which will be described hereafter with respect to the import interface 2500 of FIGS. 25-27.

[0185] Interface 2900, as indicated, illustrates a CSV mapping file that based on default titles within the third party software, and the table has similar elements except the "value" column is populated with potential user-defined names. In other implementations the tables may be organized differently, for example the "name" columns of interfaces 2800 and 2900 could be the default titles and potential user-defined names, respectively, instead of the "value" column. Interface 2900 illustrates a CSV mapping file that is based on user-defined names for elements. Referring to FIG. 29, if a user defines an element within the third party software diagram as an AIRPLANE or an ATM the import function can pick the appropriate threat model component, either an AIRPLANE or an ATM threat model component, for the generated threat model. In implementations the system may be configured to use one of these first and the other as a backup, for example the system may be configured to first use the user-defined name, so that if a match is found in the first mapping file (interface 2900) that determines the threat model component that is used, but if a match is not found then the default title of the component in the third party software is checked against the second mapping file (interface 2800), and if a match is found there that determines the threat model component that is used. In implementations this could be reversed, and in implementations a single file could be used instead of two files (for example all user-defined names and default titles included in one list and the system simply selecting the first match that is found). In other implementations the system may find a match in the first mapping file but may also check the second mapping file to verify that the match is correct. The steps of checking components of the imported third party file against the mapping files is representatively illustrated by steps 3006 and 3008 of flowchart 3000 in FIG. 30.

[0186] In implementations if no match is found a placeholder or default component could be used, and shown on the diagram interface 1000 in a way that indicates to the user that a match

IriusRisk
Ex. 1002 - Page 53

was not found for the third party diagram component. As an example, FIG. 21 shows an UNKNOWN component coupled with the IRON PORT component, indicating to the user that the system could not find a match for the user-defined name or the default title of that component during the import. This could be termed "matching" the third party diagram component with the placeholder component. In other implementations, if a matching element is not found then the third party component could be ignored so that, for instance, a blank area is left in the threat model diagram where the component would be. The step of using a placeholder component is representatively illustrated by step 3010 of flowchart 3000 in FIG. 30.

[0187] File formats other than CSV files could be used for the mapping file(s). In implementations one or more tables in the database(s) could include the same information and could be used for mapping. In the implementation in which separate files, not in the database(s) are used, they could be stored on the one or more computing devices that comprise the system 100 (including user computers, servers, and so forth). In implementations in which one or more tables of the database are used for the mapping file(s), each table is, for the purposes of this disclosure, a "file."

[0188] In FIG. 26 it is seen that the user has used the BROWSE selector to navigate to and select a desktop VSDX file and that a VALIDATE selector is now selectable. Upon selecting the VALIDATE selector the interface 2500 then either indicates that the file validation was successful, as in FIG. 27, or that the validation was unsuccessful. This validation step may include an actual import of the VSD or VSDX data file to the one or more databases or to memory of one of the one or more computing devices. It may also or alternatively include a simple check that the file is one of the supported formats. In the implementation shown in FIG. 27 the validation function simply validates that the file was successfully uploaded to the database(s) or one or more of the computing devices/servers of the system. The step of selecting a file for import (a VISIO file or other type of file) is representatively illustrated by step 3004 of flowchart 3000 in FIG. 30.

[0189] While the terminology "import" is used herein, the system could function in a variety of ways. Naturally if the third party file itself is uploaded to computing devices or databases of the system this is an "import" in the natural sense of the word, but in other implementations the VALIDATE or SUBMIT selectors could simply initiate a reading of the file locally on the computing device that is being used by the user to use the interface 2500, and then only uploading to the database(s) or other computing devices (including server(s)) of the system 100 mapping data that is used to create the threat model, without uploading the actual underlying

third party software file. Either way, whether the third party file is only stored locally on a computing device, or is uploaded to other devices or database(s) of the system, the file is at some point read to do the mapping of the third party diagram components to threat model components by referencing the mapping file(s). This may be done using executables and code which will be understood by the practitioner of ordinary skill in the art without further explanation herein, so that a detailed discussion of the code and/or executable(s) used to read the third party file and determine the threat model components correlated in the mapping file(s) with the third party diagram components is not given herein.

[0190] As has been previously described, in the database(s) the communication protocols may be stored as threat model components. In the third party diagram the lines connecting elements may or may not include communication protocol information (they may just be lines). If no communication protocol information is included these connectors may simply be represented as lines in the generated threat model, so that the user could later select the communication protocol(s) for any given connector. In some cases though a user may have given a user-defined name which indicates a communication protocol (such as HTTPS), or the component in the third party software may have a default title indicative of a communication protocol, and the system 100 may use this to represent the connector as a specific communication protocol once the threat model is generated.

[0191] Referring back to FIG. 27, at any point the user may select the CANCEL selector to cancel the creation of a threat model. After file validation the user may select the SUBMIT selector to generate a threat model based on the third party diagram in the third party software file. Again, the specific details of how the system performs this are not included because they may be accomplished in a variety of ways, but in the implementation shown selecting the SUBMIT selector initiates a process of determining, based on the uploaded file, which threat model components (including communication protocols if possible) correlate with the third party diagram components, and thereby generating a threat model including the correlated components. The step of generating a threat model is representatively illustrated by step 3012 of flowchart 3000 of FIG. 30, and may include displaying interfaces 1000, 1100, and 1200 previously described, as well as other interfaces/elements. Generating the threat model may include using the system to determine threats, threat sources, threat risk levels, threat statuses, security requirements, compensating controls, threat mitigations, and so forth—associated with the system/method or the like that was diagrammed in the third party software and imported to generate the threat model—and displaying these to the end user. After selecting the SUBMIT

IriusRisk
Ex. 1002 - Page 55

selector the diagram interface 1000 may automatically be displayed showing the visual diagram of the threat model, as in FIG. 21. From this interface the user may edit the threat model using any of the methods previously described. At any point the user may navigate to the threat report interface 1300, as described previously, to see the threat report showing the threats associated with the generated threat model. The threat report interface and its functionality, and the threat report generally, are described in detail above.

[0192] In some implementations of system and methods for automated threat model generation from third party diagram files, an admin or the like may provide default database data, such as one or more libraries of threat model components, threats, mapping files, software for local installation to view and use the interfaces described above, and the like, so that the importing and automatic threat model generation at the user's end is very simple. In other implementations (or as determined by the end user based on their expertise) the end user could modify the threat model components and threats, create new threat model components and threats, modify the mapping files, and so forth. For example the end user could open a standalone mapping file such as the above-described CSV files and make edits to it, or replace it with another CSV file, for the mapping and correlation of third party diagram components with threat model components. In implementations in which the mapping files are one or more tables stored in the database(s) there may be another interface (not shown in the drawings) at which the user may edit mapping tables.

[0193] In many implementations the diagram of the third party software may have been used to diagram a computing network or environment, so that the resulting threat model will model a computing network. By computing network is meant any networked system of components which includes computing devices in the general sense – for example an ATM is a computing device so that a system including an ATM would be a computing network. Nevertheless, in implementations the automated threat modeling systems and methods may be used to model processes, systems, applications, and a variety of other things as has been described previously.

[0194] The mapping files may include a large number of third party diagram components (for example it could include likely user-defined names and default titles for the entire component library of the third party software program), and any given data file of the third party software application that is imported would normally include only a subset of those components, so that the system is only finding the threat model components which correlate with the subset to generate the threat model. The data file of the third party software would also generally define relationships between elements, for example at least lines connecting one element or

IriusRisk
Ex. 1002 - Page 56

component with another. As previously described, these may be more detailed (such as including communication protocol information) in the third party data file, or they may be less detailed, only being lines or connectors. Either way, when two elements are connected in the third party data file this is a "relationship" between those elements as used herein, as it generally indicates some connection or coupling of the elements, so that the third party data file "defines" relationships between the elements. Since the data file may define relationships in more detail, i.e., HTTP, HTTPS, or other communication protocols, in those cases the system and method can determine threat model components that are correlated with the third party diagram relationship definitions (such as determining that an HTTP communication protocol threat model component correlates in the mapping file with a connector line in the third party diagram for which the user has given a user-defined name of "HTTP"). The third party diagram data file, such as VSD or VSDX file, will generally include data related to each component, such as location of the component, which other components it is connected to, the default title of the component, any user-defined name for the component, etc. The system 100 and methods described herein use that information to create the threat model.

[0195] Since the one or more mapping files may be stored on one or more computing devices communicatively coupled with other elements of the system, or may be stored in the database(s) directly, in either case they are communicatively coupled with the database(s). They may be accessed using one or more of the servers or other computing devices of the system to perform the functions described herein.

[0196] As has been described with respect to interface 2500, in implementations a single interface initiates, in response to user input, both reading of the third party data file by one or more computing devices of the system and, additionally, a determination of which threat model components are correlated (in the one or more mapping files) with the subset of third party diagram components that have been read. For example the VALIDATE selector may initiate reading of the file and the SUBMIT selector may initiate the correlation determination. But in other implementations a single user input on a single user interface could do both, for example the VALIDATE selector could simply upload the file and the SUBMIT selector could initiate both reading of the file and the correlation determinations (which could be done sequentially or to some extent simultaneously). In other implementations two or more selectors on separate interfaces could be used, but in the claims where "an import interface" is described and, thereafter, an interface for initiating the correlation determination, in implementations these

IriusRisk
Ex. 1002 - Page 57

may be the same interface (as with the example in the drawings, where the import interface initiates both functions in response to user input).

[0197] In most if not all cases, before the threat model is generated by the system 100, the third party data file does not include a threat model as that term is used herein. While the third party software may allow for the creation of a diagram of a system, application, or process, the systems and methods described herein allow the user to then generate, using that diagram, a threat model to almost instantaneously assess threats to the system, application or process using the threat report.

[0198] Although a mapping file specifically correlating threat model components with VISIO diagram components is not shown in the drawings, the practitioner of ordinary skill in the art will know how to create such a mapping file without specific illustration simply by reference to interfaces 2800 and 2900 and the above description. Where the above description applies generally to third party diagram components, subsets of third party diagram components, and the like, it may also apply to specific software components (for example VISIO diagram components, PDF diagram components, subsets of VISIO diagram components, subsets of PDF diagram components, XML diagram components, subsets of XML diagram components, TMT diagram components, subsets of TMT diagram components, JSON diagram components, subsets of JSON diagram components, PNG diagram components, subsets of PNG diagram components, JPEG diagram components, subsets of JPEG diagram components, and so forth.

[0199] In some implementations the mapping of the mapping files could be one-to-one, for example each potential third party diagram component has a single matching element in the list. In other implementations the matching could be multiple to one – for example a FILE SERVER or a SQL SERVER in a third party diagram both being matched to a single SERVER element with the mapping file. But in cases where there are different properties between different potential third party diagram components – for example a SQL SERVER having different properties than a generic FILE SERVER, the mapping file(s) will generally include different threat model components and map to the specific language, so that for instance a FILE SERVER in the third party diagram would be mapped to a FILE SERVER threat model component, a SQL SERVER in the third party diagram would be mapped to a SQL SERVER threat model component, and so forth. In instances where an element is just called a SERVER in the third party diagram it may be mapped to either a separate SERVER threat model component or the FILE SERVER component, etc. The mapping files accordingly allow flexibility in how a user may apply correlations between third party diagram components and threat model components.

IriusRisk
Ex. 1002 - Page 58

[0200] In implementations the systems and methods for automated threat model generation from third party diagram files may include machine learning to learn, through repeated user interaction, for example, what users usually change a specific third party diagram component to in the generated threat model. For example, if a component in a VISIO diagram has no match in the mapping file(s), so that a placeholder is used, but users generally change the placeholder to a ROUTER threat model component, the system may automatically update the mapping file(s) so that the VISIO component is mapped to the ROUTER threat model component. Alternatively or additionally, the placeholder component may, when hovered over or clicked on by a user, show suggestions which may be dictated by machine learning. For example if users in the past changed the third party diagram component to one of five threat model components, those five components may be suggested to the user. The suggestions could be listed in order of their frequency of use/selection for that component. The machine learning aspect may allow some of these features to occur without administrator input as the system learns from user behavior.

[0201] Similarly, if the system does map a specific third party diagram component to a specific threat model component, and users generally change the component to a different threat model component once the threat model is generated, the system may through machine learning suggest that the component may need to be changed (such as by a color or flag near the component or some other audio or visual notification) or, if enough or most users change the component the system may automatically update the mapping file(s) to reflect the more accurate mapping. Additionally, the system may learn how users increase the specificity of components. For example, if in VISIO users often use a generic DATABASE component, but when that component is coupled with a specific other component then in the generated threat model users generally select the more specific SQL DATABASE, the machine learning may allow the system to suggest that the generic DATABASE in the generated threat model be changed to SQL DATABASE, or if enough or most users make the change the system may update the mapping file(s) so that, when the generic DATABASE component in the third party diagramming software is next to the indicated component, it is mapped as a SQL SERVER. In such implementations the mapping file(s) would be more complex than what is shown in the drawings, for instance IF/THEN logic may need to be used so that if the third party diagram component is a SERVER then in the mapping file a SERVER listing may be found and then the system may determine if the server is coupled with an indicated component, and if so to represent it as a SQL SERVER threat model component in the generated threat model, otherwise to represent it as a SERVER threat model component in the generated threat model.

IriusRisk
Ex. 1002 - Page 59

[0202] In implementations of systems and methods for automated threat model generation from third party diagram files the method may allow a user to check some of the mapping in-situ before the threat model is generated, though in other implementations the threat model is generated without any in-situ checking and the user may make any needed modifications thereafter. In implementations there may be a one-to-many mapping so that, for example, one third party diagram component could be mapped to multiple threat model components and the decision of which threat model component is selected may depend on any user-defined names, any default titles, and the relationships between the various third party diagram components (such as described to some extent above in the SERVER vs. SQL SERVER example). In implementations the systems and methods may be able to take any diagram, in any file format type, and convert it into a threat model.

[0203] In places where the description above refers to specific embodiments of systems and methods for automated threat model generation from third party diagram files, one or more or many modifications may be made without departing from the spirit and scope thereof. Details of any specific embodiment/implementation described herein may, wherever possible, be applied to any other specific implementation/embodiment described herein.

IriusRisk
Ex. 1002 - Page 60

CLAIMS

What is claimed is:

1. A threat modeling method, comprising:

providing one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

providing one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components"); and

in response to receiving one or more user inputs, using one or more user interfaces displayed on one or more computing devices communicatively coupled with the one or more databases:

using the one or more computing devices, reading a data file generated by the third party software application, the data file comprising a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components;

determining using the one or more computing devices, for the subset of third party diagram components, correlated threat model components as defined in the one or more mapping files;

displaying on the one or more user interfaces a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, the relational diagram defining a threat model; and

generating, using the one or more computing devices, and displaying, on the one or more user interfaces, a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

2. The method of claim 1, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

IriusRisk
Ex. 1002 - Page 61

3. The method of claim 1, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

4. The method of claim 1, wherein the data file comprises one of a VSD file, a VSDX file, a TMT file, an XML file, a JSON file, a PNG file, and a JPEG file.

5. The method of claim 1, wherein determining, for the subset of third party diagram components, correlated threat model components, comprises first determining whether a user-defined name for one of the subset of third party diagram components has a match in the one or more mapping files and, if no match is found, then determining whether a default title for the third party diagram component, as defined by the third party software application, has a match in the one or more mapping files.

6. The method of claim 5, further comprising, if no match is found for the default title, matching the third party diagram component with a placeholder component and displaying the placeholder component on the relational diagram to indicate that no match was found.

7. The method of claim 1, further comprising determining, for the one or more relationships defined by the data file, correlated threat model components as defined in the one or more mapping files.

IriusRisk
Ex. 1002 - Page 62

8. A threat modeling system, comprising:

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components");

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by one of the one or more computing devices, the data file generated by the third party software application, the data file comprising a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of third party diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, the relational diagram defining a threat model; and

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

9. The system of claim 8, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

10. The system of claim 8, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

IriusRisk
Ex. 1002 - Page 63

11. The system of claim 8, wherein the data file comprises one of a VSD file, a VSDX file, a TMT File, an XML file, a JSON file, a PNG file, and a JPEG file.

12. The system of claim 8, wherein the one or more mapping files includes a file correlating user-defined names of the third party diagram components with the threat model components.

13. The system of claim 8, wherein the one or more mapping files includes a file correlating default titles of the third party diagram components with the threat model components.

14. The system of claim 8, wherein, for each of the subset of third party diagram components for which no correlated threat model component is found in the one or more mapping files, the diagram interface displays a placeholder component indicative of a match not being found.

15. The system of claim 8, further comprising one or more mapping interfaces displayed on the one or more computing devices and configured to, in response to receiving user input, modify the one or more mapping files.

IriusRisk
Ex. 1002 - Page 64

16. A threat modeling system, comprising:

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with VISIO diagram components;

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by the one or more computing devices, the data file comprising a subset of the VISIO diagram components, the data file defining one or more relationships between the subset of VISIO diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of VISIO diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of a computing network, using visual representations of the threat model components correlated with the subset of VISIO diagram components, the relational diagram defining a threat model; and

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

17. The system of claim 16, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

18. The system of claim 16, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

19. The system of claim 16, wherein the one or more mapping files includes a file correlating user-defined names of the third party diagram components with the threat model components.

IriusRisk
Ex. 1002 - Page 65

20. The system of claim 16, wherein the one or more mapping files includes a file correlating default titles of the third party diagram components with the threat model components.

IriusRisk
Ex. 1002 - Page 66

ABSTRACT OF THE DISCLOSURE

Threat modeling systems include one or more computing devices communicatively coupled with one or more databases, the database(s) including threat model components and threats associated with one another. One or more mapping files coupled with the database(s) correlate the threat model components with visual diagram components of a third party software application. An import interface initiates reading of a third party generated data file by the computing device(s), the data file including a subset of the third party diagram components and relationships between the subset. An interface receiving input initiates a determination of threat model components correlated with the subset. A diagram interface displays a relational diagram using visual representations of threat model components correlated with the subset, the relational diagram defining a threat model. A threat report interface includes a threat report displaying each threat that is associated with one of the threat model components of the threat model.

IriusRisk
Ex. 1002 - Page 67

Doc Code: TRACK1.REQ
Document Description: TrackOne Request

PTO/AIA/424 (04-14)

CERTIFICATION AND REQUEST FOR PRIORITIZED EXAMINATION UNDER 37 CFR 1.102(e) (Page 1 of 1)

First Named Inventor:	Anurag Agarwal	Nonprovisional Application Number (if known):	
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files		

APPLICANT HEREBY CERTIFIES THE FOLLOWING AND REQUESTS PRIORITIZED EXAMINATION FOR THE ABOVE-IDENTIFIED APPLICATION.

1. The processing fee set forth in 37 CFR 1.17(i)(1) and the prioritized examination fee set forth in 37 CFR 1.17(c) have been filed with the request. The publication fee requirement is met because that fee, set forth in 37 CFR 1.18(d), is currently $0. The basic filing fee, search fee, and examination fee are filed with the request or have been already been paid. I understand that any required excess claims fees or application size fee must be paid for the application.

2. I understand that the application may not contain, or be amended to contain, more than four independent claims, more than thirty total claims, or any multiple dependent claims, and that any request for an extension of time will cause an outstanding Track I request to be dismissed.

3. The applicable box is checked below:

I. ☑ Original Application (Track One) - Prioritized Examination under § 1.102(e)(1)

i. (a) The application is an original nonprovisional utility application filed under 35 U.S.C. 111(a). This certification and request is being filed with the utility application via EFS-Web.

---OR---

(b) The application is an original nonprovisional plant application filed under 35 U.S.C. 111(a). This certification and request is being filed with the plant application in paper.

ii. An executed inventor's oath or declaration under 37 CFR 1.63 or 37 CFR 1.64 for each inventor, **or** the application data sheet meeting the conditions specified in 37 CFR 1.53(f)(3)(i) is filed with the application.

II. ☐ Request for Continued Examination - Prioritized Examination under § 1.102(e)(2)

i. A request for continued examination has been filed with, or prior to, this form.
ii. If the application is a utility application, this certification and request is being filed via EFS-Web.
iii. The application is an original nonprovisional utility application filed under 35 U.S.C. 111(a), or is a national stage entry under 35 U.S.C. 371.
iv. This certification and request is being filed prior to the mailing of a first Office action responsive to the request for continued examination.
v. No prior request for continued examination has been granted prioritized examination status under 37 CFR 1.102(e)(2).

Signature /Paul B. Johnson/	Date 2019-05-31
Name (Print/Typed) Paul B. Johnson	Practitioner Registration Number 59375

Note: *This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. Submit multiple forms if more than one signature is required.**

☑ *Total of 1 forms are submitted.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76.
This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.

Secrecy Order 37 CFR 5.2:

☐ Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)

Inventor Information:

Inventor 1 — Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	Anurag		Agarwal	

Residence Information (Select One) ● US Residency Non US Residency Active US Military Service

City	State/Province	Country of Residence
Jersey City	NJ	US

Mailing Address of Inventor:

Address 1	105 Greene St. Unit 1801		
Address 2			
City	Jersey City	**State/Province**	NJ
Postal Code	07302	**Country**	US

All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the **Add** button. Add

Correspondence Information:

Enter either Customer Number or complete the Correspondence Information section below.
For further information see 37 CFR 1.33(a).

☐ **An Address is being provided for the correspondence Information of this application.**

Customer Number	145042		
Email Address	paul@pauljohnsonlaw.com	Add Email	Remove Email

Application Information:

Title of the Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files		
Attorney Docket Number	ThreatM010	**Small Entity Status Claimed**	☒
Application Type	Nonprovisional		
Subject Matter	Utility		
Total Number of Drawing Sheets (if any)	30	**Suggested Figure for Publication (if any)**	1

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

Filing By Reference:

Only complete this section when filing an application by reference under 35 U.S.C. 111(c) and 37 CFR 1.57(a). Do not complete this section if application papers including a specification and any drawings are being filed. Any domestic benefit or foreign priority information must be provided in the appropriate section(s) below (i.e., "Domestic Benefit/National Stage Information" and "Foreign Priority Information").

For the purposes of a filing date under 37 CFR 1.53(b), the description and any drawings of the present application are replaced by this reference to the previously filed application, subject to conditions and requirements of 37 CFR 1.57(a).

Application number of the previously filed application	Filing date (YYYY-MM-DD)	Intellectual Property Authority or Country

Publication Information:

☐ Request Early Publication (Fee required at time of Request 37 CFR 1.219)

☐ **Request Not to Publish.** I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application **has not and will not** be the subject of an application filed in another country, or under a multilateral international agreement, that requires publication at eighteen months after filing.

Representative Information:

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32).
Either enter Customer Number or complete the Representative Name section below. If both sections are completed the customer Number will be used for the Representative Information during processing.

Please Select One:	● Customer Number	US Patent Practitioner	○ Limited Recognition (37 CFR 11.9)
Customer Number	145042		

Domestic Benefit/National Stage Information:

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121, 365(c), or 386(c) or indicate National Stage entry from a PCT application. Providing benefit claim information in the Application Data Sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78.
When referring to the current application, please leave the "Application Number" field blank.

Prior Application Status	Pending		Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)
	Continuation in part of	16228738	2018-12-20

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

Prior Application Status	Patented				Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
16228738	Continuation in part of	15922856	2018-03-15	10200399	2019-02-05
Prior Application Status	Patented				Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	Patent Number	Issue Date (YYYY-MM-DD)
15922856	Continuation in part of	15888021	2018-02-03	10255439	2019-04-09
Prior Application Status	Expired				Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)		
15888021	Claims benefit of provisional	62507691	2017-05-17		
Prior Application Status	Expired				Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)		
15888021	Claims benefit of provisional	62520954	2017-06-16		
Prior Application Status	Expired				Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)		
15888021	Claims benefit of provisional	62527671	2017-06-30		
Prior Application Status	Expired				Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)		
15888021	Claims benefit of provisional	62530295	2017-07-10		

Additional Domestic Benefit/National Stage Data may be generated within this form by selecting the **Add** button. Add

Foreign Priority Information:

This section allows for the applicant to claim priority to a foreign application. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55. When priority is claimed to a foreign application that is eligible for retrieval under the priority document exchange program (PDX)[i] the information will be used by the Office to automatically attempt retrieval pursuant to 37 CFR 1.55(i)(1) and (2). Under the PDX program, applicant bears the ultimate responsibility for ensuring that a copy of the foreign application is received by the Office from the participating foreign intellectual property office, or a certified copy of the foreign priority application is filed, within the time period specified in 37 CFR 1.55(g)(1).

Remove

Application Number	Country[i]	Filing Date (YYYY-MM-DD)	Access Code[i] (if applicable)

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

Additional Foreign Priority Data may be generated within this form by selecting the **Add** button. Add

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications

☐ This application (1) claims priority to or the benefit of an application filed before March 16, 2013 and (2) also contains, or contained at any time, a claim to a claimed invention that has an effective filing date on or after March 16, 2013.
NOTE: By providing this statement under 37 CFR 1.55 or 1.78, this application, with a filing date on or after March 16, 2013, will be examined under the first inventor to file provisions of the AIA.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

Authorization or Opt-Out of Authorization to Permit Access:

When this Application Data Sheet is properly signed and filed with the application, applicant has provided written authority to permit a participating foreign intellectual property (IP) office access to the instant application-as-filed (see paragraph A in subsection 1 below) and the European Patent Office (EPO) access to any search results from the instant application (see paragraph B in subsection 1 below).

Should applicant choose not to provide an authorization identified in subsection 1 below, applicant **must opt-out** of the authorization by checking the corresponding box A or B or both in subsection 2 below.

NOTE: This section of the Application Data Sheet is **ONLY** reviewed and processed with the **INITIAL** filing of an application. After the initial filing of an application, an Application Data Sheet cannot be used to provide or rescind authorization for access by a foreign IP office(s). Instead, Form PTO/SB/39 or PTO/SB/69 must be used as appropriate.

1. Authorization to Permit Access by a Foreign Intellectual Property Office(s)

A. Priority Document Exchange (PDX) - Unless box A in subsection 2 (opt-out of authorization) is checked, the undersigned hereby **grants the USPTO authority** to provide the European Patent Office (EPO), the Japan Patent Office (JPO), the Korean Intellectual Property Office (KIPO), the State Intellectual Property Office of the People's Republic of China (SIPO), the World Intellectual Property Organization (WIPO), and any other foreign intellectual property office participating with the USPTO in a bilateral or multilateral priority document exchange agreement in which a foreign application claiming priority to the instant patent application is filed, access to: (1) the instant patent application-as-filed and its related bibliographic data, (2) any foreign or domestic application to which priority or benefit is claimed by the instant application and its related bibliographic data, and (3) the date of filing of this Authorization. See 37 CFR 1.14(h)(1).

B. Search Results from U.S. Application to EPO - Unless box B in subsection 2 (opt-out of authorization) is checked, the undersigned hereby **grants the USPTO authority** to provide the EPO access to the bibliographic data and search results from the instant patent application when a European patent application claiming priority to the instant patent application is filed. See 37 CFR 1.14(h)(2).

The applicant is reminded that the EPO's Rule 141(1) EPC (European Patent Convention) requires applicants to submit a copy of search results from the instant application without delay in a European patent application that claims priority to the instant application.

2. Opt-Out of Authorizations to Permit Access by a Foreign Intellectual Property Office(s)

☐ A. Applicant **DOES NOT** authorize the USPTO to permit a participating foreign IP office access to the instant application-as-filed. If this box is checked, the USPTO will not be providing a participating foreign IP office with any documents and information identified in subsection 1A above.

☐ B. Applicant **DOES NOT** authorize the USPTO to transmit to the EPO any search results from the instant patent application. If this box is checked, the USPTO will not be providing the EPO with search results from the instant application.

NOTE: Once the application has published or is otherwise publicly available, the USPTO may provide access to the application in accordance with 37 CFR 1.14.

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

Applicant Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

Applicant 1 — Remove

If the applicant is the inventor (or the remaining joint inventor or inventors under 37 CFR 1.45), this section should not be completed. The information to be provided in this section is the name and address of the legal representative who is the applicant under 37 CFR 1.43; or the name and address of the assignee, person to whom the inventor is under an obligation to assign the invention, or person who otherwise shows sufficient proprietary interest in the matter who is the applicant under 37 CFR 1.46. If the applicant is an applicant under 37 CFR 1.46 (assignee, person to whom the inventor is obligated to assign, or person who otherwise shows sufficient proprietary interest) together with one or more joint inventors, then the joint inventor or inventors who are also the applicant should be identified in this section.

Clear

● Assignee	Legal Representative under 35 U.S.C. 117	Joint Inventor
Person to whom the inventor is obligated to assign.	Person who shows sufficient proprietary interest	

If applicant is the legal representative, indicate the authority to file the patent application, the inventor is:

Name of the Deceased or Legally Incapacitated Inventor:

If the Applicant is an Organization check here. ☒

Organization Name	ThreatModeler Software Inc.		
Mailing Address Information For Applicant:			
Address 1	101 Hudson St., Ste. 2100		
Address 2			
City	Jersey City	**State/Province**	NJ
Country	US	Postal Code	07302
Phone Number		Fax Number	
Email Address			

Additional Applicant Data may be generated within this form by selecting the Add button. Add

Assignee Information including Non-Applicant Assignee Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

PTO/AIA/14 (02-18)
Approved for use through 11/30/2020. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

Assignee 1

Complete this section if assignee information, including non-applicant assignee information, is desired to be included on the patent application publication. An assignee-applicant identified in the "Applicant Information" section will appear on the patent application publication as an applicant. For an assignee-applicant, complete this section only if identification as an assignee is also desired on the patent application publication.

Remove

If the Assignee or Non-Applicant Assignee is an Organization check here. ☒

Organization Name	ThreatModeler Software Inc.

Mailing Address Information For Assignee including Non-Applicant Assignee:

Address 1	101 Hudson St., Ste. 2100		
Address 2			
City	Jersey City	**State/Province**	NJ
Country i	US	Postal Code	07302
Phone Number		Fax Number	
Email Address			

Additional Assignee or Non-Applicant Assignee Data may be generated within this form by selecting the Add button. Add

Signature:

Remove

NOTE: This Application Data Sheet must be signed in accordance with 37 CFR 1.33(b). **However, if this Application Data Sheet is submitted with the INITIAL filing of the application and either box A or B is not checked in subsection 2 of the "Authorization or Opt-Out of Authorization to Permit Access" section, then this form must also be signed in accordance with 37 CFR 1.14(c).**

This Application Data Sheet **must** be signed by a patent practitioner if one or more of the applicants is a **juristic entity** (e.g., corporation or association). If the applicant is two or more joint inventors, this form must be signed by a patent practitioner, **all** joint inventors who are the applicant, or one or more joint inventor-applicants who have been given power of attorney (e.g., see USPTO Form PTO/AIA/81) on behalf of **all** joint inventor-applicants.

See 37 CFR 1.4(d) for the manner of making signatures and certifications.

Signature	/Paul B. Johnson/			Date (YYYY-MM-DD)	2019-08-15
First Name	Paul	Last Name	Johnson	Registration Number	59375

Additional Signature may be generated within this form by selecting the Add button. Add

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	ThreatM010
	Application Number	
Title of Invention	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files	

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1 The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3 A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent CooperationTreaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Patent Application Fee Transmittal

Application Number:	
Filing Date:	
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Filer:	Paul B. Johnson
Attorney Docket Number:	ThreatM010

Filed as Small Entity

Filing Fees for Track I Prioritized Examination - Nonprovisional Application under 35 USC 111(a)

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
UTILITY FILING FEE (ELECTRONIC FILING)	4011	1	75	75
UTILITY SEARCH FEE	2111	1	330	330
UTILITY EXAMINATION FEE	2311	1	380	380
REQUEST FOR PRIORITIZED EXAMINATION	2817	1	2000	2000
Pages:				
Claims:				
Miscellaneous-Filing:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
PUBL. FEE- EARLY, VOLUNTARY, OR NORMAL	1504	1	0	0
PROCESSING FEE, EXCEPT PROV. APPLS.	2830	1	70	70

Petition:	
Patent-Appeals-and-Interference:	
Post-Allowance-and-Post-Issuance:	
Extension-of-Time:	
Miscellaneous:	
Total in USD ($)	**2855**

Electronic Acknowledgement Receipt

EFS ID:	36892393
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	15-AUG-2019
Filing Date:	
Time Stamp:	22:11:05
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	CARD
Payment was successfully received in RAM	$2855
RAM confirmation Number	E20198EM11333313
Deposit Account	
Authorized User	

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Oath or Declaration filed	DEC010.pdf	63220 1bdbe8e3e724a93fe1c7de85978f939e5d8b56eb	no	1
Warnings:					
Information:					
2	Drawings-only black and white line drawings	Drawings010.pdf	687855 4cb04c97189ccbe67288534db395bec108c242e6	no	30
Warnings:					
Information:					
3	Power of Attorney	POA.pdf	311536 23a5454943de534d179b07dec4747f0c1009e717	no	2
Warnings:					
Information:					
4		UtilityApp010.pdf	383254 d4aec670f29a8b2a6243e3ba8e7718be1cdc44ad	yes	65

Multipart Description/PDF files in .zip description

Document Description	Start	End
Specification	1	58
Claims	59	64
Abstract	65	65

Warnings:					
Information:					
5	TrackOne Request	aia0424.pdf	124424 af6f34fede4dbadd7a71e45264d93daa8e6e6cea	no	2

Warnings:					
Information:					
6	Application Data Sheet	ADS1.pdf	1256552 51e718edca774ae146e8aaf6d06f0506ba847061	no	9
Warnings:					
Information:					
7	Fee Worksheet (SB06)	fee-info.pdf	40343 ff24129a6312237b8f7133fc86b2029dc9d03b8e	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	2867184		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Docket No. THREATM010

Declaration for Patent Application

As the below named inventor, I hereby declare that:

This declaration is directed to the attached application.

The above-identified application was made or authorized to be made by me. I believe that I am the original inventor or an original joint inventor of a claimed invention in the application.

I acknowledge that any willful false statement made in this declaration is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five (5) years, or both.

Legal Name: Anuraag Agarwwal

Print or Type

Signature: Anuraag

Date: 06/26/2019

Mailing Address: 105 Greene St. Unit 1801

Jersey City, NJ 07302

Residence:

☑ Same as mailing address, or:

IriusRisk
Ex. 1002 - Page 84

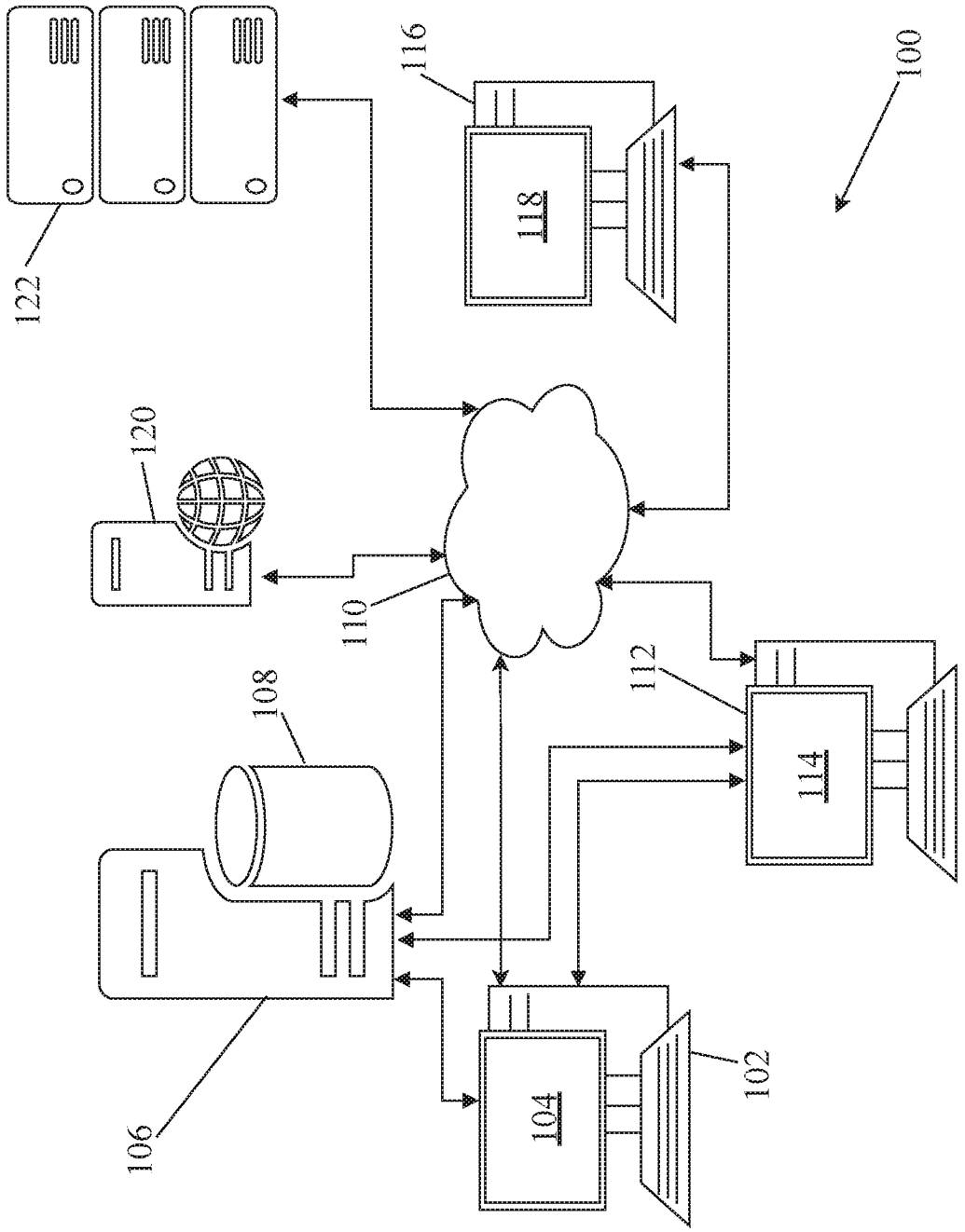


FIG. 1

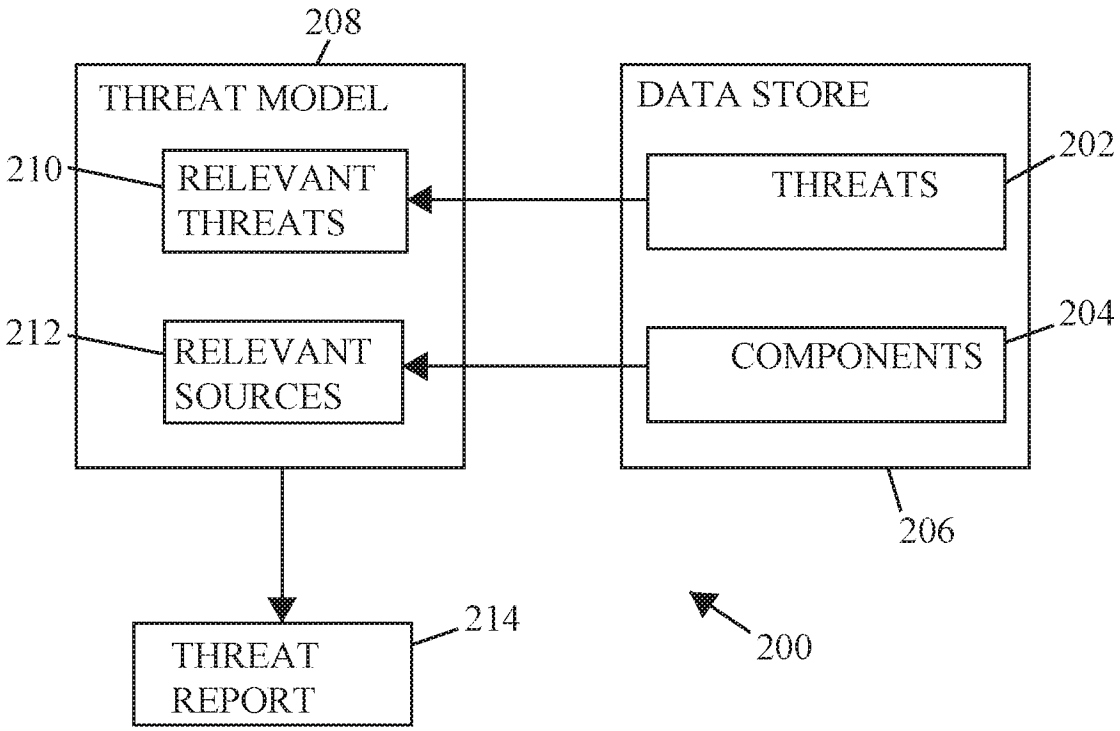

208
THREAT MODEL
DATA STORE
210
RELEVANT THREATS
202
THREATS
212
RELEVANT SOURCES
204
COMPONENTS
206
214
THREAT REPORT
200

FIG. 2

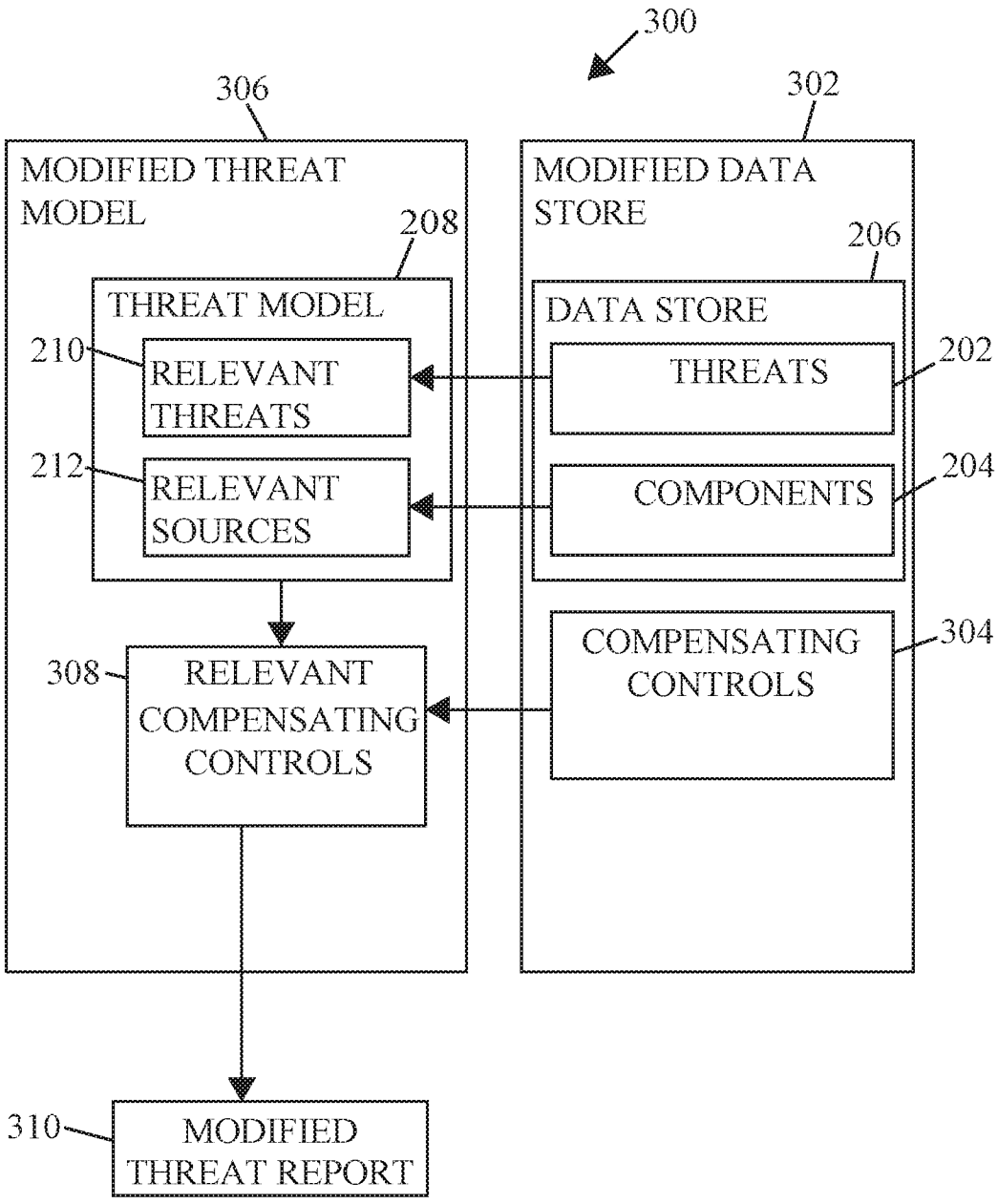

300
306
302
MODIFIED THREAT MODEL
208
MODIFIED DATA STORE
206
THREAT MODEL
DATA STORE
210
RELEVANT THREATS
THREATS
202
212
RELEVANT SOURCES
COMPONENTS
204
COMPENSATING CONTROLS
304
308
RELEVANT COMPENSATING CONTROLS
310
MODIFIED THREAT REPORT

FIG. 3

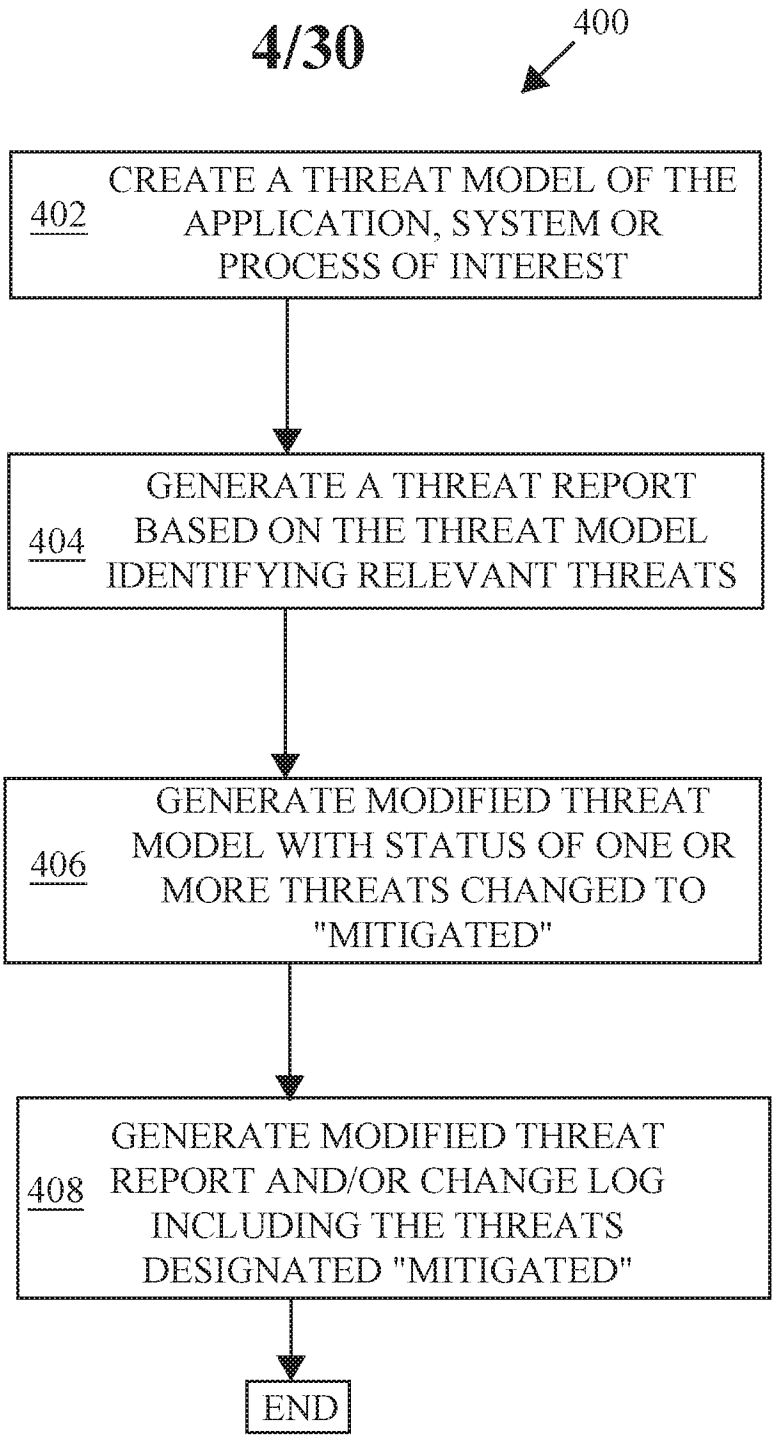

400
402
CREATE A THREAT MODEL OF THE APPLICATION, SYSTEM OR PROCESS OF INTEREST
404
GENERATE A THREAT REPORT BASED ON THE THREAT MODEL IDENTIFYING RELEVANT THREATS
406
GENERATE MODIFIED THREAT MODEL WITH STATUS OF ONE OR MORE THREATS CHANGED TO "MITIGATED"
408
GENERATE MODIFIED THREAT REPORT AND/OR CHANGE LOG INCLUDING THE THREATS DESIGNATED "MITIGATED"
END

FIG. 4

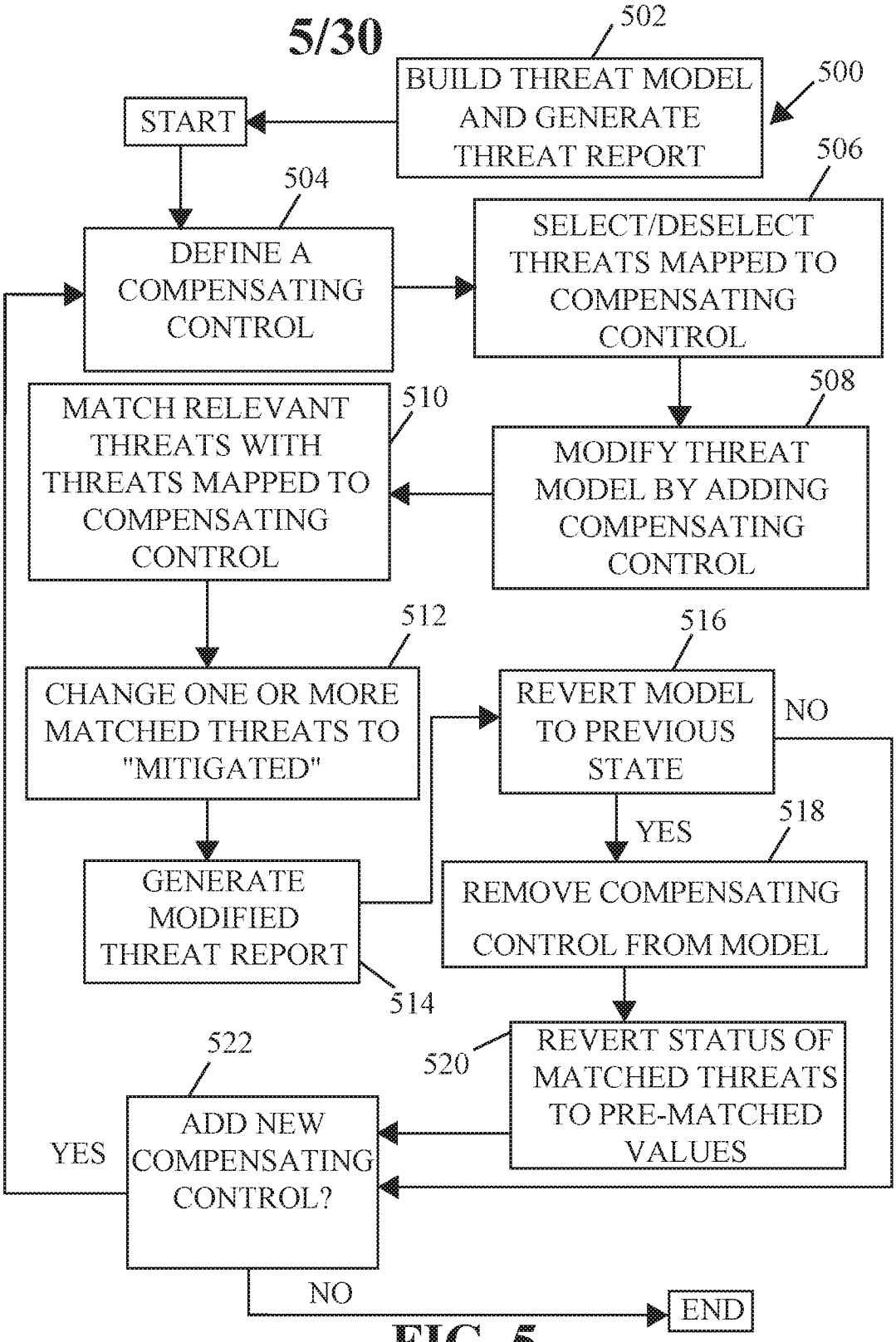

502
BUILD THREAT MODEL AND GENERATE THREAT REPORT
500
START
504
DEFINE A COMPENSATING CONTROL
506
SELECT/DESELECT THREATS MAPPED TO COMPENSATING CONTROL
508
MODIFY THREAT MODEL BY ADDING COMPENSATING CONTROL
510
MATCH RELEVANT THREATS WITH THREATS MAPPED TO COMPENSATING CONTROL
512
CHANGE ONE OR MORE MATCHED THREATS TO "MITIGATED"
516
REVERT MODEL TO PREVIOUS STATE
NO
YES
518
REMOVE COMPENSATING CONTROL FROM MODEL
GENERATE MODIFIED THREAT REPORT
514
520
REVERT STATUS OF MATCHED THREATS TO PRE-MATCHED VALUES
522
ADD NEW COMPENSATING CONTROL?
YES
NO
END

FIG. 5

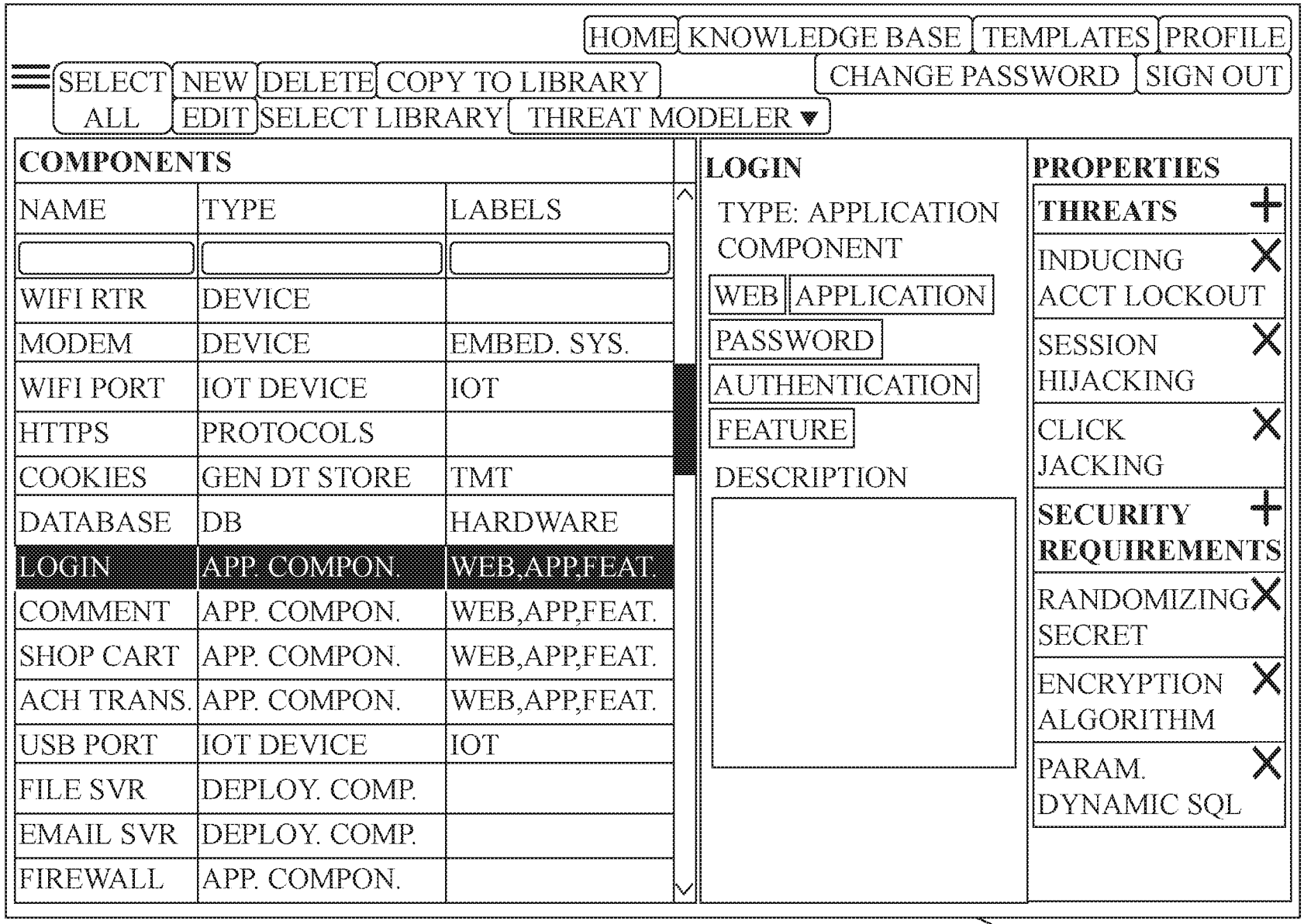


FIG. 6

6/30

IriusRisk
Ex. 1002 - Page 90

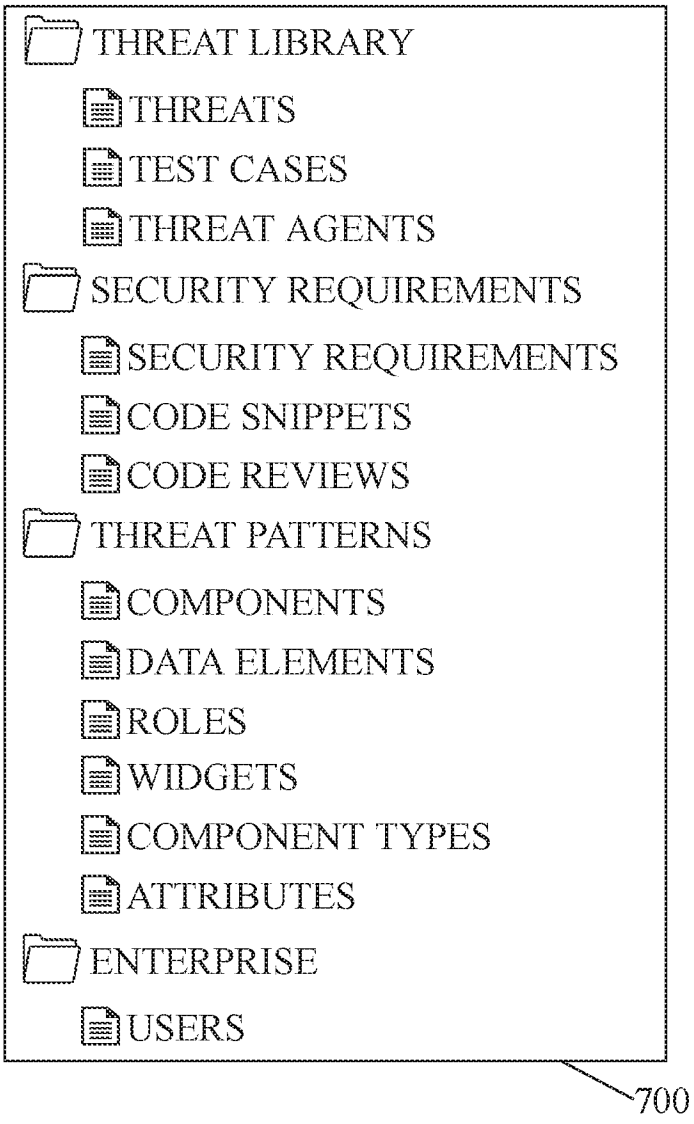

THREAT LIBRARY
THREATS
TEST CASES
THREAT AGENTS
SECURITY REQUIREMENTS
SECURITY REQUIREMENTS
CODE SNIPPETS
CODE REVIEWS
THREAT PATTERNS
COMPONENTS
DATA ELEMENTS
ROLES
WIDGETS
COMPONENT TYPES
ATTRIBUTES
ENTERPRISE
USERS
700

FIG. 7

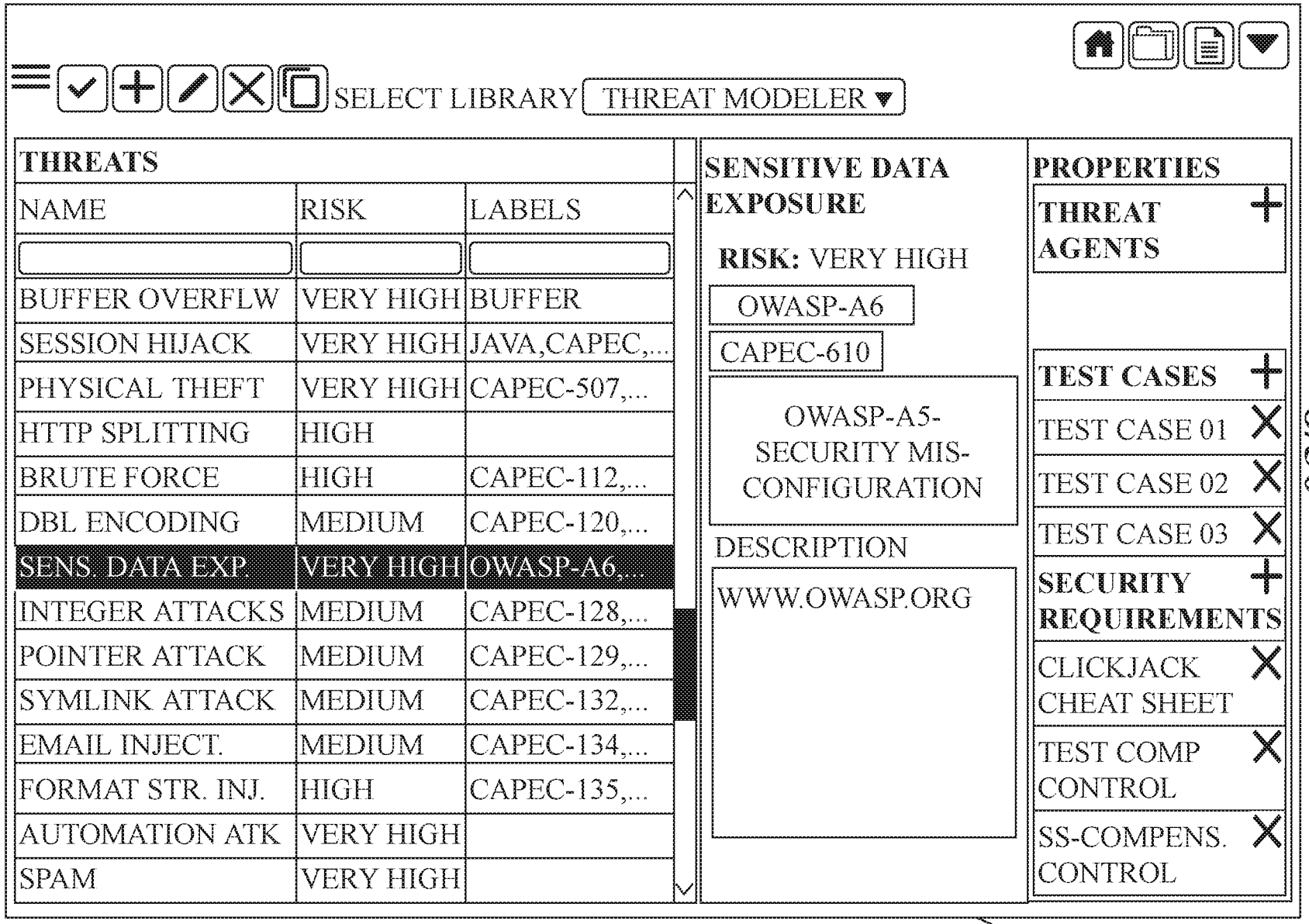


FIG. 8

8/30

IriusRisk
Ex. 1002 - Page 92

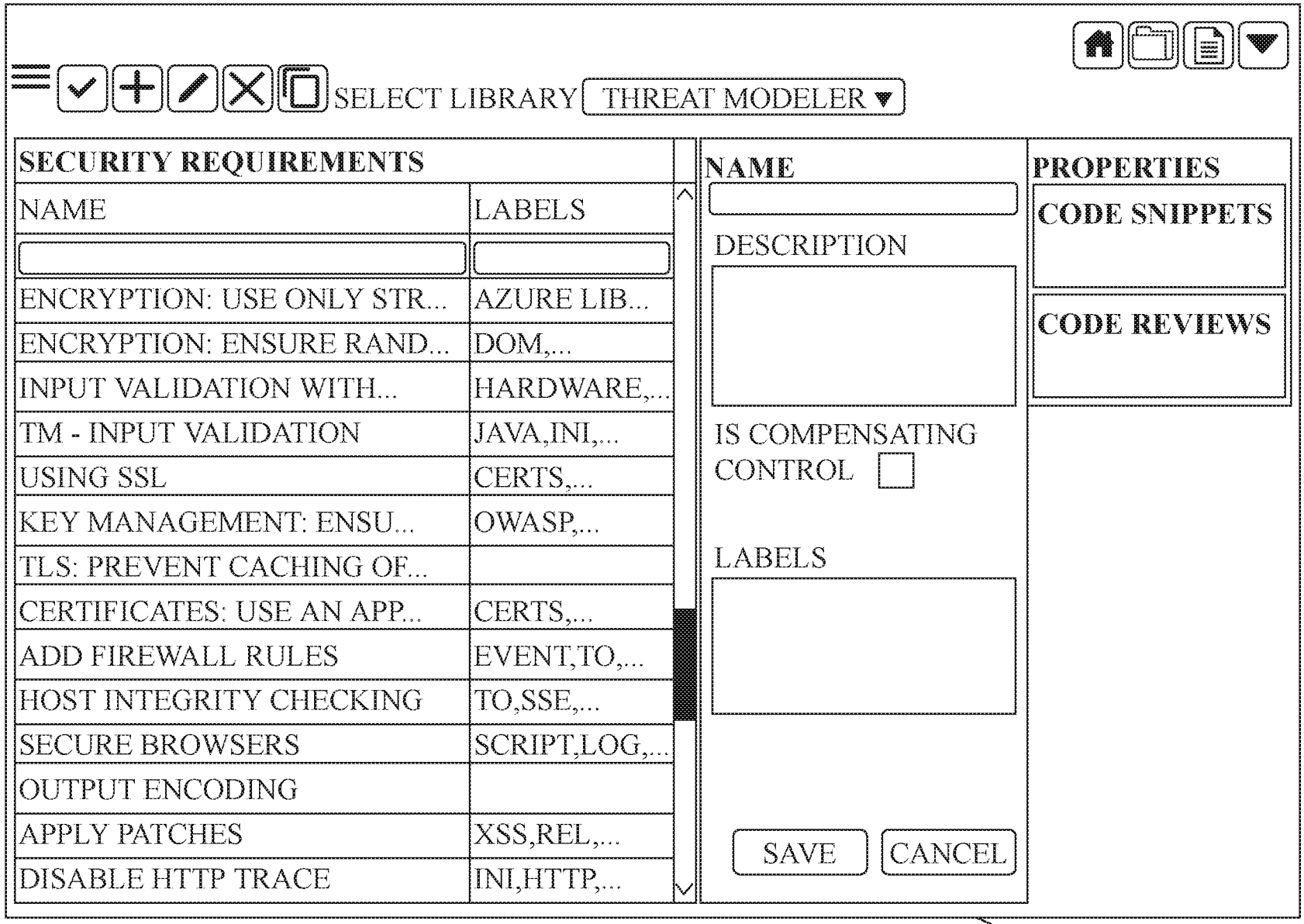
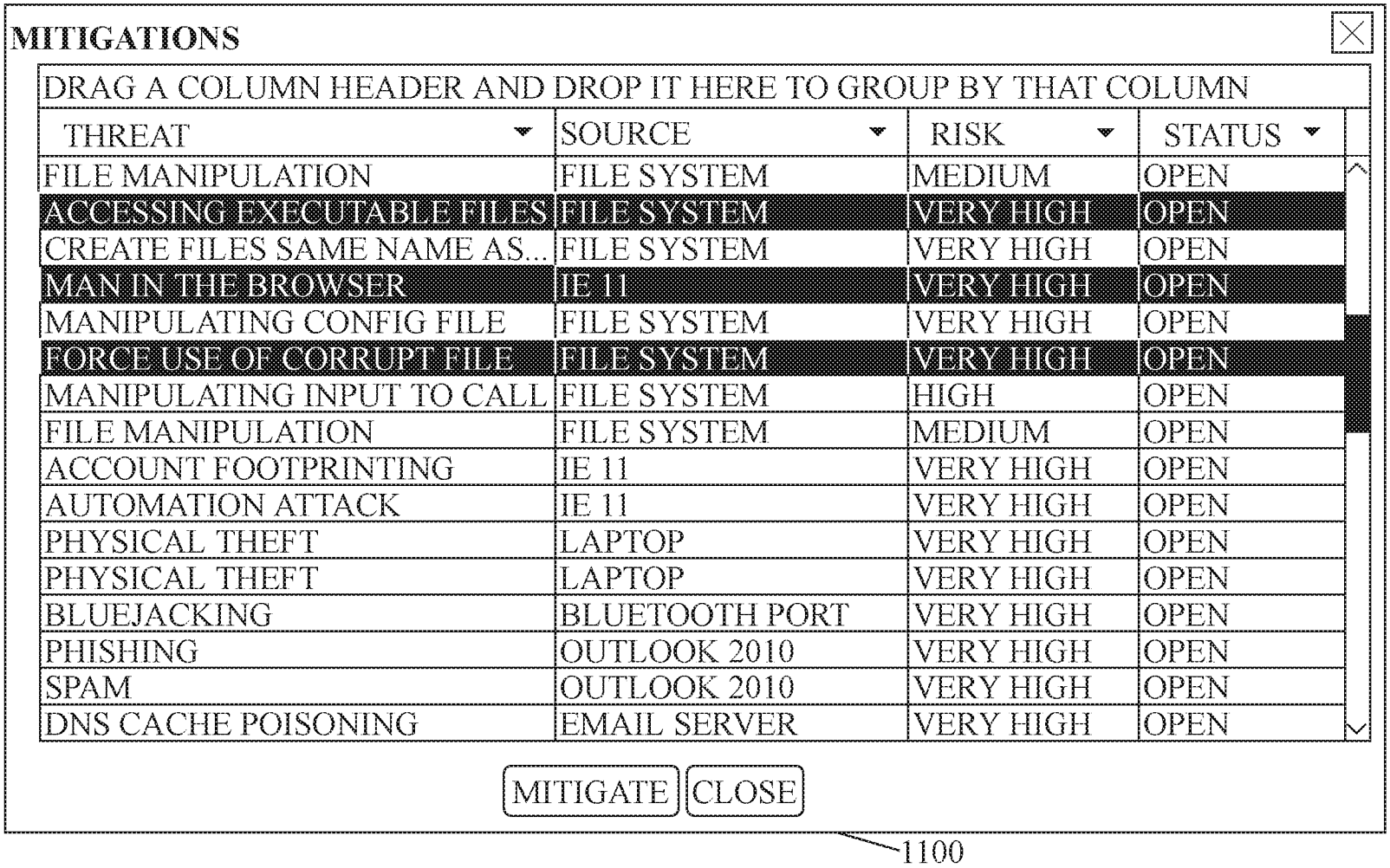


FIG. 9

IriusRisk
Ex. 1002 - Page 93

SELECT ALL
COPY
PASTE
DELETE
CLEAR CANVAS
ZOOM IN
ZOOM OUT
100%
COMMENT OVERVIEW
GROUP
UNGROUP
UNDO
REDO
TEMPLATE
SAVE AS IMAGE
PDFSETTINGS
SAVE AS TEMPLATE
SNAP SHOTS
COMPENSATING CONTROLS
TOOLBOX
SEARCH
POS
AD
IRON PORT
AV VSE
SHARE POINT
USB PORT
GUI
MS WORD
EMAIL
AV HIPS
AKAMAI DNS
BIT LOCKER
PRINTER
ETHER-NET PORT
CHECK OUT
SSO
FILE SVR
EMAIL SVR
BT PORT
FILE SYS
SHOP CART
HDMI PORT
PAY PAL
ANTI VIRUS
FIRE WALL
WIFI PORT
DLP
SKYPE
BT PROXY
PDF CLIENT
ENDPOINT SECURITY V.1
PROPERTIES
GENERAL
COMPONENTS
DATA ELEMENTS
ROLES
WIDGETS
OFFICE NETWORK
WIFI ACCESS POINT
WIFI
LAPTOP
WIFI PORT
USB PORT
BT PORT
ETHER-NET PORT
HDMI PORT
WINDOWS 7
FILE SYS
PDF CLIENT
SKYPE
OUT LOOK 2010
MCAFEE HIPS
MCAFEE AV VSE
IE 11
OFFICE 2010
BIT LOCKER
MCAFEE DLP
1002
TCP
HTTPS
SMB
PRINTER
SHARE POINT
FILE SVR
IRON PORT
MULT
EMAIL SVR
BLUE COAT PROXY
EXT EMAIL GTW
AKAMAI DNS
FIRE WALL
AZURE
ONE DRIVE
SSO
1004
1000

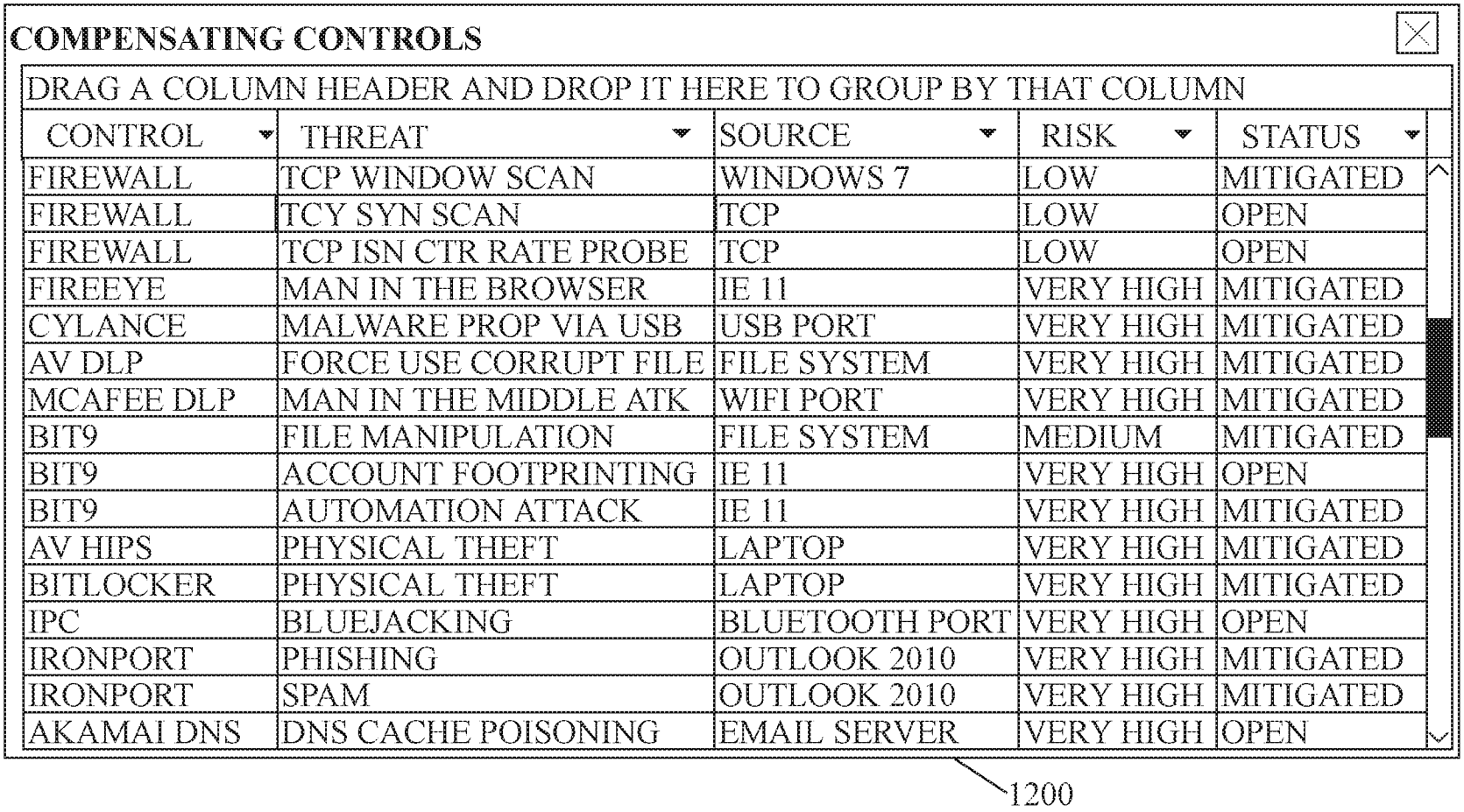
FIG. 10



11/30

FIG. 11

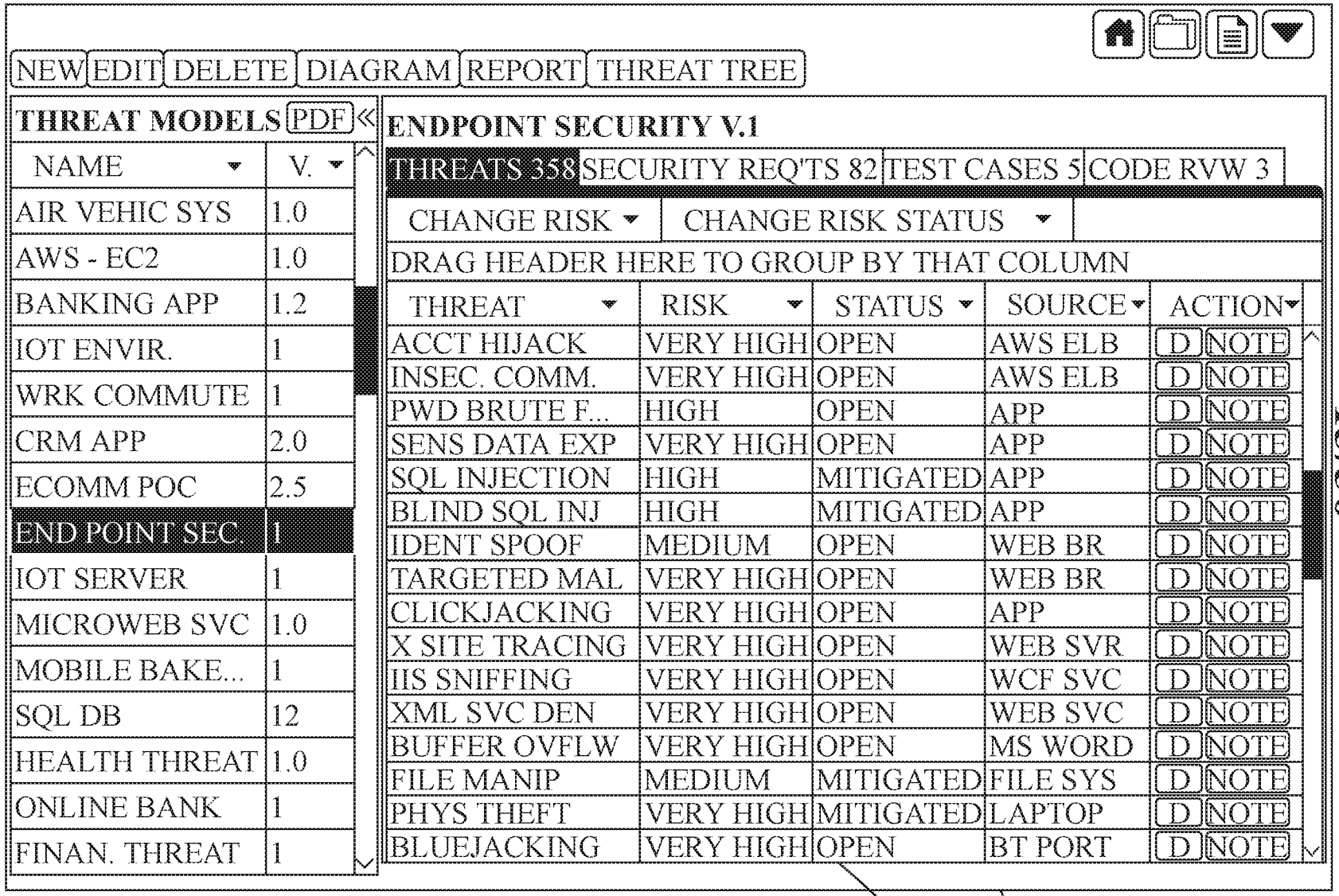
IriusRisk
Ex. 1002 - Page 95



12/30

FIG. 12

IriusRisk
Ex. 1002 - Page 96



13/30

FIG. 13

IriusRisk
Ex. 1002 - Page 97

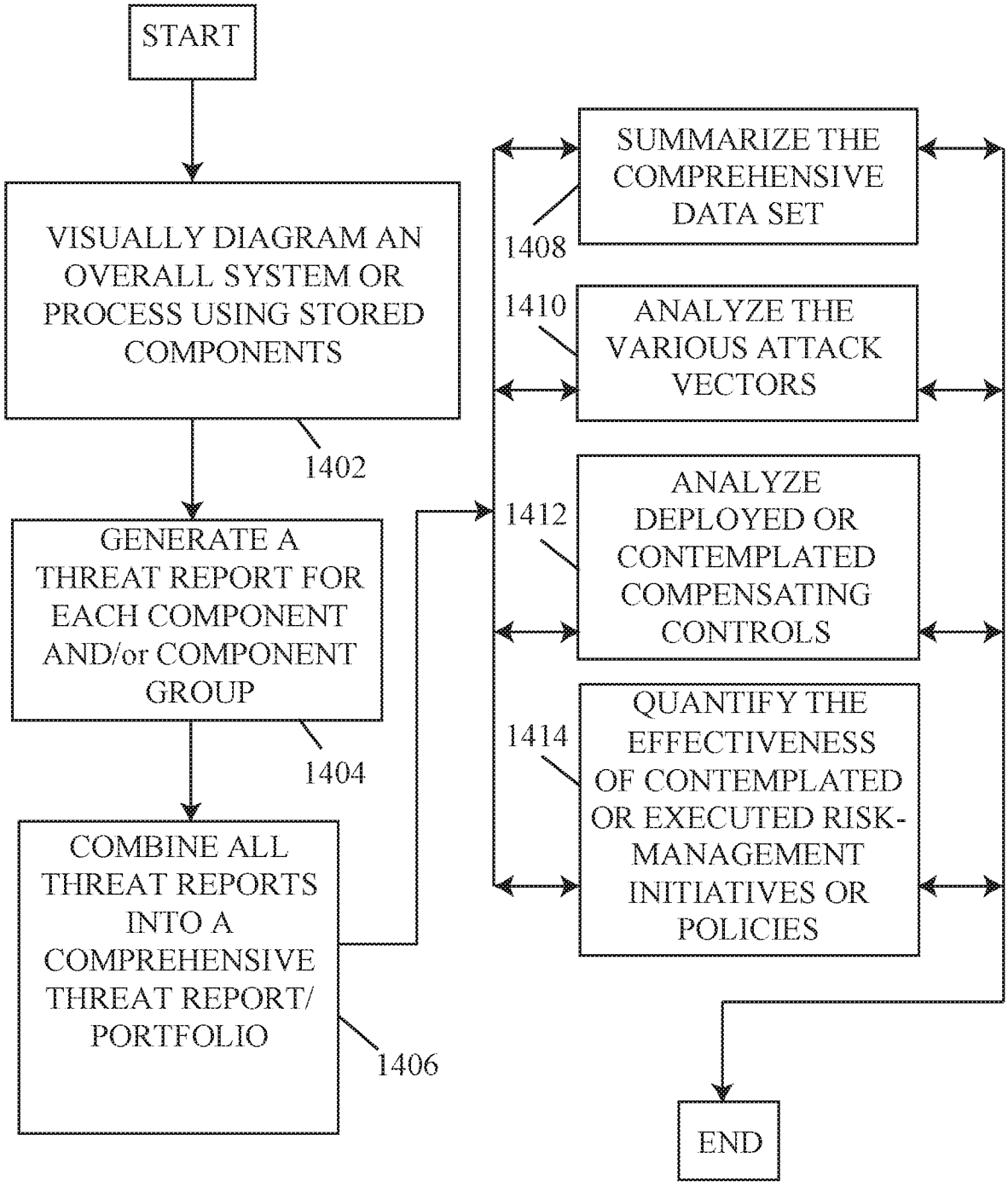

START
VISUALLY DIAGRAM AN OVERALL SYSTEM OR PROCESS USING STORED COMPONENTS
1402
GENERATE A THREAT REPORT FOR EACH COMPONENT AND/or COMPONENT GROUP
1404
COMBINE ALL THREAT REPORTS INTO A COMPREHENSIVE THREAT REPORT/ PORTFOLIO
1406
SUMMARIZE THE COMPREHENSIVE DATA SET
1408
1410
ANALYZE THE VARIOUS ATTACK VECTORS
1412
ANALYZE DEPLOYED OR CONTEMPLATED COMPENSATING CONTROLS
1414
QUANTIFY THE EFFECTIVENESS OF CONTEMPLATED OR EXECUTED RISK-MANAGEMENT INITIATIVES OR POLICIES
END

FIG. 14

SELECT ALL
COPY
PASTE
DELETE
CLEAR CANVAS
ZOOM IN
ZOOM OUT
100%
COMMENT
OVERVIEW
GROUP
UNGROUP
UNDO
REDO
TEMPLATE
SAVE AS IMAGE
PDFSETTINGS
SAVE AS TEMPLATE
SNAP SHOTS
COMPENSATING CONTROLS
TOOLBOX
SEARCH
POS
AD
IRON
AV
SHARE POINT
PORT
VSE
USB PORT
GUI
MS WORD
EMAIL
AV
HIPS
AKAMAI DNS
BIT LOCKER
PRINTER
ETHER-NET PORT
CHECK OUT
SSO
EMAIL SVR
BT PORT
FILE SVR
SHOP CART
HDMI PORT
FILE SYS
ANTI VIRUS DLP
FIRE WALL
PAY PAL
BC PROXY
SKYPE
WIFI PORT
PDF CLIENT
ENDPOINT SECURITY V.1
PROPERTIES
GENERAL
COMPONENTS
DATA ELEMENTS
ROLES
WIDGETS
1002
1000

FIG. 15

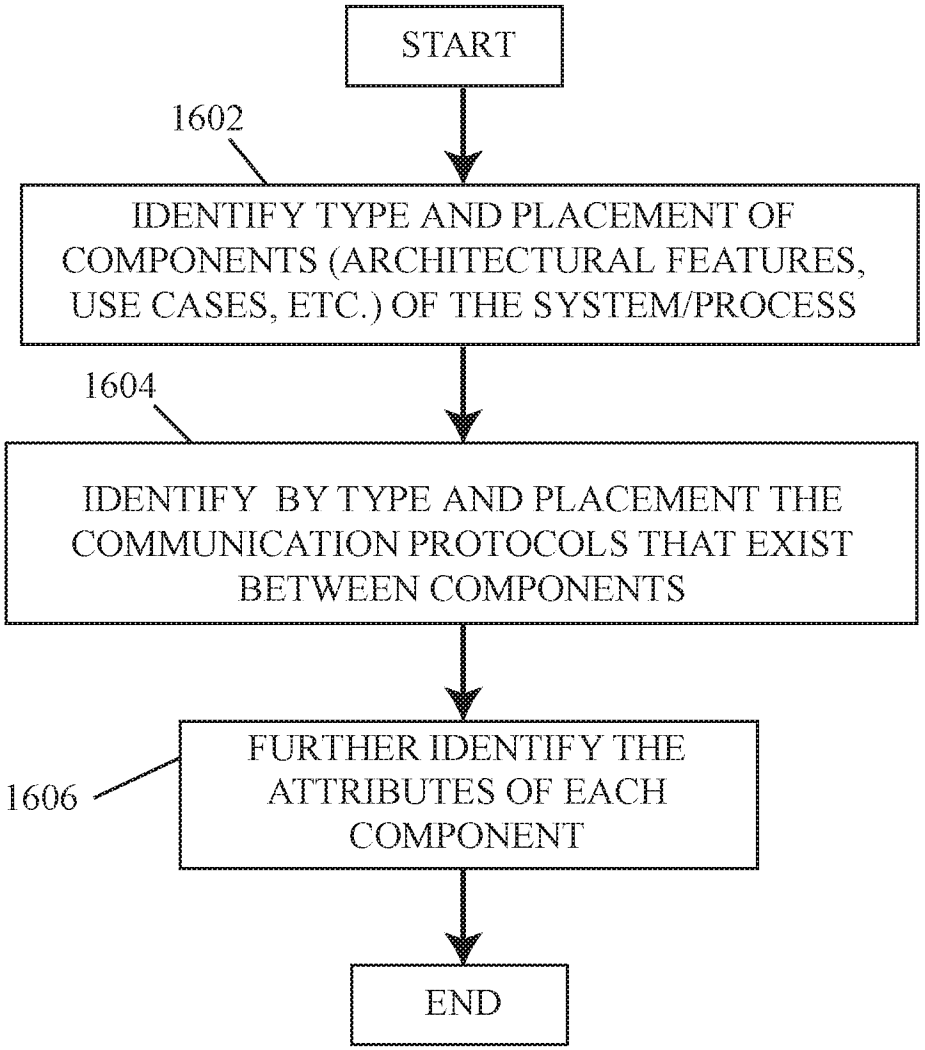

START
1602
IDENTIFY TYPE AND PLACEMENT OF COMPONENTS (ARCHITECTURAL FEATURES, USE CASES, ETC.) OF THE SYSTEM/PROCESS
1604
IDENTIFY BY TYPE AND PLACEMENT THE COMMUNICATION PROTOCOLS THAT EXIST BETWEEN COMPONENTS
1606
FURTHER IDENTIFY THE ATTRIBUTES OF EACH COMPONENT
END

FIG. 16

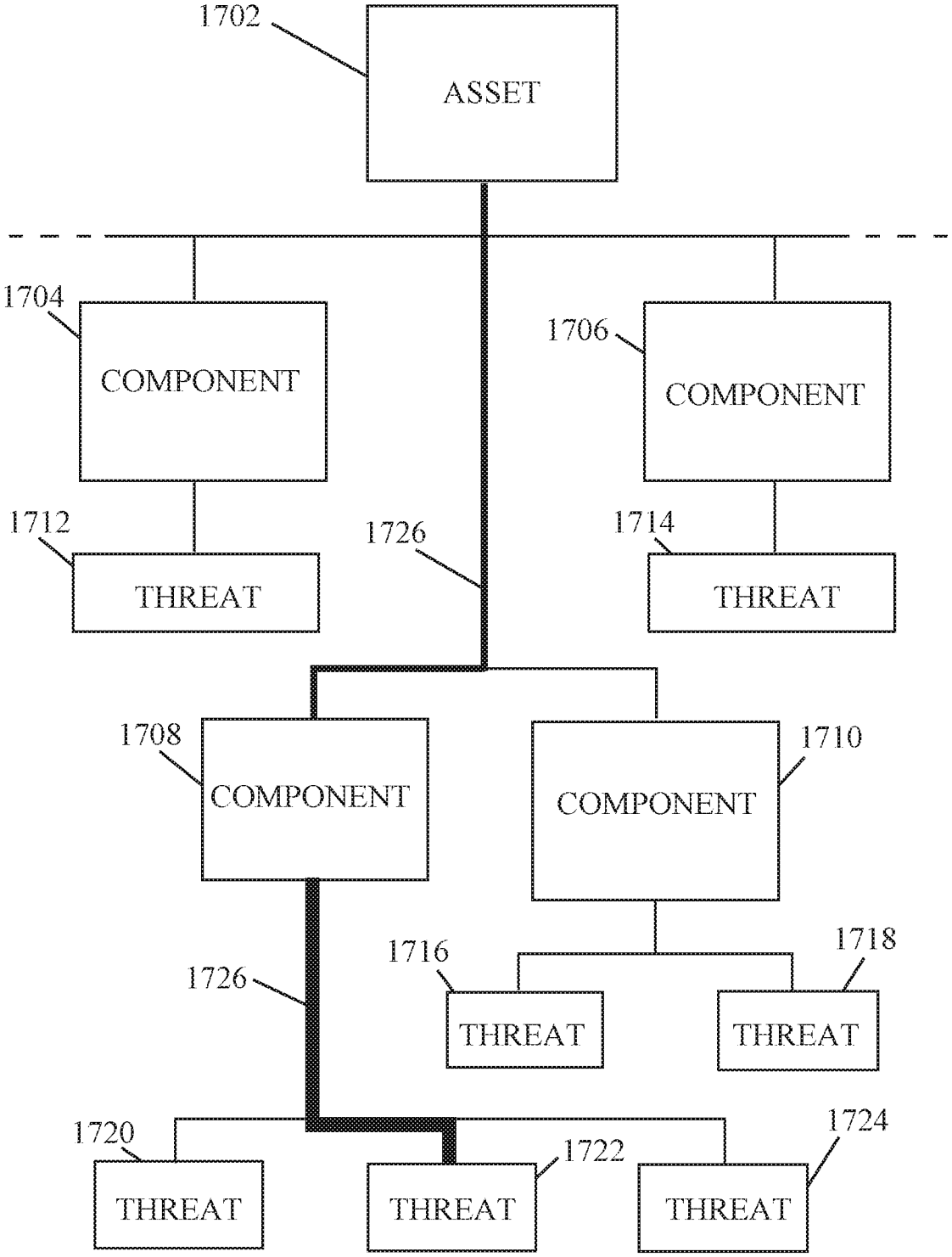

1702
ASSET
1704
COMPONENT
1706
COMPONENT
1712
THREAT
1726
1714
THREAT
1708
COMPONENT
1710
COMPONENT
1726
1716
THREAT
1718
THREAT
1720
THREAT
1722
THREAT
1724
THREAT

FIG. 17

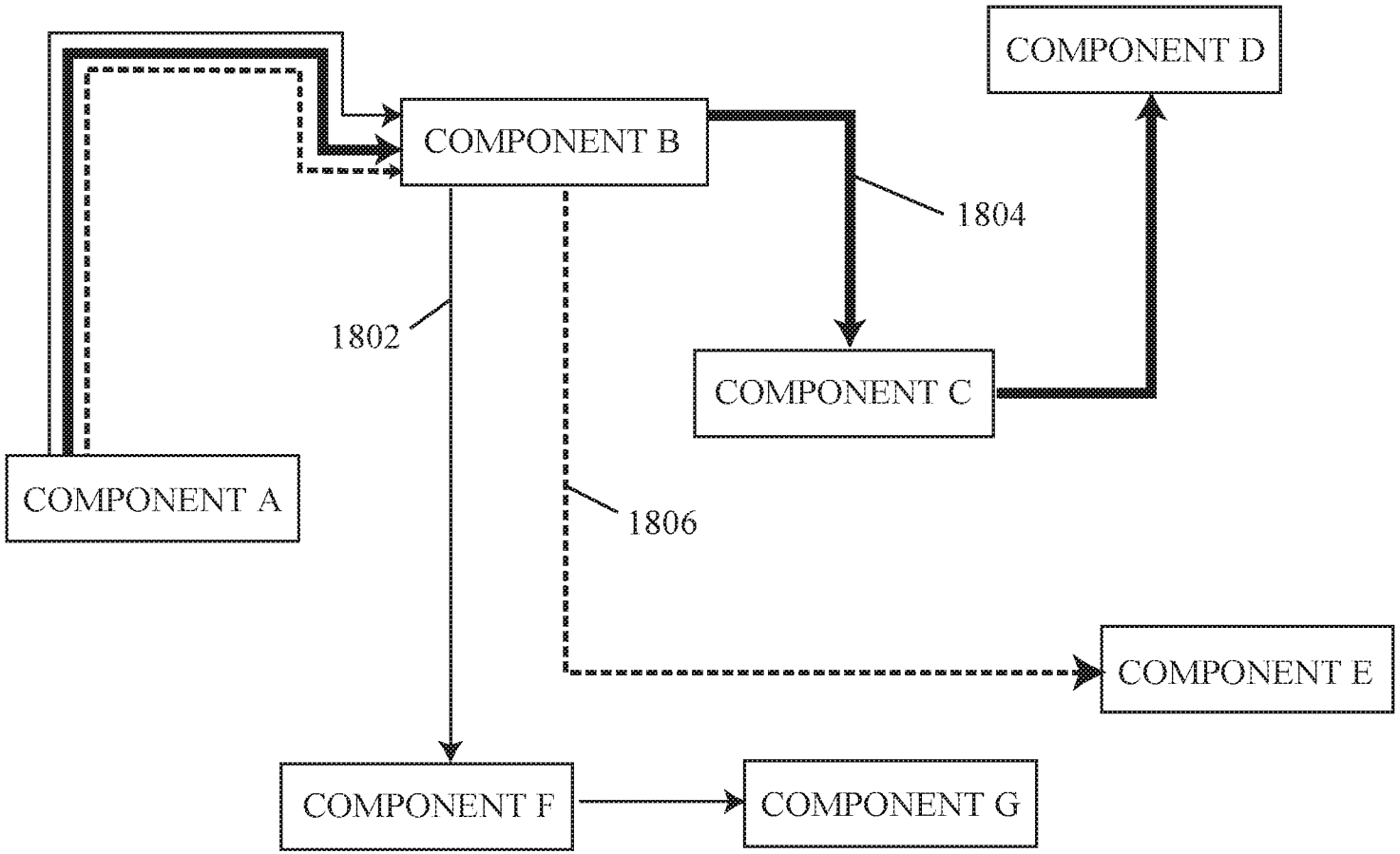


FIG. 18

IriusRisk
Ex. 1002 - Page 102

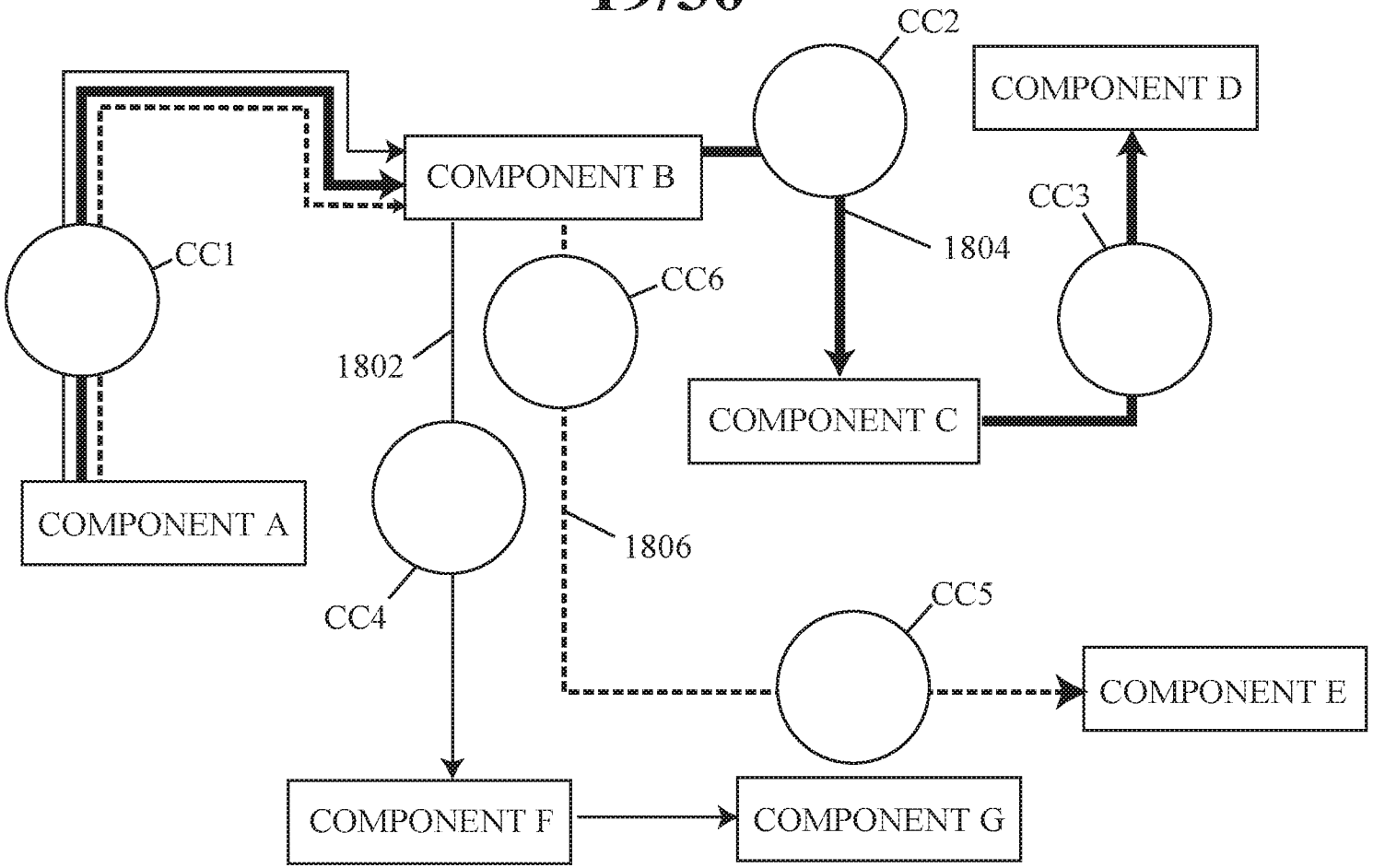


FIG. 19

IriusRisk
Ex. 1002 - Page 103

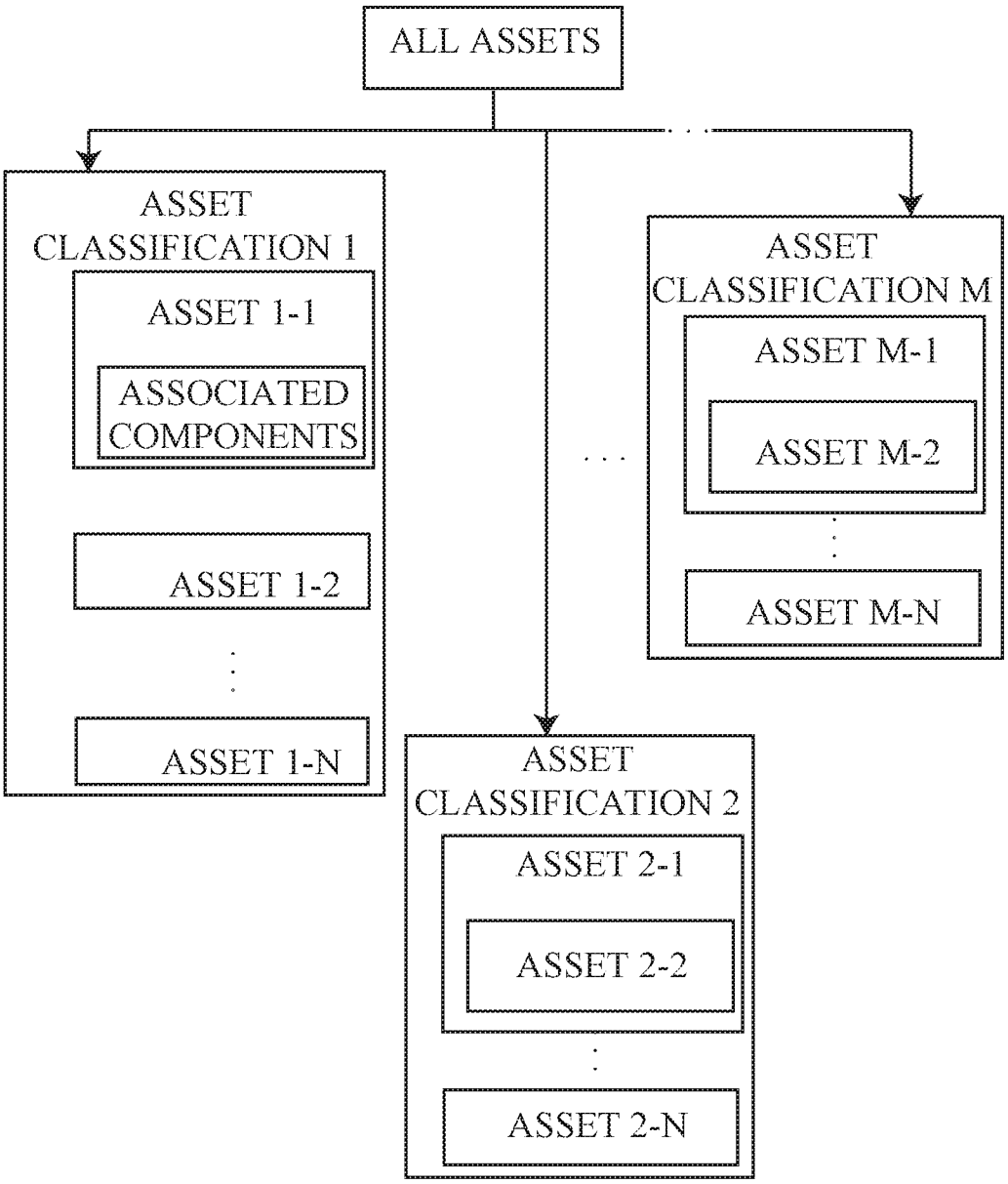

ALL ASSETS
ASSET CLASSIFICATION 1
ASSET 1-1
ASSOCIATED COMPONENTS
ASSET 1-2
ASSET 1-N
ASSET CLASSIFICATION 2
ASSET 2-1
ASSET 2-2
ASSET 2-N
ASSET CLASSIFICATION M
ASSET M-1
ASSET M-2
ASSET M-N

FIG. 20

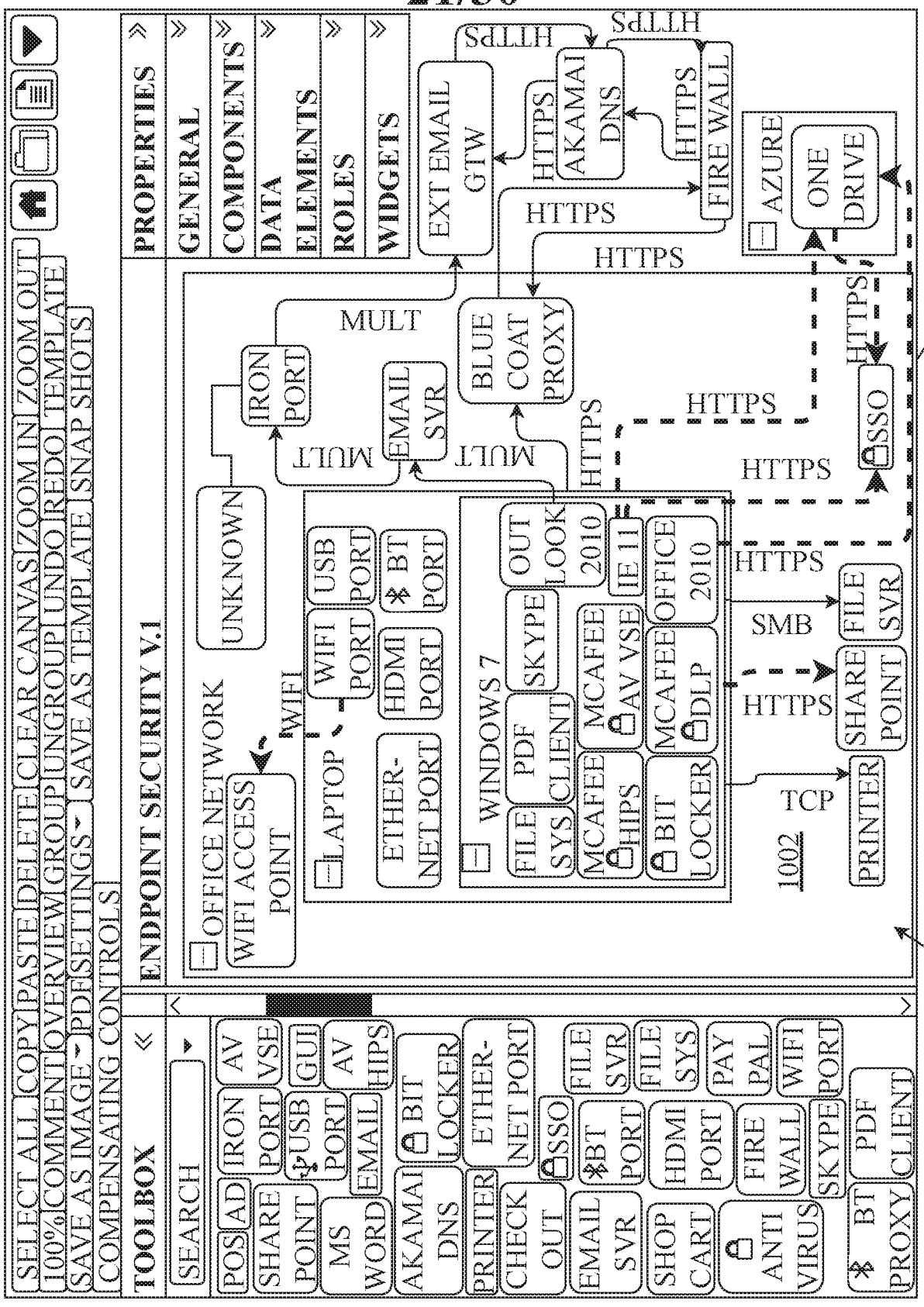

SELECT ALL
COPY
PASTE
DELETE
CLEAR CANVAS
ZOOM IN
ZOOM OUT
100%
COMMENT OVERVIEW
GROUP
UNGROUP
UNDO
REDO
TEMPLATE
SAVE AS IMAGE
PDFSETTINGS
SAVE AS TEMPLATE
SNAP SHOTS
COMPENSATING CONTROLS
TOOLBOX
SEARCH
POS
AD
IRON PORT
AV VSE
SHARE POINT
USB PORT
GUI
MS WORD
EMAIL
AV HIPS
AKAMAI DNS
BIT LOCKER
ETHER-NET PORT
PRINTER
CHECK OUT
SSO
FILE SVR
EMAIL SVR
BT PORT
FILE SYS
SHOP CART
HDMI PORT
PAY PAL
ANTI VIRUS
FIRE WALL
WIFI PORT
BT PROXY
SKYPE
PDF CLIENT
ENDPOINT SECURITY V.1
OFFICE NETWORK
WIFI ACCESS POINT
UNKNOWN
WIFI
LAPTOP
WIFI PORT
USB PORT
HDMI PORT
BT PORT
ETHER-NET PORT
IRON PORT
MULT
EMAIL SVR
WINDOWS 7
FILE SYS
PDF CLIENT
SKYPE
OUT LOOK 2010
MCAFEE HIPS
MCAFEE AV VSE
IE 11
BIT LOCKER
MCAFEE DLP
OFFICE 2010
BLUE COAT PROXY
HTTPS
EXT EMAIL GTW
AKAMAI DNS
FIRE WALL
AZURE
ONE DRIVE
SSO
SMB
FILE SVR
SHARE POINT
TCP
PRINTER
1002
1004
1000
PROPERTIES
GENERAL
COMPONENTS
DATA ELEMENTS
ROLES
WIDGETS

FIG. 21

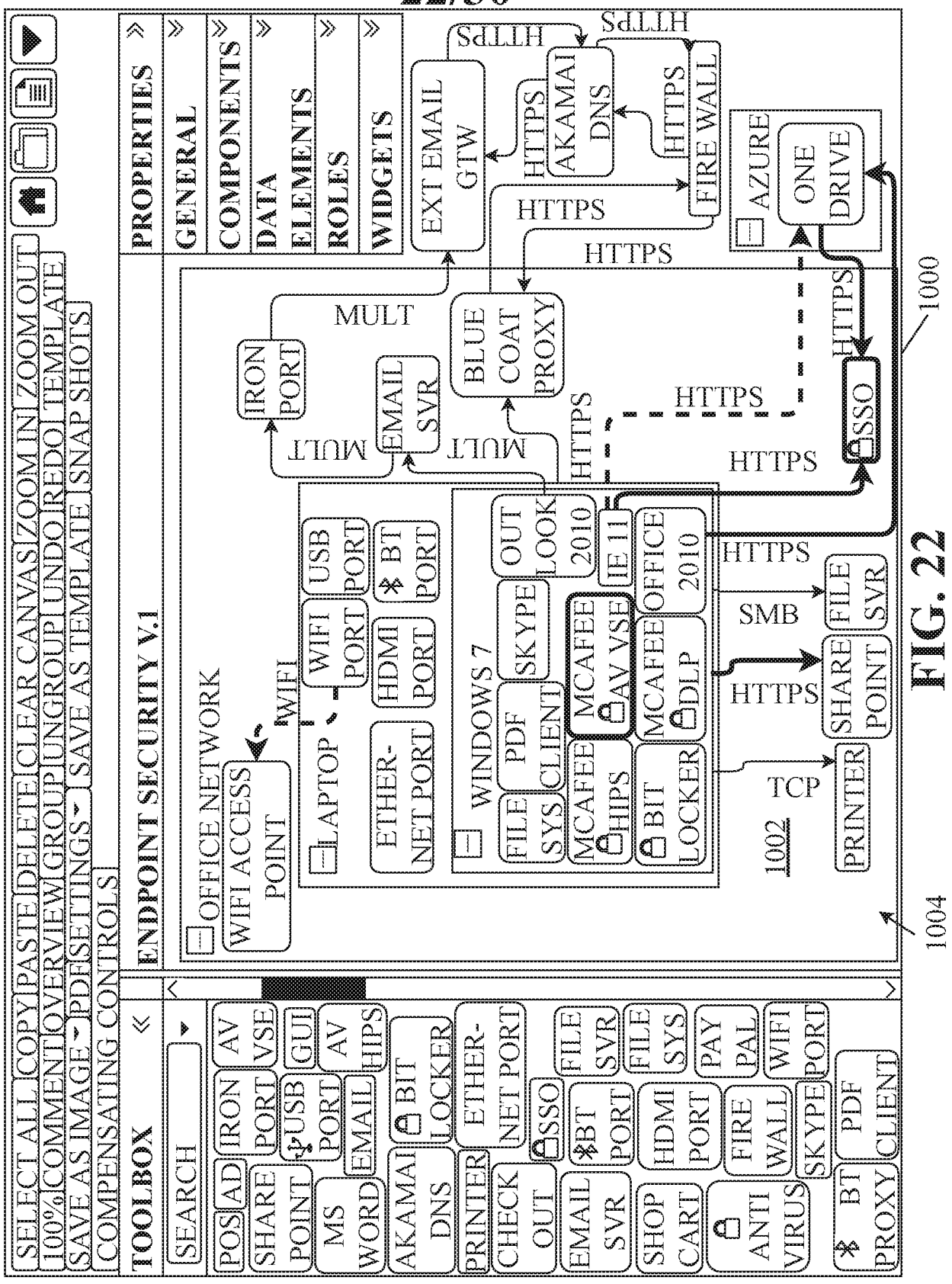

SELECT ALL
COPY
PASTE
DELETE
CLEAR CANVAS
ZOOM IN
ZOOM OUT
100%
COMMENT OVERVIEW
GROUP
UNGROUP
UNDO
REDO
TEMPLATE
SAVE AS IMAGE
PDFSETTINGS
SAVE AS TEMPLATE
SNAP SHOTS
COMPENSATING CONTROLS
TOOLBOX
SEARCH
POS
AD
IRON PORT
AV VSE
SHARE POINT
USB PORT
GUI
MS WORD
EMAIL
AV HIPS
AKAMAI DNS
BIT LOCKER
ETHER-NET PORT
PRINTER
CHECK OUT
SSO
FILE SVR
EMAIL SVR
BT PORT
FILE SYS
SHOP CART
HDMI PORT
PAY PAL
ANTI VIRUS
FIRE WALL
WIFI PORT
BT PROXY
SKYPE
PDF CLIENT
ENDPOINT SECURITY V.1
PROPERTIES
GENERAL
COMPONENTS
DATA ELEMENTS
ROLES
WIDGETS
OFFICE NETWORK
WIFI ACCESS POINT
WIFI
LAPTOP
USB PORT
WIFI PORT
BT PORT
HDMI PORT
ETHER-NET PORT
WINDOWS 7
OUT LOOK 2010
SKYPE
PDF CLIENT
FILE SYS
IE 11
MCAFEE AV VSE
MCAFEE HIPS
OFFICE 2010
MCAFEE DLP
BIT LOCKER
IRON PORT
EMAIL SVR
BLUE COAT PROXY
EXT EMAIL GTW
AKAMAI DNS
FIRE WALL
AZURE
ONE DRIVE
SSO
FILE SVR
SHARE POINT
PRINTER
1002
MULT
HTTPS
SMB
TCP
1000
1004

FIG. 22

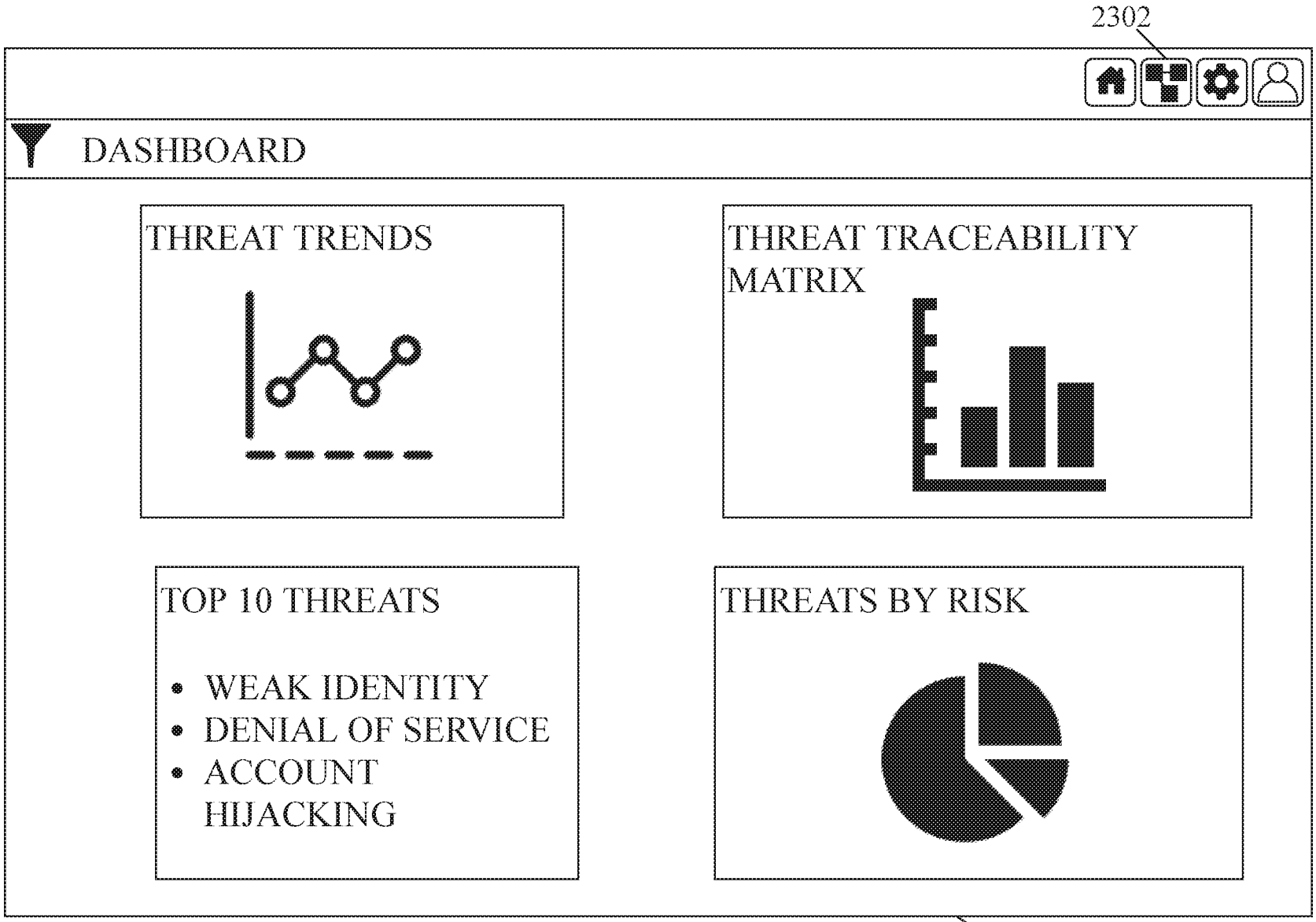


FIG. 23

IriusRisk
Ex. 1002 - Page 107

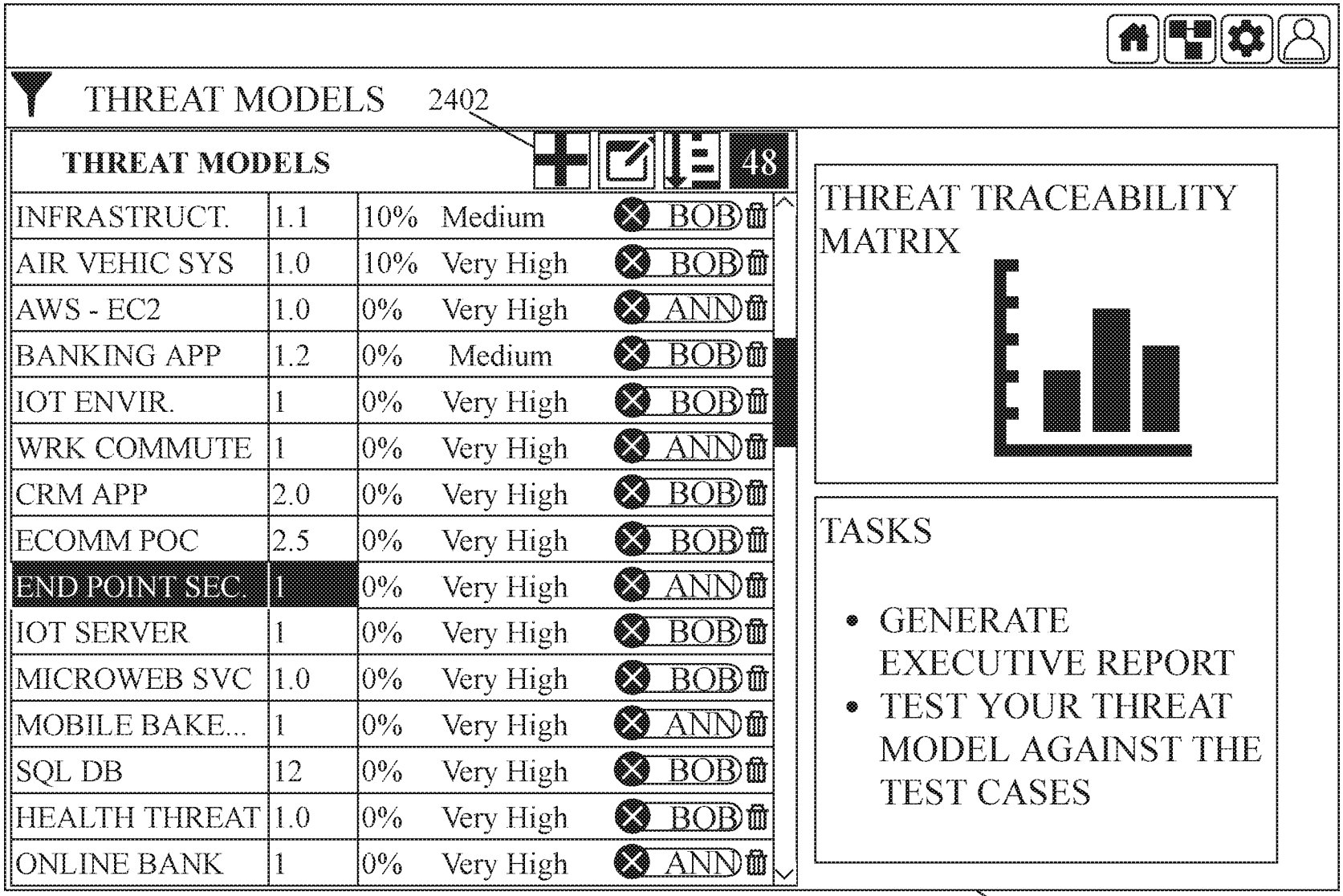


FIG. 24

IriusRisk
Ex. 1002 - Page 108

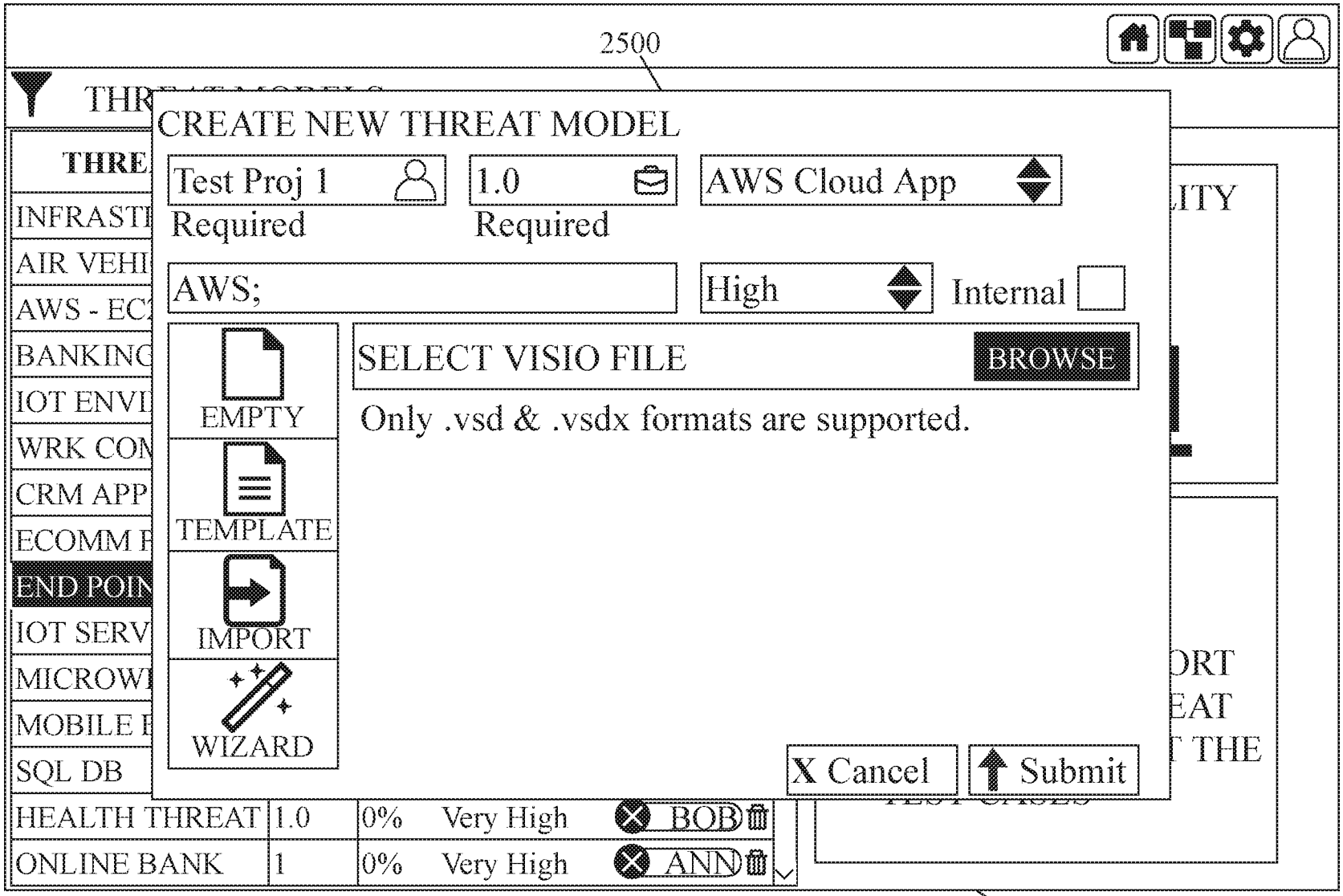


FIG. 25

IriusRisk
Ex. 1002 - Page 109

2500
CREATE NEW THREAT MODEL
Test Proj 1
Required
1.0
Required
AWS Cloud App
AWS;
High
Internal
EMPTY
C:\desktop\AWS_2.vsdx
BROWSE
Only .vsd & .vsdx formats are supported.
Please validate a file.
VALIDATE
TEMPLATE
IMPORT
WIZARD
X Cancel
Submit
THREAT
INFRASTR
AIR VEHI
AWS - EC
BANKING
IOT ENVI
WRK CON
CRM APP
ECOMM P
END POIN
IOT SERV
MICROWI
MOBILE P
SQL DB
HEALTH THREAT 1.0 0% Very High BOB
ONLINE BANK 1 0% Very High AND

FIG. 26

2500
CREATE NEW THREAT MODEL
Test Proj 1
Required
1.0
Required
AWS Cloud App
AWS;
High
Internal
EMPTY
C:\desktop\AWS_2.vsdx
BROWSE
Only .vsd & .vsdx formats are supported.
File Validation Successful
VALIDATE
TEMPLATE
IMPORT
WIZARD
X Cancel
Submit
THREAT
INFRASTR
AIR VEHI
AWS - EC
BANKING
IOT ENVI
WRK CON
CRM APP
ECOMM F
END POIN
IOT SERV
MICROWI
MOBILE F
SQL DB
HEALTH THREAT 1.0 0% Very High BOB
ONLINE BANK 1 0% Very High ANN

FIG. 27

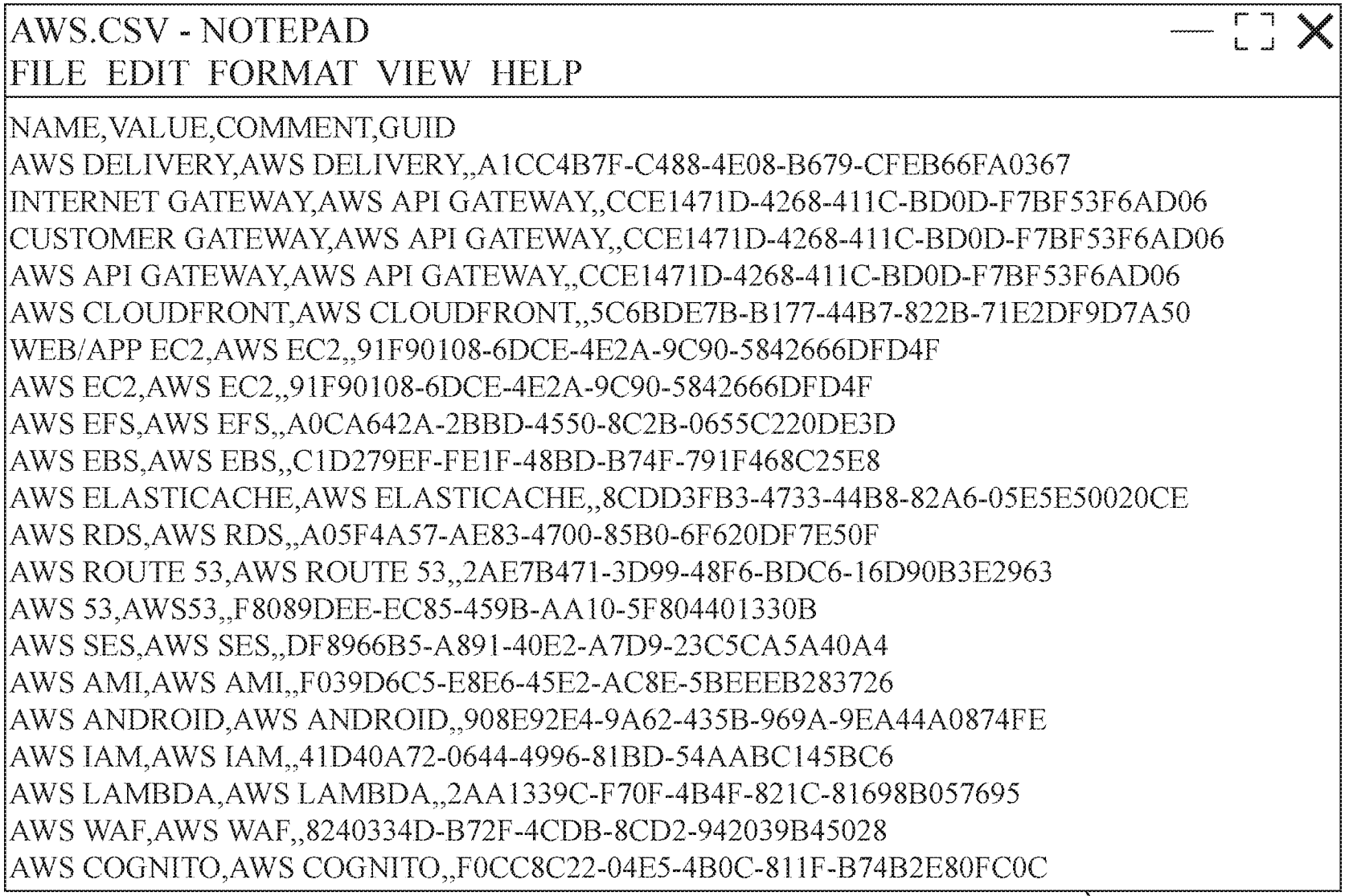


FIG. 28

2800

IriusRisk
Ex. 1002 - Page 112

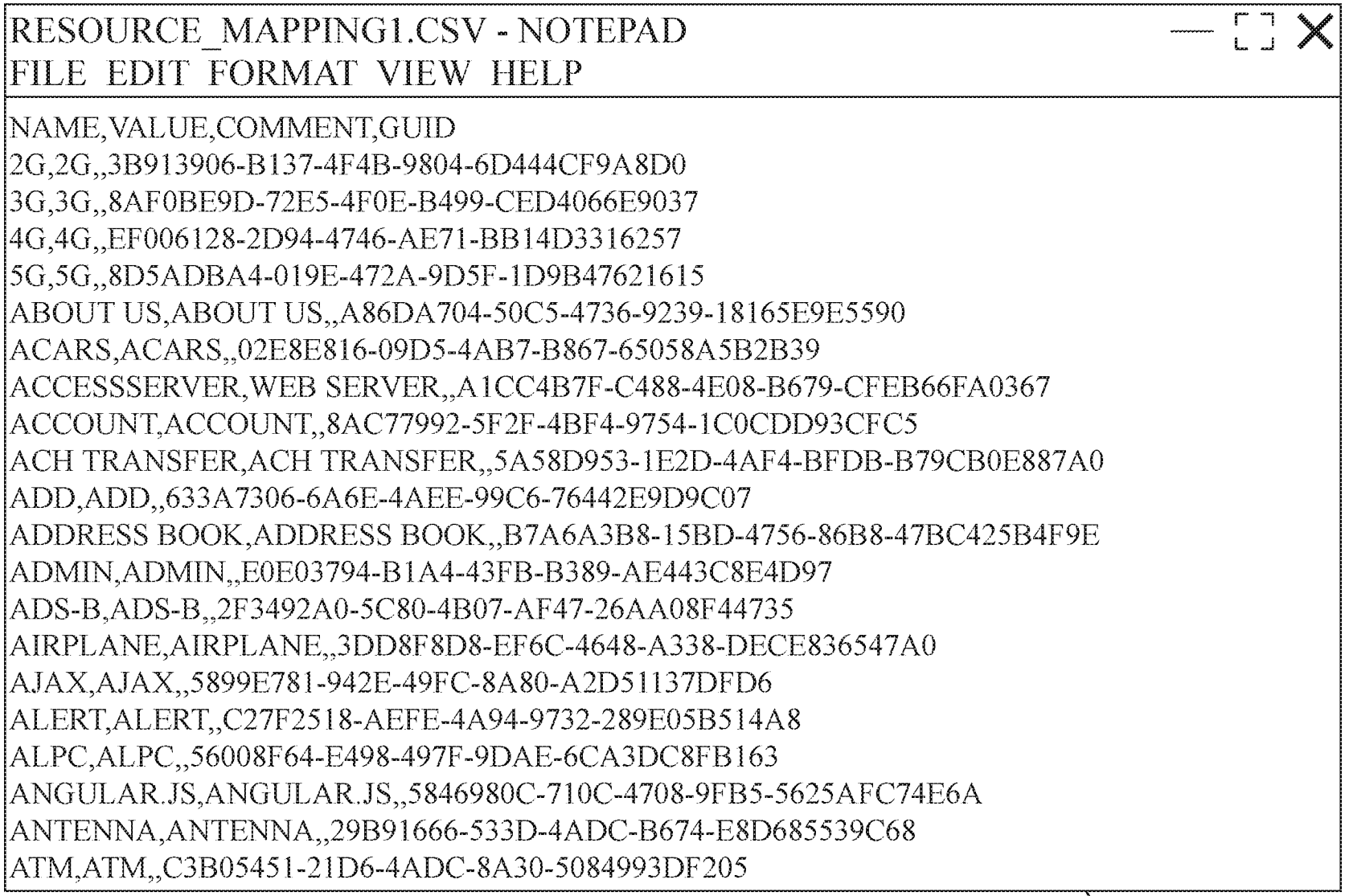


FIG. 29

2900

IriusRisk
Ex. 1002 - Page 113

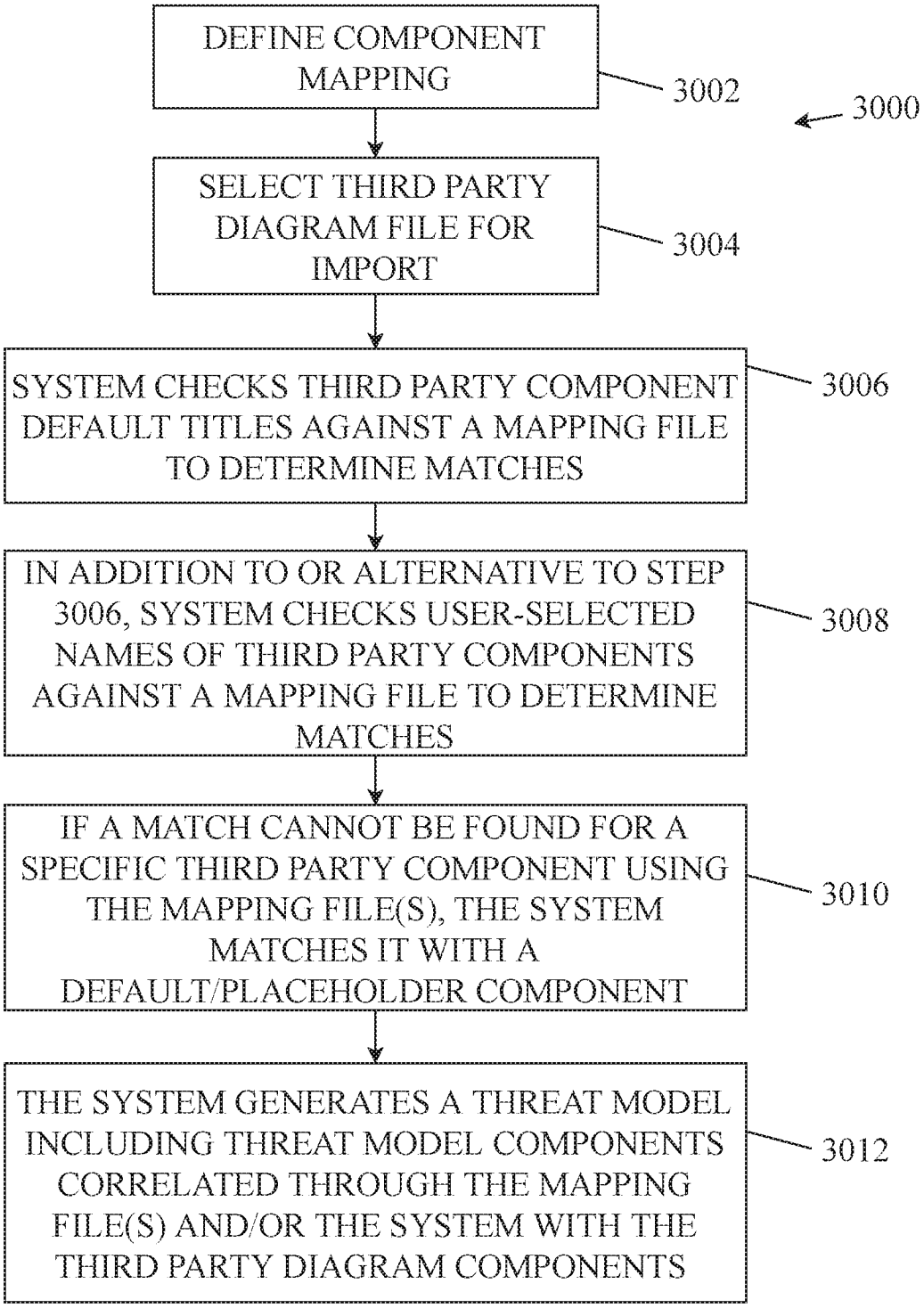

DEFINE COMPONENT MAPPING
3002
3000
SELECT THIRD PARTY DIAGRAM FILE FOR IMPORT
3004
SYSTEM CHECKS THIRD PARTY COMPONENT DEFAULT TITLES AGAINST A MAPPING FILE TO DETERMINE MATCHES
3006
IN ADDITION TO OR ALTERNATIVE TO STEP 3006, SYSTEM CHECKS USER-SELECTED NAMES OF THIRD PARTY COMPONENTS AGAINST A MAPPING FILE TO DETERMINE MATCHES
3008
IF A MATCH CANNOT BE FOUND FOR A SPECIFIC THIRD PARTY COMPONENT USING THE MAPPING FILE(S), THE SYSTEM MATCHES IT WITH A DEFAULT/PLACEHOLDER COMPONENT
3010
THE SYSTEM GENERATES A THREAT MODEL INCLUDING THREAT MODEL COMPONENTS CORRELATED THROUGH THE MAPPING FILE(S) AND/OR THE SYSTEM WITH THE THIRD PARTY DIAGRAM COMPONENTS
3012

FIG. 30

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	
First Named Inventor	Anurag Agarwal
Art Unit	
Examiner Name	
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	7891003	B2	2011-02-15	Mir	
	2	7900259	B2	2011-03-01	Jeschke et al.	
	3	9774613	B2	2017-09-26	Thomas et al.	
	4	4959015		1990-09-25	Rasinski et al.	
	5	7433829	B2	2008-10-07	Borgia et al.	
	6	7096502	B1	2006-08-22	Fox et al.	
	7	8191139	B2	2012-05-29	Heimerdinger et al.	
	8	8255995	B2	2012-08-28	Kraemer et al.	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Anurag Agarwal
	Art Unit	
	Examiner Name	
	Attorney Docket Number	ThreatM010

	9	8413237	B2	2013-04-02	O'Rourke et al.	
	10	9497203	B2	2016-11-15	Honig et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20180324207	A1	2018-11-08	Reybok, Jr. et al.	
	2	20060015941	A1	2006-01-19	McKenna	
	3	20060241991	A1	2006-10-26	Pudhukottai et al.	
	4	20110178942	A1	2011-06-21	Watters et al.	
	5	20140137257	A1	2014-05-15	Martinez et al.	
	6	20110126111	A1	2011-05-26	Gill et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Anurag Agarwal
	Art Unit	
	Examiner Name	
	Attorney Docket Number	ThreatM010

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	Threat Risk Modeling, published online at least as early as Feb. 3, 2018 by OWASP, available at https://www.owasp.org/index.php/Threat_Risk_Modeling.	
	2	Threat Model, Wikipedia, published online at least as early as Feb. 3, 2018, available at https://en.wikipedia.org/wiki/Threat_model (note that some of the material in this Wikipedia article appears to be posted by one or more inventors of the present application or its parent provisional applications - some of the material appears to be posted by Brian Beyst, for example, who is an inventor on two of the parent provisional applications).	
	3	Comparisons of Threat Model Modeling Methodologies, published online by ThreatModeler at least as early as Apr. 15, 2016, available at http://threatmodeler.com/comparison-threat-modeling-methodologies/ (note that this article appears to be affiliated with a business organization of, and may have been published under the direction of, one or more inventors of the present application or its parent provisional applications).	
	4	Microsoft Threat Modeling Tool, published online by Microsoft, different sections published at different times from Aug. 22, 2017 to Jan. 24, 2018, available online at https://docs.microsoft.com/en-us/azure/opbuildpdf/security/TOC.pdf?branch=live	
	5	A description of an on-sale version of systems and methods, on sale in the U.S. by applicants at least as early as 12/31/2011, which on-sale version disclosed some of the elements disclosed in the present application. The description includes a detailed discussion of which claim elements of the claims as originally filed in the present application were disclosed in the 2011 version and which were not, and includes screenshots of user interfaces used in the system/methods of the 2011 version.	
	6	Lockheed Martin Corporation, "Seven Ways to Apply the Cyber Kill Chain with a Threat Intelligence Platform," published at least as early as 2015.	

If you wish to add additional non-patent literature document citation information please click the Add button Add

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Anurag Agarwal
	Art Unit	
	Examiner Name	
	Attorney Docket Number	ThreatM010

EXAMINER SIGNATURE			
Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Anurag Agarwal
	Art Unit	
	Examiner Name	
	Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2019-05-31
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

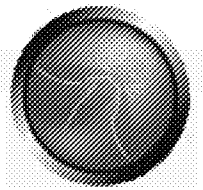
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.



Threat Risk Modeling

Home
About OWASP
Acknowledgements
Advertising
AppSec Events
Books
Brand Resources
Chapters
Donate to OWASP
Downloads
Funding
Governance
Initiatives
Mailing Lists
Membership
Merchandise
News
Community portal
Presentations
Press
Projects
Video
Volunteer

Reference
Activities
Attacks
Code Snippets
Controls
Glossary
How To...
Java Project
.NET Project
Principles

1 Threat Risk Modeling
2 Performing threat risk modeling using the Microsoft Threat Modeling Process
2.1 Identify Security Objectives
2.2 Application Overview
2.3 Decompose Application
2.4 Identify Threats
2.5 STRIDE
2.6 DREAD
3 Alternative Threat Modeling Systems
3.1 Trike
3.2 AS/NZS 4360:2004 Risk Management
3.3 CVSS
3.4 OCTAVE
4 ThreatModel SDK
5 Conclusion
6 Further Reading
7 Appendix: Alternative open-source Risk Management tools
8 Reference

When you start a web application design, it is essential to apply threat modeling; otherwise you will squander resources, time, and money on useless controls that fail to focus on the real threats. There are multiple approaches to threat modeling, as listed below:

- Software centric threat modeling
- Security centric threat modeling
- Asset or risk centric threat modeling. Below represents a mixture of Threat Modeling tools and industry references. The method used to assess risk is not nearly as important as actually performing a structured threat risk modeling. Microsoft notes that the single most important factor in their security improvement program was the corporate adoption of threat risk modeling. One of many considerations is Microsoft's threat modeling process. It is simple to adopt by designers, developers, code reviewers, and the quality assurance team. The following sections provide some overview information (or see Section 6.9,

Ex. 1002 - Page 121 IriusRisk

Technologies
Threat Agents
Vulnerabilities

Tools
What links here
Related changes
Special pages
Printable version
Permanent link
Page information

Further Reading, for additional resources).

Threat Risk Modeling

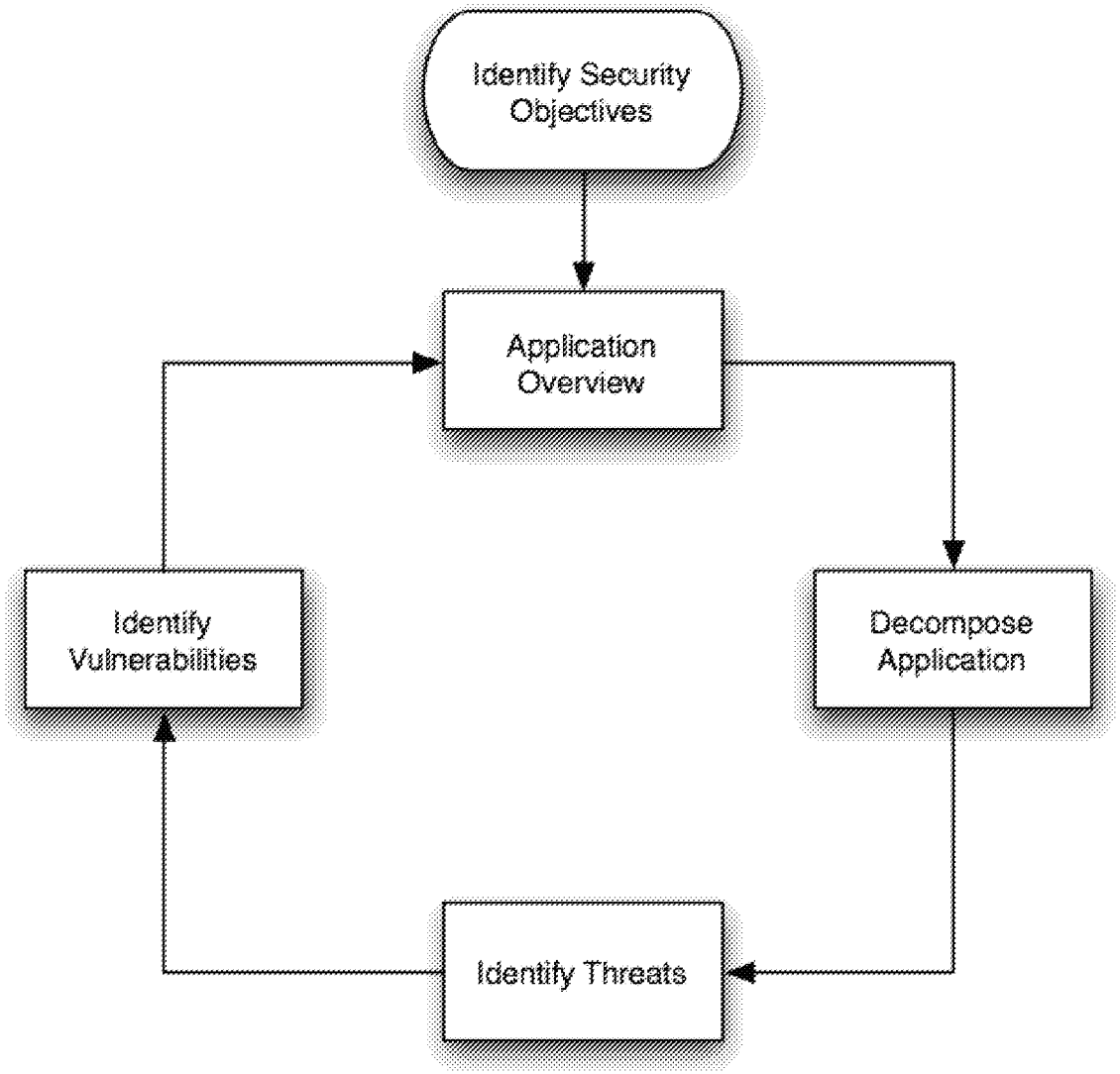
Threat risk modeling is an essential process for secure web application development. It allows organizations to determine the correct controls and to produce effective countermeasures within budget. For example, there is little point in spending $100,000 for fraud control for a system that has negligible fraud risk.

Performing threat risk modeling using the Microsoft Threat Modeling Process

The threat risk modeling process has five steps, enumerated below and shown graphically in Figure 1. They are:

1. Identify Security Objectives
2. Survey the Application
3. Decompose it
4. Identify Threats
5. Identify Vulnerabilities

IriusRisk
Ex. 1002 - Page 122



Let's consider the steps in more detail.

Identify Security Objectives

The business (or project management) leadership, in concert with the software development and quality assurance teams, all need to understand the security objectives. To facilitate this, start by breaking down the application's security objectives into the following categories:

- **Identity:** Does the application protect user identity from abuse? Are there adequate controls in place to ensure evidence of identity (as required for many banking applications?)

Ex. 1002 - Page 123 IriusRisk

- **Financial:** Assess the level of risk the organization is prepared to absorb in remediation, as a potential financial loss. For example, forum software may have a lower estimated financial risk than an Internet banking application.
- **Reputation:** Quantify or estimate of the loss of reputation derived from the application being misused or successfully attacked.
- **Privacy and Regulatory:** To what extent will the application have to protect user data? Forum software by its nature is public, but a tax preparation application is subject to tax regulations and privacy legislation requirements in most countries.
- **Availability Guarantees:** Is the application required to be available per a ***Service Level Agreement (SLA)*** or similar guarantee? Is it a nationally protected infrastructure? To what level will the application have to be available? High availability techniques are significantly more expensive, so applying the correct controls up front will save a great deal of time, resources, and money.

This is by no means an exhaustive list, but it gives an idea of some of the business risk decisions leading into selecting and building security controls.

Other sources of risk guidance come from:

- Laws (such as privacy or finance laws)
- Regulations (such as banking or e-commerce regulations)
- Standards (such as ISO/IEC 27001)
- Legal Agreements (such as payment card industry standards or merchant agreements)
- Corporate Information Security Policy

Application Overview

Once the security objectives have been defined, analyze the application design to identify the ***components***, ***data flows***, and ***trust boundaries***.

Do this by surveying the application's architecture and design documentation. In particular, look for UML component diagrams. Such high level component diagrams are generally sufficient to understand how and why data flows to various places. For example, data movement across a trust boundary (such as from the Internet to the web tier, or from the business logic to the database server), needs to be carefully analyzed, whereas data that flows within the same trust level does not need as much scrutiny.

Decompose Application

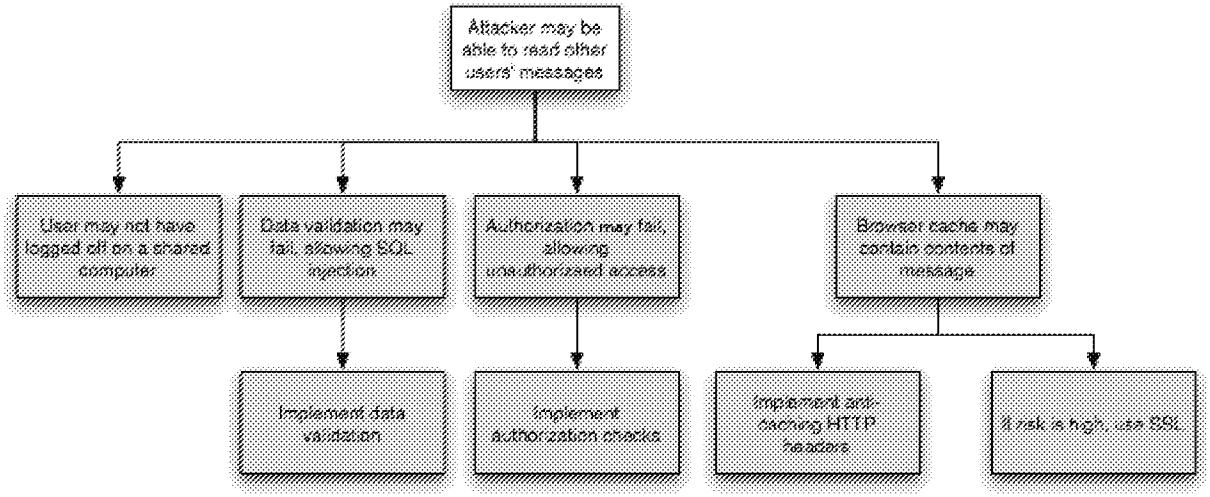
Once the application architecture is understood then decompose it further, to identify the features and modules with a security impact that need to be evaluated. For example, when investigating the authentication module, it is necessary to understand how data enters the module, how the module validates and processes the data, where the data flows, how the data is stored, and what fundamental decisions and assumptions are made by the module.

Identify Threats

Ex. 1002 - Page 124 IriusRisk

It is impossible to write down unknown threats, but it is likewise unlikely that new malware will be created to exploit new vulnerabilities within custom systems. Therefore, concentrate on known risks, which can be easily demonstrated using tools or techniques from Bugtraq.

Microsoft suggests two different approaches for writing up threats. One is a threat graph, as shown in Figure 2, and the other is a structured list.



Typically, a threat graph imparts more information quickly but it takes longer to construct, while a structured list is easier to create but it will take longer for the threat impacts to become obvious.

1. Attacker may be able to read other user's messages
2. User may not have logged off on a shared PC
3. Data validation may allow SQL injection
4. Implement data validation
5. Authorization may fail, allowing unauthorized access
6. Implement authorization checks
7. Browser cache may contain contents of message
8. Implement anti-caching directive in HTTP headers
9. If eavesdropping risk is high, use SSL

Note that it takes a motivated attacker to exploit a threat; they generally want something from your application or to obviate controls. To understand the relevant threats, use the following categories to understand who might attack the application:

- **Accidental Discovery:** An ordinary user stumbles across a functional mistake in your application, just using a web browser, and gains access to privileged information or functionality.
- **Automated Malware:** Programs or scripts, which are searching for known vulnerabilities, and then report them back to a central collection site.
- **The Curious Attacker:** a security researcher or ordinary user, who notices something wrong with the application, and

Ex. 1002 - Page 125 IriusRisk

decides to pursue further.

- **Script Kiddies:** Common renegades, seeking to compromise or deface applications for collateral gain, notoriety, or a political agenda, perhaps using the attack categories described in the *OWASP Web Application Penetration Checklist.*
- **The Motivated Attacker:** Potentially, a disgruntled staff member with inside knowledge or a paid professional attacker.
- **Organized Crime:** Criminals seeking high stake payouts, such as cracking e-commerce or corporate banking applications, for financial gain.

It is vital to understand the level of attacker you are defending against. For example, a motivated attacker, who understands your internal processes is often more dangerous than script kiddies.

STRIDE

STRIDE is a classification scheme for characterizing known threats according to the kinds of exploit that are used (or motivation of the attacker). The STRIDE acronym is formed from the first letter of each of the following categories.

Spoofing Identity "Identity spoofing" is a key risk for applications that have many users but provide a single execution context at the application and database level. In particular, users should not be able to become any other user or assume the attributes of another user.

Tampering with Data Users can potentially change data delivered to them, return it, and thereby potentially manipulate client-side validation, GET and POST results, cookies, HTTP headers, and so forth. The application should not send data to the user, such as interest rates or periods, which are obtainable only from within the application itself. The application should also carefully check data received from the user and validate that it is sane and applicable before storing or using it.

Repudiation Users may dispute transactions if there is insufficient auditing or recordkeeping of their activity. For example, if a user says, "But I didn't transfer any money to this external account!", and you cannot track his/her activities through the application, then it is extremely likely that the transaction will have to be written off as a loss.

Therefore, consider if the application requires non-repudiation controls, such as web access logs, audit trails at each tier, or the same user context from top to bottom. Preferably, the application should run with the user's privileges, not more, but this may not be possible with many off-the-shelf application frameworks.

Information Disclosure Users are rightfully wary of submitting private details to a system. If it is possible for an attacker to publicly reveal user data at large, whether anonymously or as an authorized user, there will be an immediate loss of confidence and a substantial period of reputation loss. Therefore, applications must include strong controls to prevent user ID tampering and abuse, particularly if they use a single context to run the entire application.

Also, consider if the user's web browser may leak information. Some web browsers may ignore the no caching directives in HTTP headers or handle them incorrectly. In a corresponding fashion, every secure application has a responsibility to minimize the amount of information stored by the web browser, just in case it leaks or leaves information behind, which can be used by an attacker to learn details about the application, the user, or to potentially become that user.

Finally, in implementing persistent values, keep in mind that the use of hidden fields is insecure by nature. Such storage should not be relied on to secure sensitive information or to provide adequate personal privacy safeguards.

Ex. 1002 - Page 126 IriusRisk

Denial of Service Application designers should be aware that their applications may be subject to a denial of service attack. Therefore, the use of expensive resources such as large files, complex calculations, heavy-duty searches, or long queries should be reserved for authenticated and authorized users, and not available to anonymous users.

For applications that do not have this luxury, every facet of the application should be engineered to perform as little work as possible, to use fast and few database queries, to avoid exposing large files or unique links per user, in order to prevent simple denial of service attacks.

Elevation of Privilege If an application provides distinct user and administrative roles, then it is vital to ensure that the user cannot elevate his/her role to a higher privilege one. In particular, simply not displaying privileged role links is insufficient. Instead, all actions should be gated through an authorization matrix, to ensure that only the permitted roles can access privileged functionality.

DREAD

DREAD is a classification scheme for quantifying, comparing and prioritizing the amount of risk presented by each evaluated threat. The DREAD acronym is formed from the first letter of each category below.

DREAD modeling influences the thinking behind setting the risk rating, and is also used directly to sort the risks. The DREAD algorithm, shown below, is used to compute a risk value, which is an average of all five categories.

Risk_DREAD = (DAMAGE + REPRODUCIBILITY + EXPLOITABILITY + AFFECTED USERS + DISCOVERABILITY) / 5

The calculation always produces a number between 0 and 10; the higher the number, the more serious the risk.

Here are some examples of how to quantify the DREAD categories.

Damage Potential

- If a threat exploit occurs, how much damage will be caused?
  - 0 = Nothing
  - 5 = Individual user data is compromised or affected.
  - 10 = Complete system or data destruction

Reproducibility

- How easy is it to reproduce the threat exploit?
  - 0 = Very hard or impossible, even for administrators of the application.
  - 5 = One or two steps required, may need to be an authorized user.
  - 10 = Just a web browser and the address bar is sufficient, without authentication.

Exploitability

- What is needed to exploit this threat?
  - 0 = Advanced programming and networking knowledge, with custom or advanced attack tools.

Ex. 1002 - Page 127
IriusRisk

- 5 = Malware exists on the Internet, or an exploit is easily performed, using available attack tools.
- 10 = Just a web browser

Affected Users

- How many users will be affected?
  - 0 = None
  - 5 = Some users, but not all
  - 10 = All users

Discoverability

- How easy is it to discover this threat?
  - 0 = Very hard to impossible; requires source code or administrative access.
  - 5 = Can figure it out by guessing or by monitoring network traces.
  - 9 = Details of faults like this are already in the public domain and can be easily discovered using a search engine.
  - 10 = The information is visible in the web browser address bar or in a form.

Note: When performing a security review of an existing application, "Discoverability" will often be set to 10 by convention, as it is assumed the threat issues will be discovered.

Note: Using DREAD can be difficult at first. It may be helpful to think of Damage Potential and Affected Users in terms of Impact, while thinking of Reproducibility, Exploitability, and Discoverability in terms of Probability. Using the Impact vs Probability approach (which follows best practices such as defined in NIST-800-30), I would alter the formula to make the Impact score equal to the Probability score. Otherwise the probability scores have more weight in the total.

Alternative Threat Modeling Systems

OWASP recognizes that the adoption of the Microsoft modeling process may not fit all organizations. If STRIDE and DREAD are unacceptable for some reason, we recommend that your organization "dry run" the other threat risk models discussed against an existing application or design. This will allow you to determine which approach works best for you, and to adopt the most appropriate threat modeling tools for your organization.

In summary, performing threat modeling provides a far greater return than most any other control in this Guide. Therefore, make threat risk modeling an early priority in your application design process.

Trike

Trike is a threat modeling framework with similarities to the Microsoft threat modeling processes. However, Trike differs because it uses a risk based approach with distinct implementation, threat, and risk models, instead of using the STRIDE/DREAD aggregated threat model (attacks, threats, and weaknesses). From the Trike paper, Trike's goals are:

- With assistance from the system stakeholders, to ensure that the risk this system entails to each asset is acceptable to all stakeholders.

Ex. 1002 - Page 128
IriusRisk

- Be able to tell whether we have done this.
- Communicate what we've done and its effects to the stakeholders.
- Empower stakeholders to understand and reduce the risks to them and other stakeholders implied by their actions within their domains.

For more information on Trike, please see Section 6.9, reference 8.

AS/NZS 4360:2004 Risk Management

The Australian/New Zealand Standard AS/NZS 4360, first issued in 1999, and revised in 2004, is the world's first formal standard for documenting and managing risk and is still one of the few formal standards for managing it. The standard's approach is simple (it's only 28 pages long), flexible, and iterative. Furthermore, it does not lock organizations into a particular risk management methodology, provided the methodology fulfils the AS/NZS 4360 five steps. It also provides several sets of risk tables as examples, and allows organizations to freely develop and adopt their own.

The five steps of the AS/NZS 4360 process are:

- **Establish Context:** Establish the risk domain, i.e., which assets/systems are important?
- **Identify the Risks:** Within the risk domain, what specific risks are apparent?
- **Analyze the Risks:** Look at the risks and determine if there are any supporting controls in place.
- **Evaluate the Risks:** Determine the residual risk.
- **Treat the Risks:** Describe the method to treat the risks so that risks selected by the business will be mitigated.

AS/NZS 4360 assumes that risk will be managed by an ***operational risk group***, and that the organization has adequate skills and risk management resources in house to identify, analyze, and treat the risks.

The advantages of AS/NZS 4360:

- AS/NZS 4360 works well as a risk management methodology for organizations requiring Sarbanes-Oxley compliance.
- AS/NZS 4360 works well for organizations that prefer to manage risks in a traditional way, such as just using likelihood and consequence to determine an overall risk.
- AS/NZS 4360 is familiar to most risk managers worldwide, and your organization may already have implemented an AS/NZS 4360 compatible approach.
- You are an Australian organization, and may be required to use it if you are audited on a regular basis, or to justify why you aren't using it. Luckily, the STRIDE/DREAD model discussed earlier is AS/NZS 4360 compatible.

The limitations of AS/NZS 4360:

- The AS/NZS 4360 approach works best for business or systemic risks than for technical risks.
- AS/NZS 4360 does not define the methodology to perform a structured threat risk modeling exercise.
- As AS/NZS 4360 is a generic framework for managing risk, it does not provide any structured method to enumerate web application security risks.

Although AS/NZS 4360 may be used to rank risks for security reviews, the lack of structured methods of enumerating threats

Ex. 1002 - Page 129
IriusRisk

for web applications makes it less desirable than other methodologies described earlier.

CVSS

The US Department of Homeland Security (DHS) established the NIAC Vulnerability Disclosure Working Group, which incorporates input from Cisco Systems, Symantec, ISS, Qualys, Microsoft, CERT/CC, and eBay. One of the group's outputs is the ***Common Vulnerability Scoring System (CVSS).***

The advantages of CVSS:

- You have just received notification from a security researcher or other source that your product has vulnerability, and you wish to ensure that it has an accurate and normalized severity rating, so as to alert your customers to the appropriate level of action required when you release the patch.
- You are a security researcher, and have found several threat exploits within an application. You would like to use the CVSS ranking system to produce reliable risk rankings, to ensure that the ISV will take the exploits seriously as indicated by their rating.
- CVSS has been recommended by the working group for use by US Government departments. However, it is unclear if it will become policy or be widely adopted at the time of this writing.

The limitations of CVSS:

- CVSS does not find or reduce the attack surface area (i.e. design flaws), or help enumerate risks within any arbitrary piece of code, as it is just a scoring system, not a modeling methodology.
- CVSS is more complex than STRIDE/DREAD, as it aims to calculate the risk of announced vulnerabilities as applied to deployed software and environmental factors.
- The CVSS risk ranking is complex -- a spreadsheet is required to calculate the risk components as the assumption behind CVSS is that a specific vulnerability has been identified and announced, or a worm or Trojan has been released targeting a small number of attack vectors.
- The overhead of calculating the CVSS risk ranking is quite high if applied to a thorough code review, which may have 250 or more threats to rank.

OCTAVE

OCTAVE is a heavyweight risk methodology approach originating from Carnegie Mellon University's Software Engineering Institute (SEI) in collaboration with CERT. OCTAVE focuses on organizational risk, not technical risk. OCTAVE comes in two versions: Full OCTAVE, for large organizations, and OCTAVE-S for small organizations, both of which have specific catalogs of practices, profiles, and worksheets to document the modeling outcomes.

OCTAVE is popular with many sites and is useful when:

- Implementing an organizational culture of risk management and controls becomes necessary.
- Documenting and measuring business risk becomes timely.
- Documenting and measuring the overall IT security risk, particularly as it relates to the corporate IT risk management,

Ex. 1002 - Page 130 IriusRisk

becomes necessary.
- When documenting risks surrounding complete systems becomes necessary.
- To accommodate a fundamental reorganization, such as when an organization does not have a working risk methodology in place, and requires a robust risk management framework to be put in place.

The limitations of OCTAVE are:

- OCTAVE is incompatible with AS/NZS 4360, as it mandates Likelihood = 1 (i.e., It assumes a threat will always occur) and this is inappropriate for many organizations. OCTAVE-S makes the inclusion of this probability optional, but this is not part of the more comprehensive OCTAVE standard.
- Consisting of 18 volumes, OCTAVE is large and complex, with many worksheets and practices to implement.
- It does not provide a list of "out of the box" practices for assessing and mitigating web application security risks.

Because of these issues, OWASP does not anticipate that OCTAVE will be used at large by application designers or developers, because it fails to take threat risk modeling into consideration, which is useful during all stages of development, by all participants, to reduce the overall risk of an application becoming vulnerable to attack.

ThreatModel SDK

The ThreatModel SDK is a minimalistic Java library that provides a basic vendor-neutral object model along with the ability to parse reports generated from common threat modeling tools. Supported Threat Modeling Tools:

- Microsoft Threat Modeling Tool 2016

Planned Threat Modeling Tools:

- Mozilla SeaSponge

For more information visit: *https://github.com/stevespringett/threatmodel-sdk*

Conclusion

In this chapter, we have touched on the basic principles of threat risk modeling, risk management, and web application security. Applications that leverage the underlying intent of these principles will be more secure than their counterparts, which will only be minimally compliant just by including specific controls.

Further Reading

- *Threat Analysis & Modeling v2.1.2*, © Microsoft Corporation, 2007.
- *Threat Modeling Web Applications*, J.D. Meier, Alex Mackman, Blaine Wastell, © Microsoft Corporation, May 2005.
- *Improving Web Application Security: Threats and Countermeasures*, J.D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla and Anandha Murukan, © Microsoft Corporation, June 2003.
- *Threat Modeling*, Frank Swiderski and Window Snyder, Microsoft Press, June 2004, ISBN 0-7356-1991-3.

Ex. 1002 - Page 131 IriusRisk

- Writing Secure Code, 2nd Edition, Howard and LeBlanc, (pp. 69 – 124), Microsoft Press, 2003, ISBN 0-7356-1722-8.
- The STRIDE Threat Model, © Microsoft Corporation, 2005.
- DREADful - the DREAD system, © Microsoft Corporation, 2005.
- A Conceptual Model for Threat Modeling Applications, Saitta, Larcom, and Michael Eddington, July 2005, http://dymaxion.org/trike/.
- AS/NZS 4360:2004 Risk Management, Standards Australia and Standards New Zealand.
- CVSS, U.S. Department of Homeland Security library, February 2005.
- OCTAVE, CERT library.

Appendix: Alternative open-source Risk Management tools

- OSMR
- MARCO
- CORAS Risk Assessment Platform
- ISO 17799 Risk Assessment Toolkit
- Easy Threat Risk Assessment
- ARMS
- Minaccia
- ThreatMind
- Open Source Requirements Management Tool

Reference

Development Guide Table of Contents

Categories: FIXME | *OWASP Guide Project* | Activity | Externally Linked Page | Threat Modeling | SAMM-TA-1

This page was last modified on 13 July 2017, at 04:20.

Content is available under Creative Commons Attribution-ShareAlike unless otherwise noted.

Privacy policy About OWASP Disclaimers

Ex. 1002 - Page 132 IriusRisk

Not logged in Talk Contributions Create account Log in

Article Talk Read Edit View history Search Wikipedia Search

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Français
中文

Threat model

From Wikipedia, the free encyclopedia

Threat modeling is a process by which potential threats, such as structural vulnerabilities can be identified, enumerated, and prioritized – all from a hypothetical attacker's point of view. The purpose of threat modeling is to provide defenders with a systematic analysis of the probable attacker's profile, the most likely attack vectors, and the assets most desired by an attacker. Threat modeling answers the questions "Where are the high-value assets?" "Where am I most vulnerable to attack?" "What are the most relevant threats?" "Is there an attack vector that might go unnoticed?"

Conceptually, most people incorporate some form of threat modeling in their daily life and don't even realize it. Commuters use threat modeling to consider what might go wrong during the morning drive to work and to take preemptive action to avoid possible accidents. Children engage in threat modeling when determining the best path toward an intended goal while avoiding the playground bully. In a more formal sense, threat modeling has been used to prioritize military defensive preparations since antiquity.

Contents

1 Evolution of IT-Based Threat Modeling
2 Threat Modeling Methodologies for IT Purposes
  2.1 STRIDE Methodology
  2.2 P.A.S.T.A.
  2.3 Trike
  2.4 VAST
3 Generally Accepted IT Threat Modeling Processes
  3.1 Visual Representations based on Data Flow Diagrams
  3.2 Visual Representations based on Process Flow Diagrams
4 Threat Modeling Tools
5 Further fields of application
6 References

Evolution of IT-Based Threat Modeling [edit]

Shortly after shared computing made its debut in the early 1960s individuals began seeking ways to exploit security vulnerabilities for personal gain.[1] As a result, engineers and computer scientists soon began developing threat modeling

Ex. 1002 - Page 133 IriusRisk

concepts for information technology systems.

Early IT-based threat modeling methodologies were based on the concept of architectural patterns[2] first presented by Christopher Alexander in 1977. In 1988 Robert Barnard developed and successfully applied the first profile for an IT-system attacker.

In 1994, Edward Amoroso put forth the concept of a "threat tree" in his book, "Fundamentals of Computer Security Technology.[3]" The concept of a threat tree was based on decision tree diagrams. Threat trees graphically represent how a potential threat to an IT system can be exploited.

Independently, similar work was conducted by the NSA and DARPA on a structured graphical representation of how specific attacks against IT-systems could be executed. The resulting representation were called "attack trees." In 1998 Bruce Schneier published his analysis of cyber risks utilizing attack trees in his paper entitled "Toward a Secure System Engineering Methodology.[4]" The paper proved to be a seminal contribution in the evolution of threat modeling for IT-systems. In Schneier's analysis, the attacker's goal is represented as a "root node," with the potential means of reaching the goal represented as "leaf nodes." Utilizing the attack tree in this way allowed cybersecurity professionals to systematically consider multiple attack vectors against any defined target.

In 1999, Microsoft cybersecurity professionals Loren Kohnfelder and Praerit Gard applied Schneier's attack tree analysis to develop a methodical means to consider potential attacks relevant to the Microsoft Windows development environment. The resultant STRIDE[5] threat model (STRIDE is an acronym for Spoofing identity, Tampering with data, Repudiation, Information disclosure, Denial of service, Elevation of privilege) required security professionals to systematically determine how a potential attacker could utilize any threat categorized by the STRIDE acronym at each node of a Schneier-type attack tree.

In 2003, OCTAVE[6] (Operationally Critical Threat, Asset, and Vulnerability Evaluation) method, a risk-based assessment threat modeling methodology, was first introduced with a primary focus on organizational risk management.

In 2004, Frank Swiderski and Window Snyder wrote "Threat Modeling," by Microsoft press. In it they developed the concept of using threat models to create secure applications.

In 2014 Ryan Stillions expressed the idea that cyber threats should be expressed with different semantic levels, and proposed the DML (Detection Maturity Level) model.[7] An attack is an instantiation of a threat scenario which is caused by a specific attacker with a specific goal in mind and a strategy for reaching that goal. The goal and strategy represent the highest semantic levels of the DML model. This is followed by the TTP (Tactics, Techniques and Procedures) which represent intermediate semantic levels. The lowest semantic levels of the DML model are the tools used by the attacker, host and observed network artefacts such as packets and payloads, and finally atomic indicators such as IP addresses at the lowest semantic level. Current SIEM tools typically only provide indicators at the lowest semantic levels. There is therefore a need to develop SIEM tools that can provide threat indicators at higher semantic levels.[8]

Threat Modeling Methodologies for IT Purposes [edit]

Conceptually a threat modeling practice flows from a methodology. Numerous threat modeling methodologies are available for

Ex. 1002 - Page 134 IriusRisk

implementation. Based on volume of published online content, the four methodologies discussed below are the most well known.

STRIDE Methodology [edit]

The STRIDE approach to threat modeling was introduced in 1999 at Microsoft, providing a mnemonic for developers to find 'threats to our products' [9] . STRIDE, Patterns and Practices, and Asset/entry point were amongst the threat modeling approaches developed and published by Microsoft. References to "the" Microsoft methodology commonly mean STRIDE.

P.A.S.T.A. [edit]

The Process for Attack Simulation and Threat Analysis (PASTA) is a seven-step, risk-centric methodology.[10] It provides a seven-step process for aligning business objectives and technical requirements, taking into account compliance issues and business analysis. The intent of the method is to provide a dynamic threat identification, enumeration, and scoring process. Once the threat model is completed security subject matter experts develop a detailed analysis of the identified threats. Finally, appropriate security controls can be enumerated. This methodology is intended to provide an attacker-centric view of the application and infrastructure from which defenders can develop an asset-centric mitigation strategy.

Trike [edit]

The focus of the Trike methodology[11] is using threat models as a risk-management tool. Within this framework, threat models are used to satisfy the security auditing process. Threat models are based on a "requirements model." The requirements model establishes the stakeholder-defined "acceptable" level of risk assigned to each asset class. Analysis of the requirements model yields a threat model form which threats are enumerated and assigned risk values. The completed threat model is used to construct a risk model based on asset, roles, actions, and calculated risk exposure.

VAST [edit]

VAST is an acronym for Visual, Agile, and Simple Threat modeling.[12] The underlying principle of this methodology is the necessity of scaling the threat modeling process across the infrastructure and entire SDLC, and integrating it seamlessly into an Agile software development methodology. The methodology seeks to provide actionable outputs for the unique needs of various stakeholders: application architects and developers, cybersecurity personnel, and senior executives. The methodology provides a unique application and infrastructure visualization scheme such that the creation and use of threat models do not require specific security subject matter expertise.

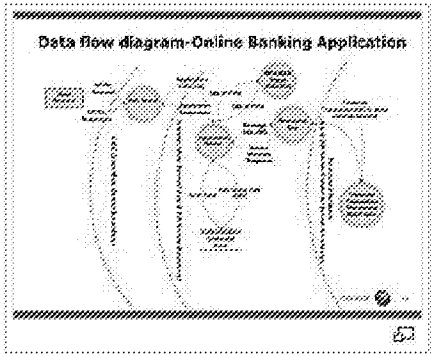
Generally Accepted IT Threat Modeling Processes [edit]

All IT-related threat modeling processes start with creating a visual representation of the application and / or infrastructure being analyzed. The application / infrastructure is decomposed into various elements to aid in the analysis. Once completed, the visual representation is used to identify and enumerate potential threats. Further analysis of the model regarding risks associated with identified threats, prioritization of threats, and enumeration of the appropriate mitigating controls depends on

Ex. 1002 - Page 135
IriusRisk

the methodological basis for the threat model process being utilized.

Visual Representations based on Data Flow Diagrams [edit]

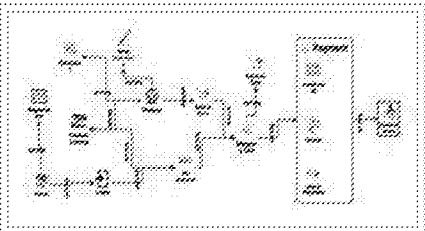


The Microsoft methodology, PASTA, and Trike each develop a visual representation of the application-infrastructure utilizing data flow diagrams (DFD).[13] DFDs were developed in the 1970s as tool for system engineers to communicate, on a high level, how an application caused data to flow, be stored, and manipulated by the infrastructure upon which the application runs. Traditionally, DFDs utilize only four unique symbols: data flows, data stores, processes, and interactors. In the early 2000s, an additional symbol, trust boundaries, were added to allow DFDs to be utilized for threat modeling.

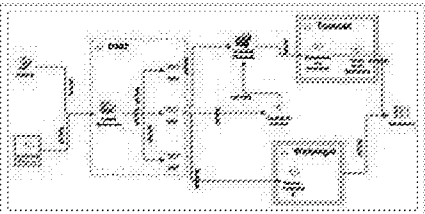
Once the application-infrastructure system is decomposed into its five elements, security experts consider each identified threat entry point against all known threat categories. Once the potential threats are identified, mitigating security controls can be enumerated or additional analysis can be performed.

Visual Representations based on Process Flow Diagrams [edit]

The VAST methodology distinguished between application threat models (ATM) and operational or infrastructure threat models (OTM).[14]



Process Flow Diagram Application Threat Model



End-to-end Data Flow Operational Threat Model

ATMs are built with process flow diagrams (PFD).[13] PFDs were developed in 2011 as a tool to allow Agile software development teams to create threat models based on the application design process. Applications are decomposed into their various features or use cases. Each feature is described in terms of the widgets or code building blocks required to build that feature. Features are then connected by communication protocols. The resulting visualization is a map of how a user moves through the various features of an application.

OTMs are built with end-to-end data flow diagrams very similar to traditional DFDs. End to end data flow diagrams decompose an IT system into its various independent, grouped, and shared components. Each component is described in terms of specific attributes. Components are then connected by communication pathways and protocols.

Once the ATM or OTM is constructed, the methodology specifies how potential threats are identified, enumerated, prioritized, and associated with their relevant risks and mitigating security controls.

[edit]

Threat Modeling Tools

There are currently five tools available for organizational threat modeling:

- Microsoft's free threat modeling tool – the Threat Modeling Tool (formerly SDL Threat Modeling Tool).[15] This tool also utilizes the Microsoft threat modeling methodology, is DFD-based, and identifies threats based on the STRIDE threat classification scheme. It is intended primarily for general use.
- MyAppSecurity offers the first commercially available threat modeling tool - ThreatModeler[16] It utilizes the VAST methodology, is PFD-based, and identifies threats based on a customizable comprehensive threat library.[17] It is intended for collaborative use across all organizational stakeholders.
- IriusRisk offers both a community and a commercial version of the tool. This tool focus on the creation and maintenance of a live Threat Model through the entire SDLC. It drives the process by using fully customizable questionnaires and Risk Pattern Libraries, and connects with other several different tools (OWASP ZAP, BDD-Security, Threadfix...) to empower automation.[18]
- securiCAD is a threat modelling and risk management tool by the Scandinavian company foreseeti. It is intended for company cyber security management, from CISO, to security engineer, to technician. securiCAD conducts automated attack simulations to current and future IT architectures, identifies and quantifies risks holistically including structural vulnerabilities, and provides decision support based on the findings. securiCAD is offered in both commercial and community editions. [19]
- SD Elements by Security Compass is a software security requirements management platform that includes automated threat modeling capabilities. A set of threats is generated by completing a short questionnaire about the technical details and compliance drivers of the application. Countermeasures are included in the form of actionable tasks for developers that can be tracked and managed throughout the entire SDLC.[20]

Further fields of application [edit]

Threat modeling is being applied not only to IT but also to other areas such as building and home automation.[21] In this context, threats to security and privacy like information about the inhabitant's movement profiles, working times, and health situations are modeled as well as physical or network-based attacks. The latter could make use of more and more available smart building features, i.e., sensors (e.g., to spy on the inhabitant) and actuators (e.g., to unlock doors).[21]

References [edit]

1. ^ McMillan, Robert (2012). *"The World's First Computer Password? It Was Useless Too"*. Wired Business.
2. ^ Shostack, Adam (2014). *"Threat Modeling: Designing for Security"*. John Wiley & Sons Inc: Indianapolis.
3. ^ Amoroso, Edward G (1994). *"Fundamentals of Computer Security Technology"*. AT&T Bell Labs. Prentice-Hall: Upper Saddle River.
12. ^ Agarwal, Anurag "Archie," etal. *VAST Methodology: Visual, Agile, and Simple Threat Modeling*. Various Interviews. Transformational Opportunities: Prescott Valley. 2016
13. ^ [a] [b] Agarwal, Anurag "Archie" (2016). *"Threat Modeling – Data Flow Diagram vs Process Flow Diagram"*. *ThreatModeler.com*.

Ex. 1002 - Page 137
IriusRisk

4. ^ Schneier, Bruce; et al. (1998). "Toward A Secure System Engineering Methodology" (PDF). National Security Agency: Washington.
5. ^ "The STRIDE Threat Mode". Microsoft. 2016.
6. ^ Alberts, Christopher (2003). "Introduction to the OCTAVE® Approach" (PDF). Software Engineering Institute, Carnegie Mellon: Pitsburg.
7. ^ Stillions, Ryan (2014). "The DML Model". *Ryan Stillions security blog*. Ryan Stillions.
8. ^ Bromander, Siri (2016). "Semantic Cyberthreat Modelling" (PDF). Semantic Technology for Intelligence, Defence and Security (STIDS 2016).
9. ^ Kohnfelder, Loren; Garg, Praerit. "Threats to Our Products". Microsoft. Retrieved 20 September 2016.
10. ^ Ucedavélez, Tony and Marco M. Morana (2015). "Risk Centric Threat Modeling: Process for Attack Simulation and Threat Analysis". John Wiley & Sons: Hobekin.
11. ^ Eddington, Michael, Brenda Larcom, and Eleanor Saitta (2005). "Trike v1 Methodology Document". *Octotrike.org*.
14. ^ Agarwal, Anurag "Archie" (2016). "Application Threat Modeling vs Operational Threat Modeling". *ThreatModeler.com*.
15. ^ "What's New with Microsoft Threat Modeling Tool 2016". *Microsoft Secure Blog*. Microsoft. 2015.
16. ^ "ThreatModeler Home". *ThreatModeler*.
17. ^ Agarwal, Anurag "Archie," etal. *Comprehensive Threat Library*. Various Interviews. Transformational Opportunities: Prescott Valley. 2016
18. ^ "Irius Risk Risk Management Tool". Continuum Security. 2016.
19. ^ "Cyber Threat Modelling and Risk Management - securiCAD by foreseeti". foreseeti.
20. ^ "SD Elements by Security Compass". *www.securitycompass.com*. Retrieved 2017-03-24.
21. ^ [a] [b] Meyer, D.; Haase, J.; Eckert, M.; Klauer, B. (2016-07-01). "A threat-model for building and home automation". *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*: 860–866. doi:10.1109/INDIN.2016.7819280.

Categories: Computer security | Computer security exploits

This page was last edited on 10 January 2018, at 09:09.

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.

Privacy policy About Wikipedia Disclaimers Contact Wikipedia Developers Cookie statement Mobile view

Comparison of Threat Modeling Methodologies

Posted on *April 15, 2016* by *Reef DSouza*

Microsoft Threat Modeling Methodology: Stride Threat Model

Microsoft's threat modeling practice aligns with their Trustworthy Computing directive of January 2002 and its primary focus is to help ensure secure software begins during the design phase. In order for an application to meet the security properties of Confidentiality, Integrity, and Availability (CIA), along with Authorization, Authentication and Non-Repudiation, Microsoft uses its "STRIDE" methodology. STRIDE is an acronym that defines a threat classification system that stands for Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege. Stride threat model designed by Microsoft.

Data Flow Diagrams (DFD) are used to graphically represent the system and the DFDs use a standard set of symbols consisting of four elements: data flows, data stores, processes, and interactors, and to incorporate threat modeling, trust boundaries are also included. An application is decomposed into individual components and the relevant threats from the STRIDE system can be applied to each component. The mitigating controls for each component are then determined.

From a methodology point of view, scalability with a software development process such as AGILE is a challenge. And, since the Trustworthy Computing directives apply specifically to a Microsoft Windows development environment, it is limited in scope, since it does not address the many other development platforms in use by organizations. Also worth noting is that practitioners of this process require subject matter expertise in order to fully understand the threats, threat classifications, and relevance of security properties to the elements of a threat model. Compare Stride Threat Model with P.A.S.T.A. and Trike below.

Ex. 1002 - Page 139
IriusRisk

P.A.S.T.A (Process for Attack Simulation and Threat Analysis):

This is a new application threat modeling methodology introduced by Marco Morana and Tony "UV." P.A.S.T.A is a seven-step process that is applicable to most application development methodologies and is platform agnostic. It not only aligns business objectives with technical requirements, but also takes into account compliance requirements, business impact analysis, and a dynamic approach to threat management, enumeration, and scoring.

The process begins with a clear definition of business objectives, security and compliance requirements, along with business impact analysis. Similar to the Microsoft process, an application is decomposed into components with use case diagrams and DFDs that are used to illustrate the threat model, from which threat and vulnerability analysis can be performed.

The next step involves the use of threat trees, abuse cases, scoring systems, and enumerations for more detailed analysis. The user can then view a threat model from an attacker's perspective by performing attack modeling in attack trees, along with attack surface analysis. In the final step, risk and business impact can be qualified and quantified, in conjunction with the necessary countermeasures.

This process combines the best of various threat modeling approaches using attack trees to provide an attacker-centric view of threats, in combination with risk and impact analysis. This helps organizations establish an asset-centric approach to planning its mitigation strategy. The use of threat trees to map threats to existing vulnerabilities lends itself to a much easier and scalable threat management process.

Beyond the technical aspect of this approach, the risk and business impact analysis elevates threat modeling from a "software development only" exercise by linking vulnerability management to business relevancy, since it involves the participation of key decision makers in the process. What differentiates this methodology from Trike (see below) is that it includes risk management steps in the final stage of the process to help ensure P.A.S.T.A is not limited to a specific risk calculation formula.

Ex. 1002 - Page 140
IriusRisk

Trike:

Trike is a methodology used to build threat models in order to help satisfy the security auditing process from a risk management perspective. The creators of Trike also provide a tool to accompany this methodology.

The foundation of this methodology is a "requirements model" designed to ensure the level of risk assigned to each asset is classified as "acceptable" by the system's stakeholders. Once the requirements model is in place, DFDs are used to create an implementation model that illustrates data flows, including the actions that can be performed in real time and within a system state.

Finally, the implementation model is analyzed to produce a threat model. Once threats are enumerated, appropriate risk values are assigned to them and attack graphs are created. Mitigating controls are also assigned to the vulnerabilities that lead to these threats. Once this process is completed, a risk model is created based on assets, roles, actions and threat exposure.

While Trike has similarities to the Microsoft threat modeling process, it differs in that it uses a risk- based approach with distinct implementation, threat, and risk models. The Trike methodology appears to be in an experimental stage and with lack of significant documentation and support, it seems difficult to implement. Systems with more than roughly a dozen actors and assets are difficult to model as a single system using Trike v1, as this current version suffers from performance issues, even at these relatively low thresholds.

ThreatModeler, MyAppSecurity's flagship offering, is the industry's first automated, scalable, and repeatable threat modeling product. Please contact us to learn more about ThreatModeler™

Contact us for a ThreatModeler Demo

References:

http://msdn.microsoft.com/en-us/library/ff648006.aspx#tmwa_approach

http://msdn.microsoft.com/en-gb/magazine/cc163519.aspx

http://msdn.microsoft.com/en-us/library/cc307748.aspx

Ex. 1002 - Page 141
IriusRisk

http://securesoftware.blogspot.com/2011/06/attack-simulation-and-threat-analysis.html

http://octotrike.org/papers/

This entry was posted in *Threat Modeling* and tagged *Application Security*, *Architecture Risk Assessment*, *Threat Modeling*. Bookmark the *permalink*.

---

3 Approaches to Threat Modeling ⟶

BLOG

- Why Implement Enterprise Threat Modeling
- Meltdown and Spectre Addressed by ThreatModeler
- Improving DevOps: Including Legacy Systems
- Improving DevOps: Shifting Security Right
- DevOps Data-Driven Transformational Leadership

IriusRisk
Ex. 1002 - Page 142

© MyAppSecurity 2010-2020. All Rights Reserved.

IriusRisk
Ex. 1002 - Page 143

Microsoft Threat Modeling Tool

8/25/2017 • 1 min to read • Edit Online

The Threat Modeling Tool is a core element of the Microsoft Security Development Lifecycle (SDL). It allows software architects to identify and mitigate potential security issues early, when they are relatively easy and cost-effective to resolve. As a result, it greatly reduces the total cost of development. Also, we designed the tool with non-security experts in mind, making threat modeling easier for all developers by providing clear guidance on creating and analyzing threat models.

The tool enables anyone to:

- Communicate about the security design of their systems
- Analyze those designs for potential security issues using a proven methodology
- Suggest and manage mitigations for security issues

Here are some tooling capabilities and innovations, just to name a few:

- **Automation:** Guidance and feedback in drawing a model
- **STRIDE per Element:** Guided analysis of threats and mitigations
- **Reporting:** Security activities and testing in the verification phase
- **Unique Methodology:** Enables users to better visualize and understand threats
- **Designed for Developers and Centered on Software:** many approaches are centered on assets or attackers. We are centered on software. We build on activities that all software developers and architects are familiar with -- such as drawing pictures for their software architecture
- **Focused on Design Analysis:** The term "threat modeling" can refer to either a requirements or a design analysis technique. Sometimes, it refers to a complex blend of the two. The Microsoft SDL approach to threat modeling is a focused design analysis technique

Next steps

The table below contains important links to get you started with the Threat Modeling Tool:

STEP	DESCRIPTION
1	Download the Threat Modeling Tool
2	Read Our getting started guide
3	Get familiar with the features
4	Learn about generated threat categories
5	Find mitigations to generated threats

Resources

Here are a few older articles still relevant to threat modeling today:

- Article on the Importance of Threat Modeling
- Training Published by Trustworthy Computing

Check out what a few Threat Modeling Tool experts have done:

- Threats Manager
- Simone Curzi Security Blog

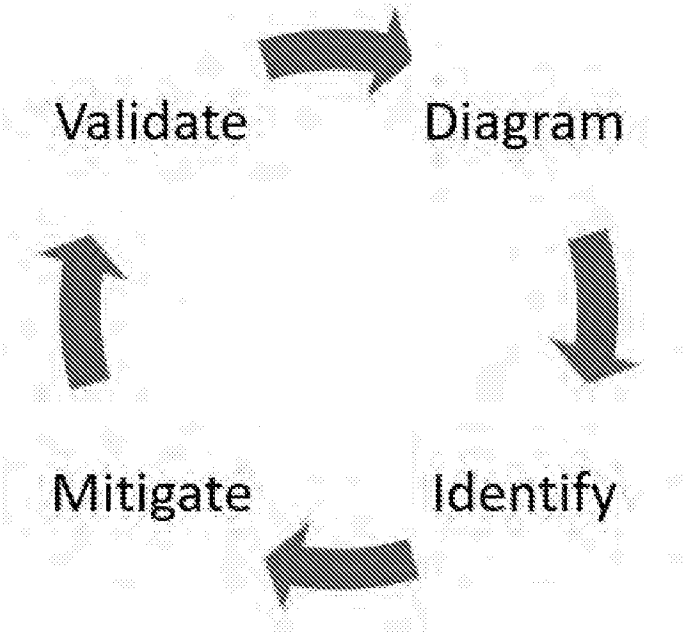
Getting started with the Threat Modeling Tool

12/18/2017 • 7 min to read • Edit Online

The Cloud and Enterprise Security Tools team released the Threat Modeling Tool Preview earlier this year as a free click-to-download. The change in delivery mechanism allows us to push the latest improvements and bug fixes to customers each time they open the tool, making it easier to maintain and use. This article takes you through the process of getting started with the Microsoft SDL threat modeling approach and shows you how to use the tool to develop great threat models as a backbone of your security process.

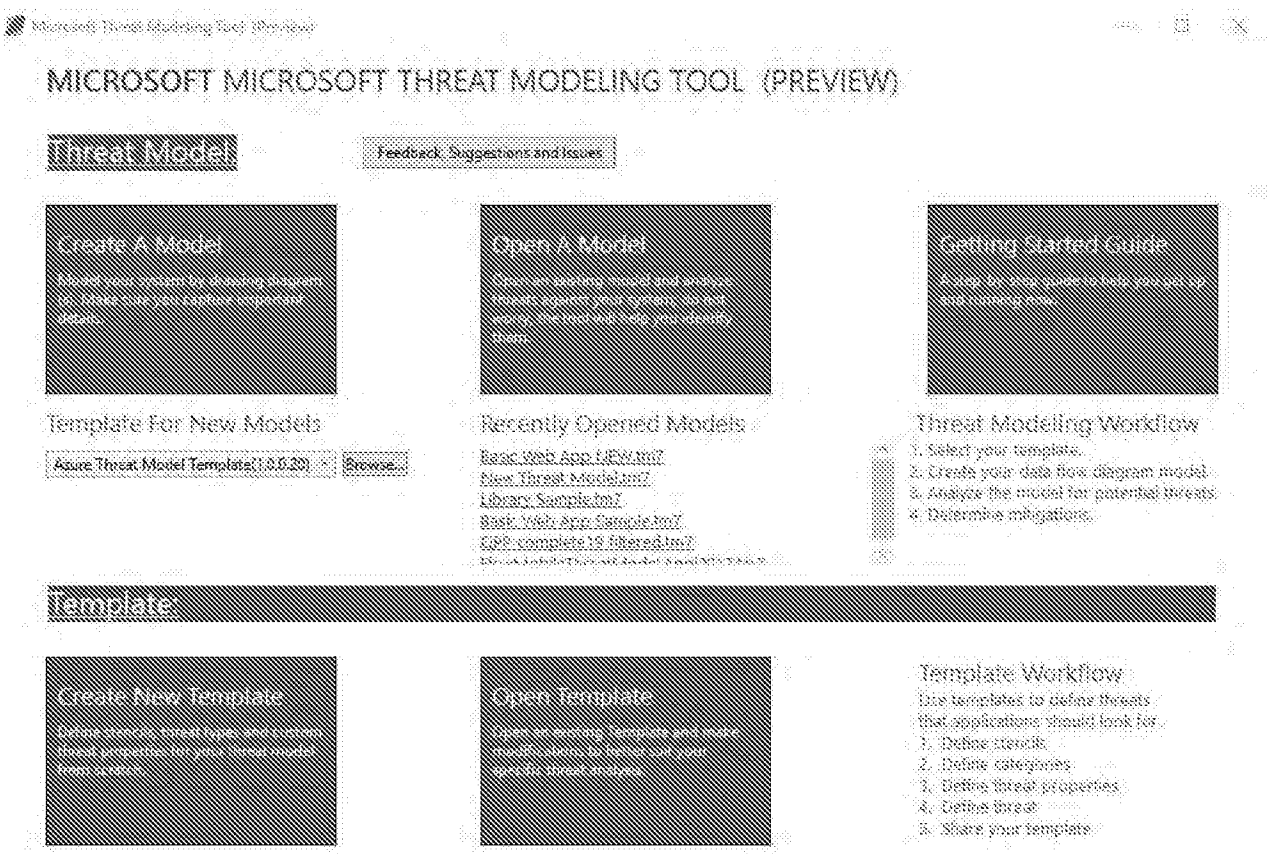
This article builds on existing knowledge of the SDL threat modeling approach. For a quick review, refer to **Threat Modeling Web Applications** and an archived version of **Uncover Security Flaws Using the STRIDE Approach** MSDN article published in 2006.

To quickly summarize, the approach involves creating a diagram, identifying threats, mitigating them and validating each mitigation. Here's a diagram that highlights this process:



Starting the threat modeling process

When you launch the Threat Modeling Tool, you'll notice a few things, as seen in the picture:



Threat model section

COMPONENT	DETAILS
Feedback, Suggestions and Issues Button	Takes you the MSDN Forum for all things SDL. It gives you an opportunity to read through what other users are doing, along with workarounds and recommendations. If you still can't find what you're looking for, email tmtextsupport@microsoft.com for our support team to help you
Create a Model	Opens a blank canvas for you to draw your diagram. Make sure to select which template you'd like to use for your model
Template for New Models	You must select which template to use before creating a model. Our main template is the Azure Threat Model Template, which contains Azure-specific stencils, threats and mitigations. For generic models, select the SDL TM Knowledge Base from the drop-down menu. Want to create your own template or submit a new one for all users? Check out our Template Repository GitHub Page to learn more
Open a Model	Opens previously saved threat models. The Recently Opened Models feature is great if you need to open your most recent files. When you hover over the selection, you'll find 2 ways to open models: • Open From this Computer -- classic way of opening a file using local storage • Open from OneDrive -- teams can use folders in OneDrive to save and share all their threat models in a single location to help increase productivity and collaboration

COMPONENT	DETAILS
Getting Started Guide	Opens the **Microsoft Threat Modeling Tool** main page

Template section

COMPONENT	DETAILS
Create New Template	Opens a blank template for you to build on. Unless you have extensive knowledge in building templates from scratch, we recommend you to build from existing ones
Open Template	Opens existing templates for you to make changes to

The Threat Modeling Tool team is constantly working to improve tool functionality and experience. A few minor changes might take place over the course of the year, but all major changes require rewrites in the guide. Refer to it often to ensure you get the latest announcements.

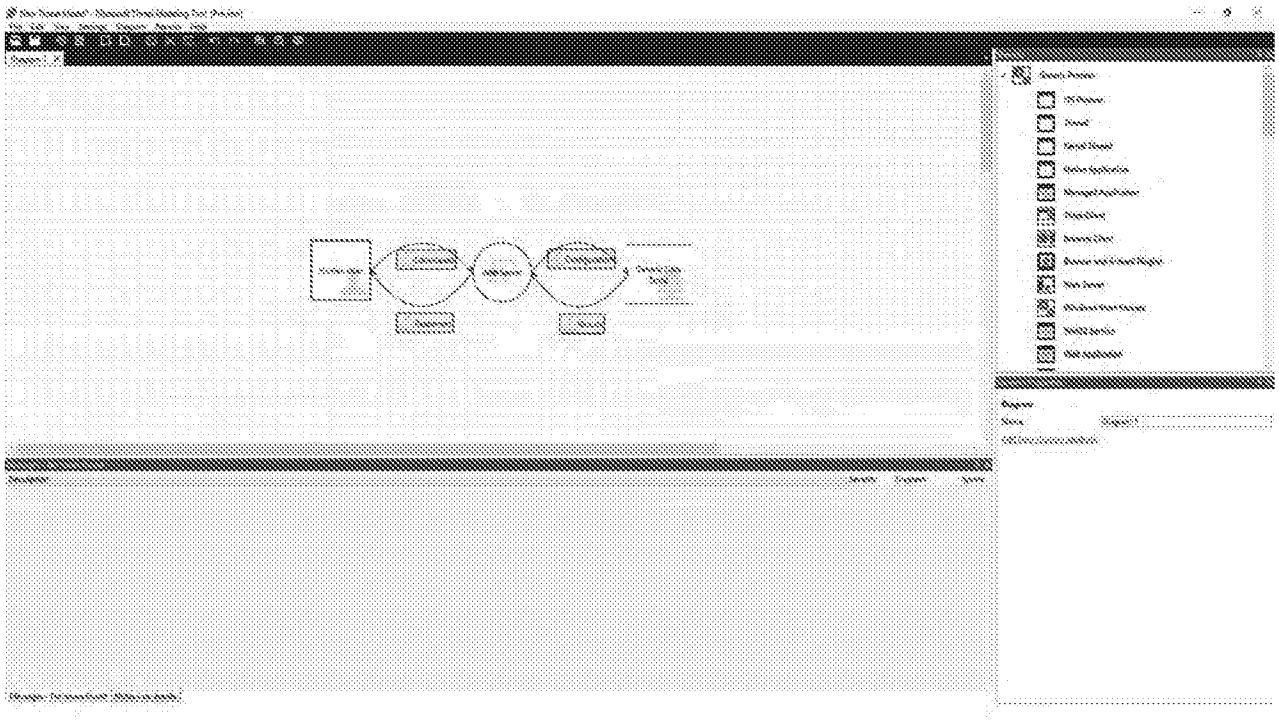
Building a model

In this section, we follow:

- Cristina (a developer)
- Ricardo (a program manager) and
- Ashish (a tester)

They are going through the process of developing their first threat model.

> Ricardo: Hi Cristina, I worked on the threat model diagram and wanted to make sure we got the details right. Can you help me look it over? Cristina: Absolutely. Let's take a look. Ricardo opens the tool and shares his screen with Cristina.



> Cristina: Ok, looks straightforward, but can you walk me through it? Ricardo: Sure! Here is the breakdown:
>
> - Our human user is drawn as an outside entity---a square

- They're sending commands to our Web server---the circle
- The Web server is consulting a database (two parallel lines)

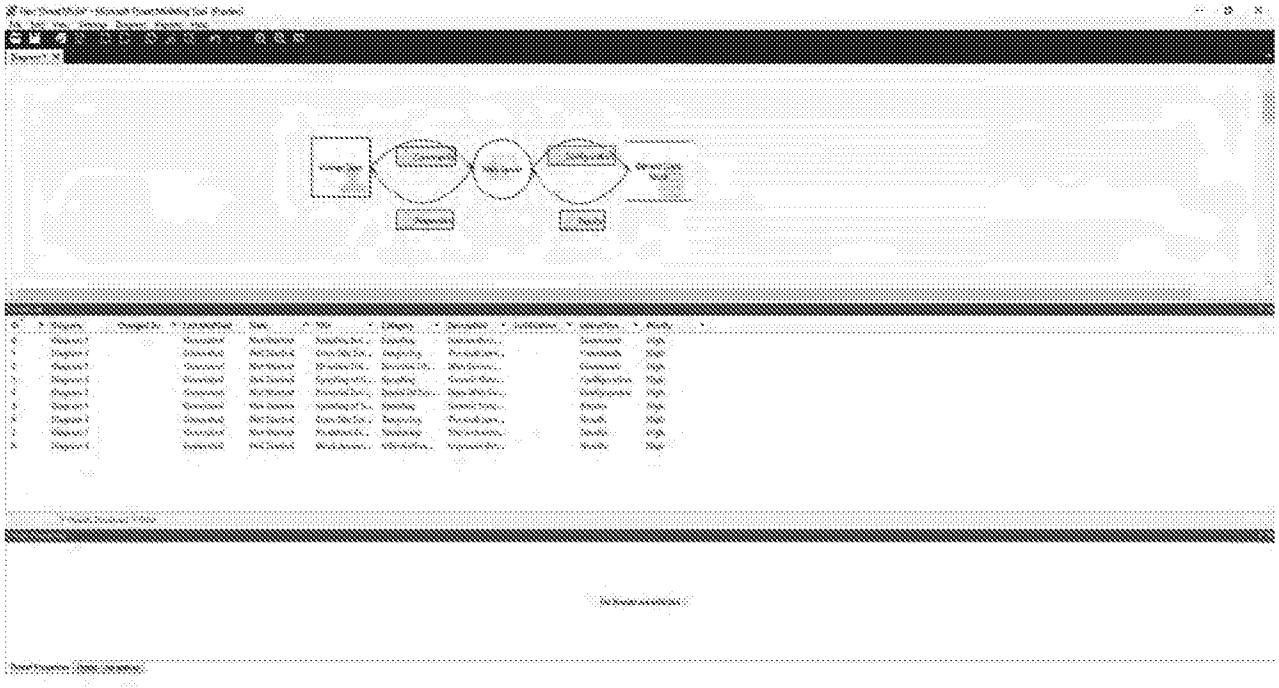
What Ricardo just showed Cristina is a DFD, short for Data Flow Diagram. The Threat Modeling Tool allows users to specify trust boundaries, indicated by the red dotted lines, to show where different entities are in control. For example, IT administrators require an Active Directory system for authentication purposes, so the Active Directory is outside of their control.

Cristina: Looks right to me. What about the threats? Ricardo: Let me show you.

Analyzing threats

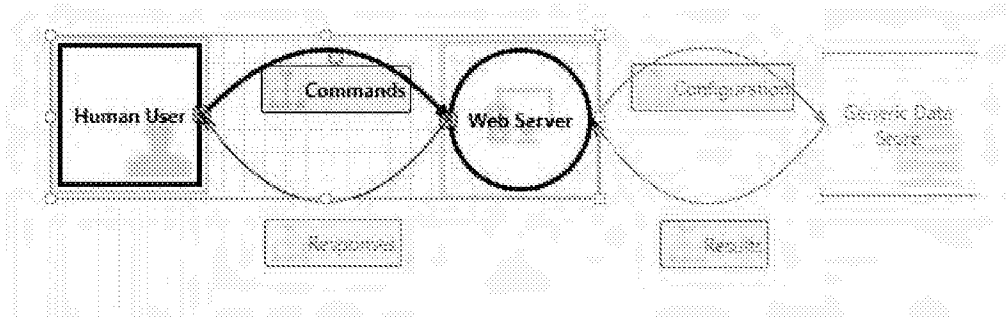
Once he clicks on the analysis view from the icon menu selection (file with magnifying glass), he is taken to a list of generated threats the Threat Modeling Tool found based on the default template, which uses the SDL approach called STRIDE (Spoofing, Tampering, Info Disclosure, Denial of Service and Elevation of Privilege). The idea is that software comes under a predictable set of threats, which can be found using these 6 categories.

This approach is like securing your house by ensuring each door and window has a locking mechanism in place before adding an alarm system or chasing after the thief.

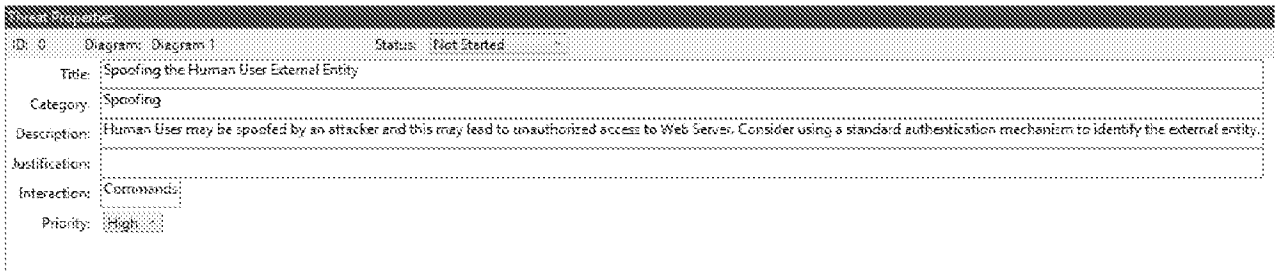


Ricardo begins by selecting the first item on the list. Here's what happens:

First, the interaction between the two stencils is enhanced



Second, additional information about the threat appears in the Threat Properties window



The generated threat helps him understand potential design flaws. The STRIDE categorization gives him an idea on potential attack vectors, while the additional description tells him exactly what's wrong, along with potential ways to mitigate it. He can use editable fields to write notes in the justification details or change priority ratings depending on his organization's bug bar.

Azure templates have additional details to help users understand not only what's wrong, but also how to fix it by adding descriptions, examples and hyperlinks to Azure-specific documentation.

The description made him realize the importance of adding an authentication mechanism to prevent users from being spoofed, revealing the first threat to be worked on. A few minutes into the discussion with Cristina, they understood the importance of implementing access control and roles. Ricardo filled in some quick notes to make sure these were implemented.

As Ricardo went into the threats under Information Disclosure, he realized the access control plan required some read-only accounts for audit and report generation. He wondered whether this should be a new threat, but the mitigations were the same, so he noted the threat accordingly. He also thought about information disclosure a bit more and realized that the backup tapes were going to need encryption, a job for the operations team.

Threats not applicable to the design due to existing mitigations or security guarantees can be changed to "Not Applicable" from the Status drop-down. There are three other choices: Not Started – default selection, Needs Investigation – used to follow up on items and Mitigated – once it's fully worked on.

Reports & sharing

Once Ricardo goes through the list with Cristina and adds important notes, mitigations/justifications, priority and status changes, he selects Reports -> Create Full Report -> Save Report, which prints out a nice report for him to go through with colleagues to ensure the proper security work is implemented.

Threat Modeling Report

Created on 7/31/2017 12:35:49 PM
Threat Model Name:
Owner:
Reviewer:
Contributors:
Description:
Assumptions:
External Dependencies:

Threat Model Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Diagram: Diagram 1

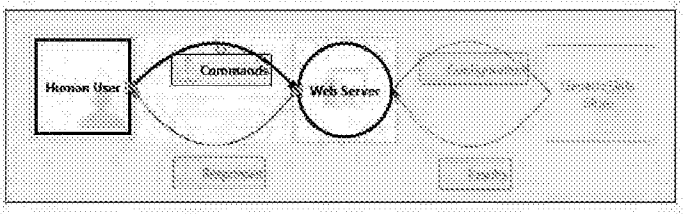
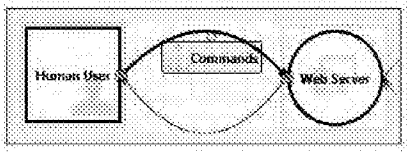


Diagram 1 Diagram Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Interaction: Commands



1. Spoofing the Human User External Entity [State: Not Started] [Priority: High]

Category:	Spoofing
Description:	Human User may be spoofed by an attacker and this may lead to unauthorized access to Web Server. Consider using a standard authentication mechanism to identify the external entity.
Justification:	<no mitigation provided>
Possible Mitigation(s):	
SDL Phase:	Design

2. Cross Site Scripting [State: Not Started] [Priority: High]

Category:	Tampering
Description:	The web server 'Web Server' could be a subject to a cross-site scripting attack because it does not sanitize untrusted input.
Justification:	<no mitigation provided>
Possible Mitigation(s):	
SDL Phase:	Design

If Ricardo wants to share the file instead, he can easily do so by saving in his organization's OneDrive account. Once he does that, he can copy the document link and share it with his colleagues.

Threat modeling meetings

When Ricardo sent his threat model to his colleague using OneDrive, Ashish, the tester, was underwhelmed. Seemed like Ricardo and Cristina missed quite a few important corner cases, which could be easily compromised. His skepticism is a complement to threat models.

In this scenario, after Ashish took over the threat model, he called for two threat modeling meetings: one meeting to synchronize on the process and walk through the diagrams and then a second meeting for threat review and sign-off.

In the first meeting, Ashish spent 10 minutes walking everyone through the SDL threat modeling process. He then pulled up the threat model diagram and started explaining it in detail. Within five minutes, an important missing component had been identified.

A few minutes later, Ashish and Ricardo got into an extended discussion of how the Web server was built. It was not the ideal way for a meeting to proceed, but everyone eventually agreed that discovering the discrepancy early was going to save them time in the future.

In the second meeting, the team walked through the threats, discussed some ways to address them, and signed off on the threat model. They checked the document into source control and continued with development.

Thinking about assets

Some readers who have threat modeled may notice that we haven't talked about assets at all. We've discovered that many software engineers understand their software better than they understand the concept of assets and what assets an attacker may be interested in.

If you're going to threat model a house, you might start by thinking about your family, irreplaceable photos or valuable artwork. Perhaps you might start by thinking about who might break in and the current security system. Or you might start by considering the physical features, like the pool or the front porch. These are analogous to thinking about assets, attackers, or software design. Any of these three approaches work.

The approach to threat modeling we've presented here is substantially simpler than what Microsoft has done in the past. We found that the software design approach works well for many teams. We hope that include yours.

Next Steps

Send your questions, comments and concerns to tmtextsupport@microsoft.com. Download the Threat Modeling Tool to get started.

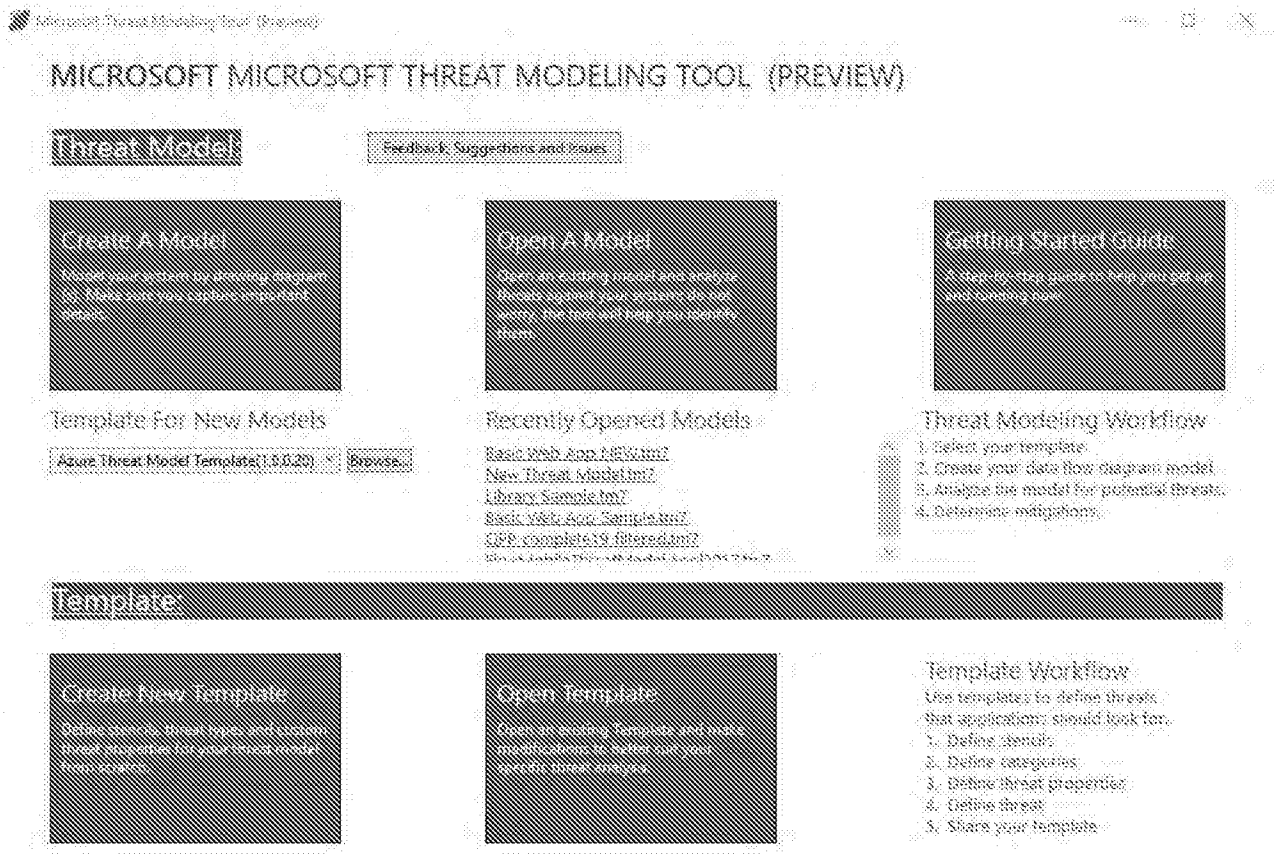
Threat Modeling Tool feature overview

12/18/2017 • 5 min to read • Edit Online

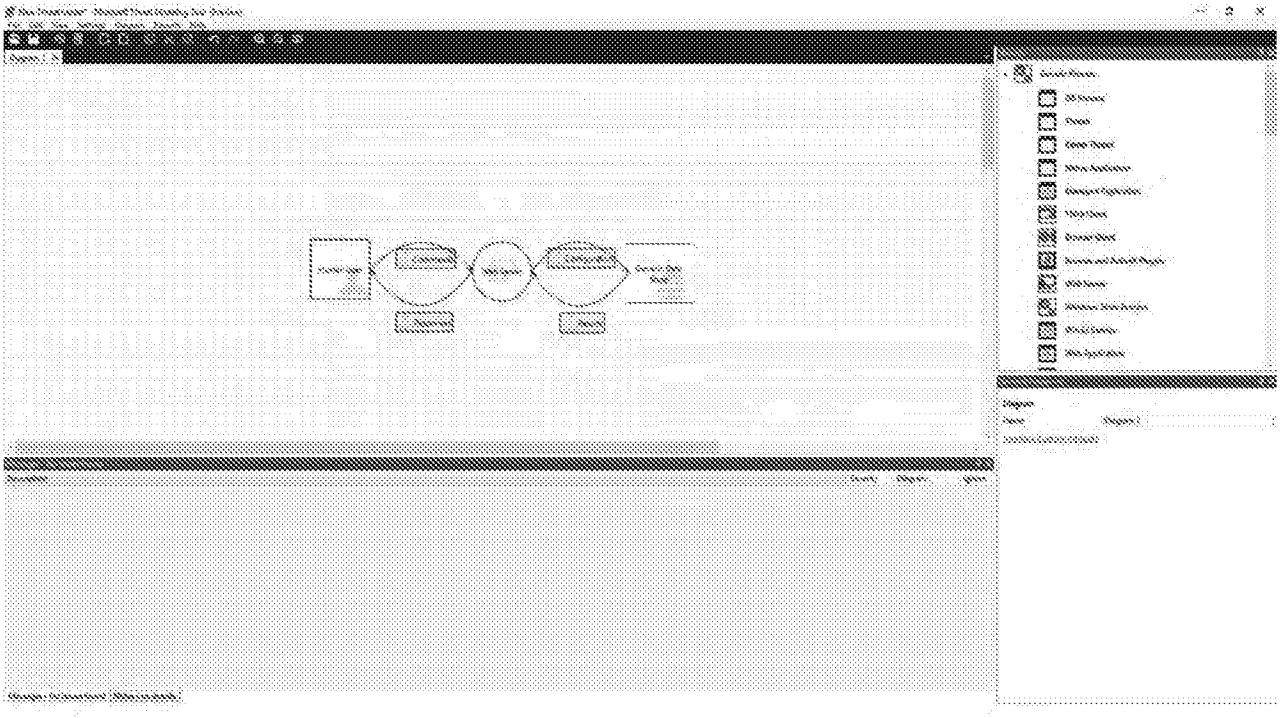
The Threat Modeling Tool can help you with your threat modeling needs. For a basic introduction to the tool, see Get started with the Threat Modeling Tool.

> **NOTE**
> The Threat Modeling Tool is updated frequently, so check this guide often to see our latest features and improvements.

To open a blank page, select **Create A Model**.



To see the features currently available in the tool, use the threat model created by our team in the Get started example.



Navigation

Before we discuss the built-in features, let's review the main components found in the tool.

Menu items

The experience is similar to other Microsoft products. Let's review the top-level menu items.



LABEL	DETAILS
File	• Open, save, and close files • Sign in and sign out of OneDrive accounts. • Share links (view and edit). • View file information. • Apply a new template to existing models.
Edit	Undo and redo actions, as well as copy, paste, and delete.
View	• Switch between **Analysis** and **Design** views. • Open closed windows (for example, stencils, element properties, and messages). • Reset layout to default settings.
Diagram	Add and delete diagrams, and move through tabs of diagrams.
Reports	Create HTML reports to share with others.
Help	Find guides to help you use the tool.

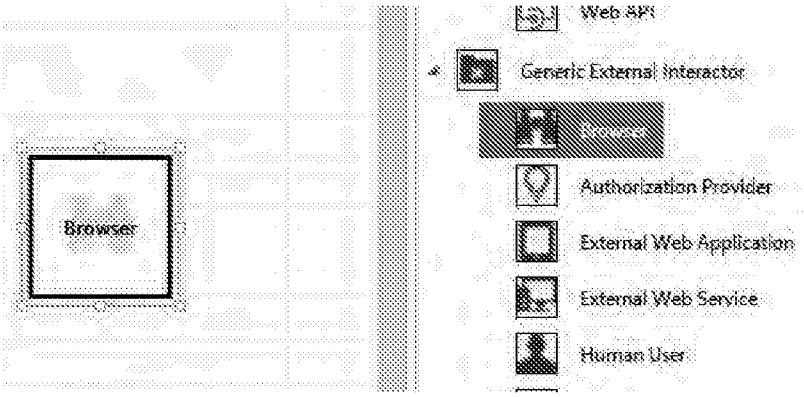
The symbols are shortcuts for the top-level menus:

SYMBOL	DETAILS
Open	Opens a new file.
Save	Saves the current file.
Design	Opens the **Design** view, where you can create models.
Analyze	Shows generated threats and their properties.
Add diagram	Adds a new diagram (similar to new tabs in Excel).
Delete diagram	Deletes the current diagram.
Copy/Cut/Paste	Copies, cuts, and pastes elements.
Undo/Redo	Undoes and redoes actions.
Zoom in/Zoom out	Zooms in and out of the diagram for a better view.
Feedback	Opens the MSDN Forum.

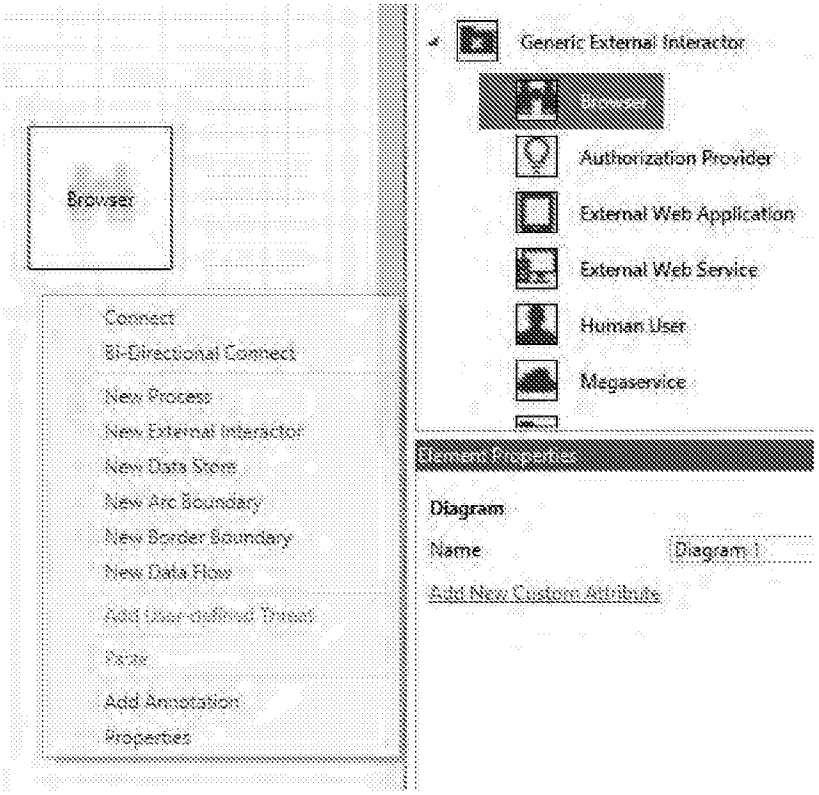
Canvas

The canvas is the space where you drag and drop elements. Drag and drop is the quickest and most efficient way to build models. You can also right-click and select items from the menu to add generic versions of elements, as shown:

Drop the stencil on the canvas



Select the stencil



Stencils

Based on the template you select, you can find all the stencils available to use. If you can't find the right elements, use another template. Or you can modify a template to fit your needs. Generally, you can find a combination of categories like these:

STENCIL NAME	DETAILS
Process	Applications, browser plug-ins, threads, virtual machines
External interactor	Authentication providers, browsers, users, web applications
Data store	Cache, storage, configuration files, databases, registry
Data flow	Binary, ALPC, HTTP, HTTPS/TLS/SSL, IOCTL, IPSec, named pipe, RPC/DCOM, SMB, UDP
Trust line/Border boundary	Corporate networks, internet, machine, sandbox, user/kernel mode

Notes/messages

COMPONENT	DETAILS
Messages	Internal tool logic that alerts users whenever there's an error, such as no data flows between elements.
Notes	Manual notes are added to the file by engineering teams throughout the design and review process.

Element properties

Element properties vary by the elements you select. Apart from trust boundaries, all other elements contain three general selections:

ELEMENT PROPERTY	DETAILS
Name	Useful for naming your processes, stores, interactors, and flows so that they're easily recognized.
Out of scope	If selected, the element is taken out of the threat-generation matrix (not recommended).
Reason for out of scope	Justification field to let users know why out of scope was selected.

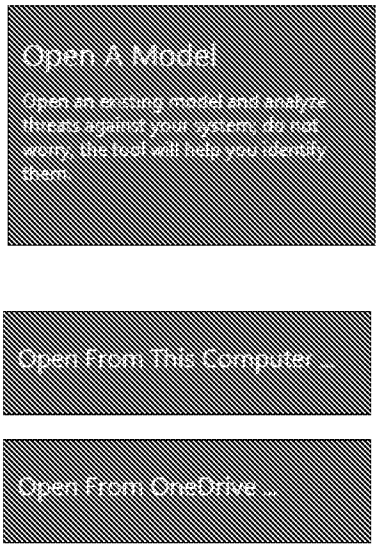
Properties are changed under each element category. Select each element to inspect the available options. Or you can open the template to learn more. Let's review the features.

Welcome screen

When you open the app, you see the **Welcome** screen.

Open a model

Hover over **Open A Model** to reveal two options: **Open From This Computer** and **Open From OneDrive**. The first option opens the **File Open** screen. The second option takes you through the sign-in process for OneDrive. After successful authentication, you can select folders and files.



Feedback, suggestions, and issues

When you select **Feedback, Suggestions and Issues**, you go to the MSDN Forum for SDL Tools. You can read what other people are saying about the tool, including workarounds and new ideas.



Design view

When you open or create a new model, the **Design** view opens.

Add elements

You can add elements on the grid in two ways:

- **Drag and drop**: Drag the desired element to the grid. Then use the element properties to provide additional information.
- **Right-click**: Right-click anywhere on the grid, and select items from the drop-down menu. A generic

representation of the element you select appears on the screen.

Connect elements

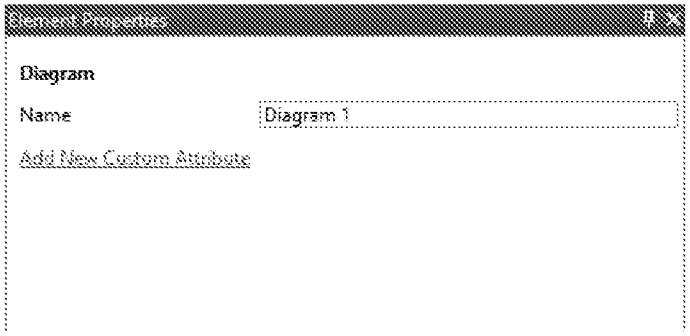
You can connect elements in two ways:

- **Drag and drop**: Drag the desired dataflow to the grid, and connect both ends to the appropriate elements.
- **Click + Shift**: Click the first element (sending data), press and hold the Shift key, and then select the second element (receiving data). Right-click, and select **Connect**. If you use a bi-directional data flow, the order is not as important.

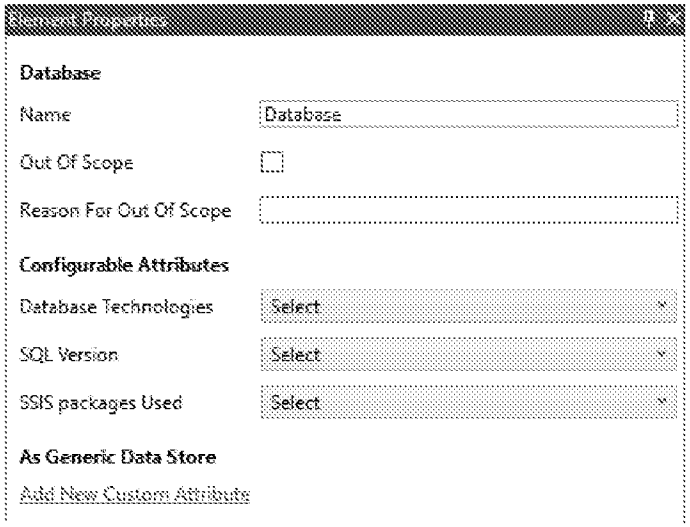
Properties

To see the properties that can be modified on the stencils, select the stencil and the information populates accordingly. The following example shows before and after a **Database** stencil is dragged onto the diagram:

Before

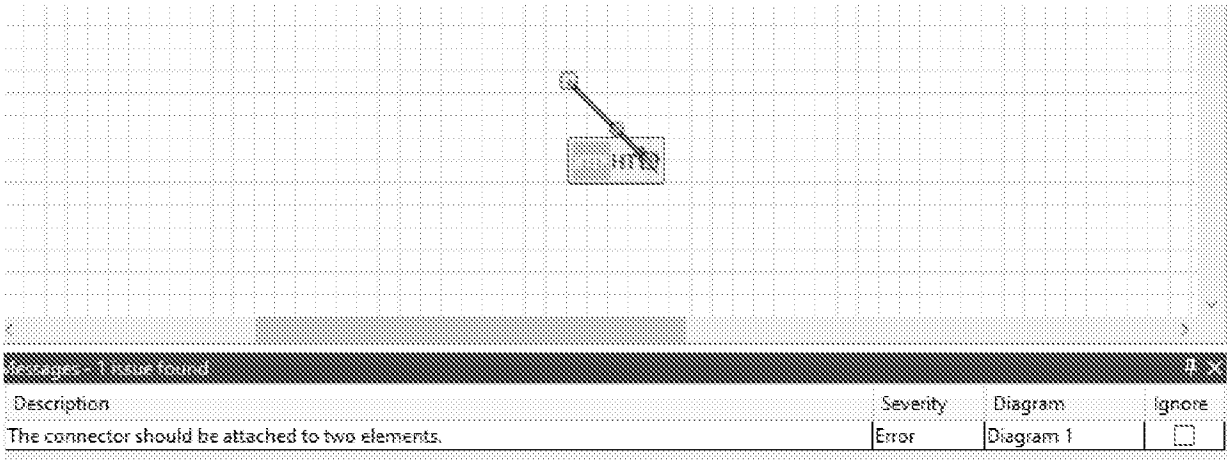


After



Messages

If you create a threat model and forget to connect data flows to elements, you get a notification. You can ignore the message, or you can follow the instructions to fix the issue.

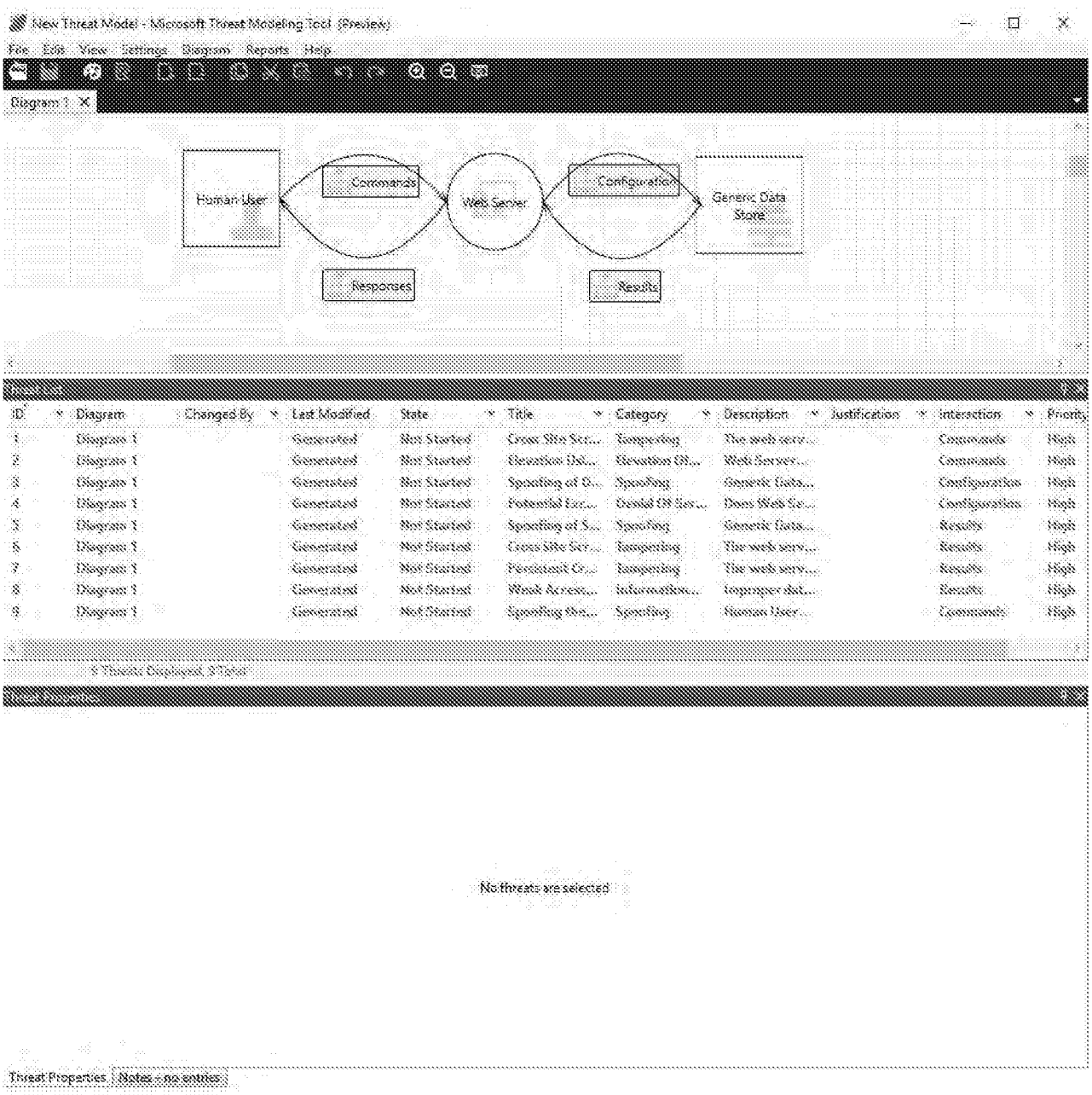


Notes

To add notes to your diagram, switch from the **Messages** tab to the **Notes** tab.

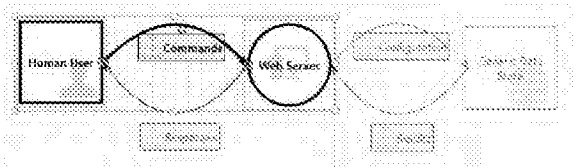
Analysis view

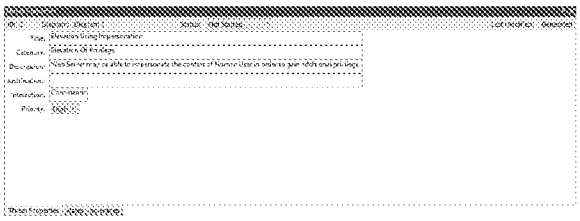
After you build your diagram, select the **Analysis** symbol (the magnifying glass) on the shortcuts toolbar to switch to the **Analysis** view.



Generated threat selection

When you select a threat, you can use three distinct functions:

FEATURE	INFORMATION
Read indicator	The threat is marked as read, which helps you keep track of the items you reviewed. 
Interaction focus	Interaction in the diagram that belongs to a threat is highlighted. 

FEATURE	INFORMATION
Threat properties	Additional information about the threat appears in the **Threat Properties** window. 

Priority change

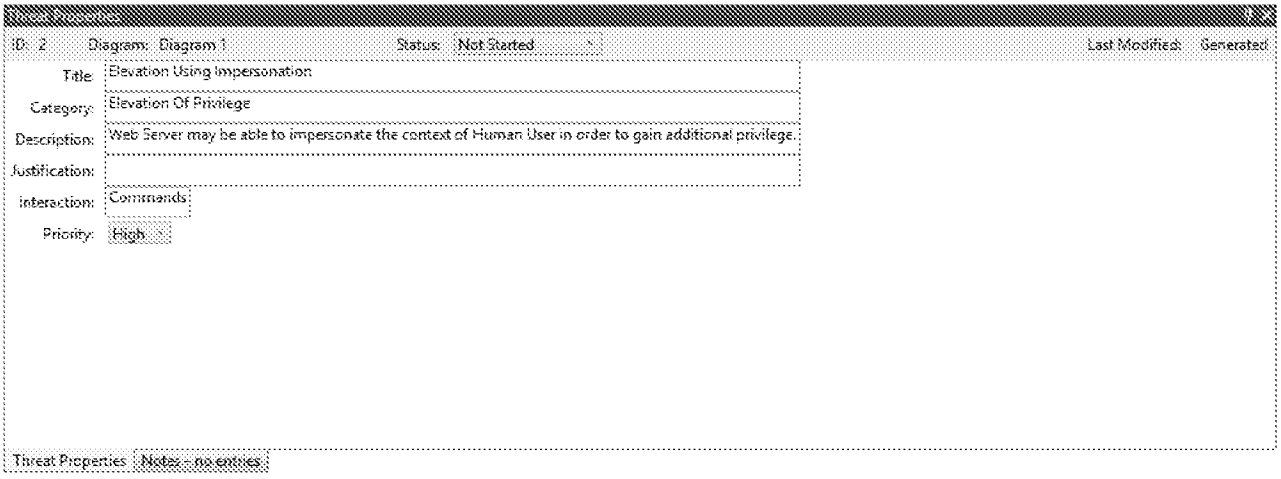
You can change the priority level of each generated threat. Different colors make it easy to identify high-, medium-, and low-priority threats.



ID	Diagram	Changed By	Last Modified	State	Title	Category	Description	Justification	Interaction	Priority
1	Diagram 1	[redacted]	[redacted]	Not Started	[redacted]	Tampering	[redacted]		[redacted]	[redacted]
2	Diagram 1	[redacted]	[redacted]	Not Started	Elevation Usin...	Elevation Of Pr...	Web Server ma...		Commands	Low
3	Diagram 1	[redacted]	[redacted]	Not Started	Spoofing of De...	Spoofing	Human User i...		Configuration	Medium
4	Diagram 1	[redacted]	[redacted]	Not Started	Potential Data ...	Denial Of Servi...	Data Flow Sniff...		Configuration	Medium
5	Diagram 1		Generated	Not Started	Spoofing of So...	Spoofing	Generic Data S...		Results	High
6	Diagram 1		Generated	Not Started	Cross Site Scri...	Tampering	The web server...		Results	High
7	Diagram 1	[redacted]	[redacted]	Not Started	Persistent Cro...	Tampering	The web server...		Results	Low
8	Diagram 1		Generated	Not Started	Weak Access ...	Information ...	Improper dat...		Results	High
9	Diagram 1		Generated	Not Started	Spoofing the ...	Spoofing	Human User i...		Commands	High

Threat properties editable fields

As seen in the preceding image, you can change the information generated by the tool. You can also add information to certain fields, such as justification. These fields are generated by the template. If you need more information for each threat, you can make modifications.



Reports

After you finish changing priorities and updating the status of each generated threat, you can save the file and/or print out a report. Go to **Report** > **Create Full Report**. Name the report, and you should see something similar to the following image:

Threat Modeling Report

Created on 7/31/2017 12:35:42 PM
Threat Model Name:
Owner:
Reviewer:
Contributors:
Description:
Assumptions:
External Dependencies:

Threat Model Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Diagram: Diagram 1

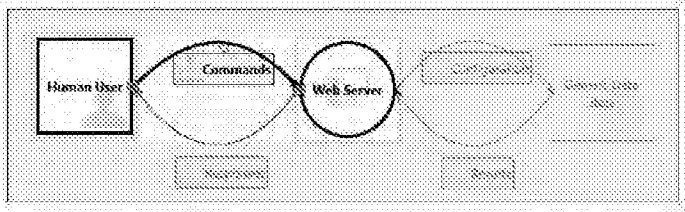
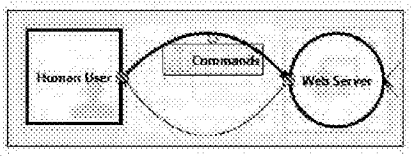


Diagram 1 Diagram Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Interaction: Commands



1: Spoofing the Human User External Entity [State: Not Started] [Priority: High]

Category:	Spoofing
Description:	Human User may be spoofed by an attacker and this may lead to unauthorized access to Web Server. Consider using a standard authentication mechanism to identify the external entity.
Justification:	<no mitigation provided>
Possible Mitigation(s):	
SDL Phase:	Design

2: Cross Site Scripting [State: Not Started] [Priority: High]

Category:	Tampering
Description:	The web server 'Web Server' could be a subject to a cross-site scripting attack because it does not sanitize untrusted input
Justification:	<no mitigation provided>
Possible Mitigation(s):	
SDL Phase:	Design

Next steps

- To contribute a template for the community, go to our GitHub page.
- To get started with the tool, go to the Download page.

Microsoft Threat Modeling Tool threats

8/25/2017 • 2 min to read • Edit Online

The Threat Modeling Tool is a core element of the Microsoft Security Development Lifecycle (SDL). It allows software architects to identify and mitigate potential security issues early, when they are relatively easy and cost-effective to resolve. As a result, it greatly reduces the total cost of development. Also, we designed the tool with non-security experts in mind, making threat modeling easier for all developers by providing clear guidance on creating and analyzing threat models.

> Visit the **Threat Modeling Tool** to get started today!

The Threat Modeling Tool helps you answer certain questions, such as the ones below:

- How can an attacker change the authentication data?
- What is the impact if an attacker can read the user profile data?
- What happens if access is denied to the user profile database?

STRIDE model

To better help you formulate these kinds of pointed questions, Microsoft uses the STRIDE model, which categorizes different types of threats and simplifies the overall security conversations.

CATEGORY	DESCRIPTION
Spoofing	Involves illegally accessing and then using another user's authentication information, such as username and password
Tampering	Involves the malicious modification of data. Examples include unauthorized changes made to persistent data, such as that held in a database, and the alteration of data as it flows between two computers over an open network, such as the Internet
Repudiation	Associated with users who deny performing an action without other parties having any way to prove otherwise—for example, a user performs an illegal operation in a system that lacks the ability to trace the prohibited operations. Non-Repudiation refers to the ability of a system to counter repudiation threats. For example, a user who purchases an item might have to sign for the item upon receipt. The vendor can then use the signed receipt as evidence that the user did receive the package
Information Disclosure	Involves the exposure of information to individuals who are not supposed to have access to it—for example, the ability of users to read a file that they were not granted access to, or the ability of an intruder to read data in transit between two computers
Denial of Service	Denial of service (DoS) attacks deny service to valid users—for example, by making a Web server temporarily unavailable or unusable. You must protect against certain types of DoS threats simply to improve system availability and reliability

CATEGORY	DESCRIPTION
Elevation of Privilege	An unprivileged user gains privileged access and thereby has sufficient access to compromise or destroy the entire system. Elevation of privilege threats include those situations in which an attacker has effectively penetrated all system defenses and become part of the trusted system itself, a dangerous situation indeed

Next steps

Proceed to **Threat Modeling Tool Mitigations** to learn the different ways you can mitigate these threats with Azure.

Microsoft Threat Modeling Tool mitigations

8/25/2017 • 2 min to read • Edit Online

The Threat Modeling Tool is a core element of the Microsoft Security Development Lifecycle (SDL). It allows software architects to identify and mitigate potential security issues early, when they are relatively easy and cost-effective to resolve. As a result, it greatly reduces the total cost of development. Also, we designed the tool with non-security experts in mind, making threat modeling easier for all developers by providing clear guidance on creating and analyzing threat models.

Visit the **Threat Modeling Tool** to get started today!

Mitigation categories

The Threat Modeling Tool mitigations are categorized according to the Web Application Security Frame, which consists of the following:

CATEGORY	DESCRIPTION
Auditing and Logging	Who did what and when? Auditing and logging refer to how your application records security-related events
Authentication	Who are you? Authentication is the process where an entity proves the identity of another entity, typically through credentials, such as a user name and password
Authorization	What can you do? Authorization is how your application provides access controls for resources and operations
Communication Security	Who are you talking to? Communication Security ensures all communication done is as secure as possible
Configuration Management	Who does your application run as? Which databases does it connect to? How is your application administered? How are these settings secured? Configuration management refers to how your application handles these operational issues
Cryptography	How are you keeping secrets (confidentiality)? How are you tamper-proofing your data or libraries (integrity)? How are you providing seeds for random values that must be cryptographically strong? Cryptography refers to how your application enforces confidentiality and integrity
Exception Management	When a method call in your application fails, what does your application do? How much do you reveal? Do you return friendly error information to end users? Do you pass valuable exception information back to the caller? Does your application fail gracefully?

CATEGORY	DESCRIPTION
Input Validation	How do you know that the input your application receives is valid and safe? Input validation refers to how your application filters, scrubs, or rejects input before additional processing. Consider constraining input through entry points and encoding output through exit points. Do you trust data from sources such as databases and file shares?
Sensitive Data	How does your application handle sensitive data? Sensitive data refers to how your application handles any data that must be protected either in memory, over the network, or in persistent stores
Session Management	How does your application handle and protect user sessions? A session refers to a series of related interactions between a user and your Web application

This helps you identify:

- Where are the most common mistakes made
- Where are the most actionable improvements

As a result, you use these categories to focus and prioritize your security work, so that if you know the most prevalent security issues occur in the input validation, authentication and authorization categories, you can start there. For more information visit this patent link

Next steps

Visit **Threat Modeling Tool Threats** to learn more about the threat categories the tool uses to generate possible design threats.

Security Frame: Auditing and Logging | Mitigations

8/22/2017 • 8 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Dynamics CRM	• Identify sensitive entities in your solution and implement change auditing
Web Application	• Ensure that auditing and logging is enforced on the application • Ensure that log rotation and separation are in place • Ensure that the application does not log sensitive user data • Ensure that Audit and Log Files have Restricted Access • Ensure that User Management Events are Logged • Ensure that the system has inbuilt defenses against misuse • Enable diagnostics logging for web apps in Azure App Service
Database	• Ensure that login auditing is enabled on SQL Server • Enable Threat detection on Azure SQL
Azure Storage	• Use Azure Storage Analytics to audit access of Azure Storage
WCF	• Implement sufficient Logging • Implement sufficient Audit Failure Handling
Web API	• Ensure that auditing and logging is enforced on Web API
IoT Field Gateway	• Ensure that appropriate auditing and logging is enforced on Field Gateway
IoT Cloud Gateway	• Ensure that appropriate auditing and logging is enforced on Cloud Gateway

Identify sensitive entities in your solution and implement change auditing

TITLE	DETAILS
Component	Dynamics CRM

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Identify entities in your solution containing sensitive data and implement change auditing on those entities and fields

Ensure that auditing and logging is enforced on the application

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Enable auditing and logging on all components. Audit logs should capture user context. Identify all important events and log those events. Implement centralized logging

Ensure that log rotation and separation are in place

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Log rotation is an automated process used in system administration in which dated log files are archived. Servers which run large applications often log every request: in the face of bulky logs, log rotation is a way to limit the total size of the logs while still allowing analysis of recent events. Log separation basically means that you have to store your log files on a different partition as where your OS/application is running on in order to avert a Denial of service attack or the downgrading of your application its performance

Ensure that the application does not log sensitive user data

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Check that you do not log any sensitive data that a user submits to your site. Check for intentional logging as well as side effects caused by design issues. Examples of sensitive data include: • User Credentials • Social Security number or other identifying information • Credit card numbers or other financial information • Health information • Private keys or other data that could be used to decrypt encrypted information • System or application information that can be used to more effectively attack the application

Ensure that Audit and Log Files have Restricted Access

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	Check to ensure access rights to log files are appropriately set. Application accounts should have write-only access and operators and support personnel should have read-only access as needed. Administrators accounts are the only accounts which should have full access. Check Windows ACL on log files to ensure they are properly restricted: • Application accounts should have write-only access • Operators and support personnel should have read-only access as needed • Administrators are the only accounts that should have full access

Ensure that User Management Events are Logged

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that the application monitors user management events such as successful and failed user logins, password resets, password changes, account lockout, user registration. Doing this helps to detect and react to potentially suspicious behavior. It also enables to gather operations data; for example, to track who is accessing the application

Ensure that the system has inbuilt defenses against misuse

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	Controls should be in place which throw security exception in case of application misuse. E.g., If input validation is in place and an attacker attempts to inject malicious code that does not match the regex, a security exception can be thrown which can be an indicative of system misuse For example, it is recommended to have security exceptions logged and actions taken for the following issues: • Input validation • CSRF violations • Brute force (upper limit for number of requests per user per resource) • File upload violations

Enable diagnostics logging for web apps in Azure App Service

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	EnvironmentType - Azure
References	N/A
Steps	Azure provides built-in diagnostics to assist with debugging an App Service web app. It also applies to API apps and mobile apps. App Service web apps provide diagnostic functionality for logging information from both the web server and the web application. These are logically separated into web server diagnostics and application diagnostics

Ensure that login auditing is enabled on SQL Server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	Configure Login Auditing
Steps	Database Server login auditing must be enabled to detect/confirm password guessing attacks. It is important to capture failed login attempts. Capturing both successful and failed login attempts provides additional benefit during forensic investigations

Enable Threat detection on Azure SQL

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure
Attributes	SQL Version - V12
References	Get Started with SQL Database Threat Detection
Steps	Threat Detection detects anomalous database activities indicating potential security threats to the database. It provides a new layer of security, which enables customers to detect and respond to potential threats as they occur by providing security alerts on anomalous activities. Users can explore the suspicious events using Azure SQL Database Auditing to determine if they result from an attempt to access, breach or exploit data in the database. Threat Detection makes it simple to address potential threats to the database without the need to be a security expert or manage advanced security monitoring systems

Use Azure Storage Analytics to audit access of Azure Storage

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Using Storage Analytics to monitor authorization type

TITLE	DETAILS
Steps	For each storage account, one can enable Azure Storage Analytics to perform logging and store metrics data. The storage analytics logs provide important information such as authentication method used by someone when they access storage. This can be really helpful if you are tightly guarding access to storage. For example, in Blob Storage you can set all of the containers to private and implement the use of an SAS service throughout your applications. Then you can check the logs regularly to see if your blobs are accessed using the storage account keys, which may indicate a breach of security, or if the blobs are public but they shouldn't be.

Implement sufficient Logging

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework
Attributes	N/A
References	MSDN, Fortify Kingdom
Steps	The lack of a proper audit trail after a security incident can hamper forensic efforts. Windows Communication Foundation (WCF) offers the ability to log successful and/or failed authentication attempts. Logging failed authentication attempts can warn administrators of potential brute-force attacks. Similarly, logging successful authentication events can provide a useful audit trail when a legitimate account is compromised. Enable WCF's service security audit feature

Example

The following is an example configuration with auditing enabled

```
<system.serviceModel>
    <behaviors>
        <serviceBehaviors>
            <behavior name=""NewBehavior"">
                <serviceSecurityAudit auditLogLocation=""Default""
                suppressAuditFailure=""false""
                serviceAuthorizationAuditLevel=""SuccessAndFailure""
                messageAuthenticationAuditLevel=""SuccessAndFailure"" />
                ...
            </behavior>
        </servicebehaviors>
    </behaviors>
</system.serviceModel>
```

Implement sufficient Audit Failure Handling

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework
Attributes	N/A
References	MSDN, Fortify Kingdom
Steps	Developed solution is configured not to generate an exception when it fails to write to an audit log. If WCF is configured not to throw an exception when it is unable to write to an audit log, the program will not be notified of the failure and auditing of critical security events may not occur.

Example

The `<behavior/>` element of the WCF configuration file below instructs WCF to not notify the application when WCF fails to write to an audit log.

```
<behaviors>
    <serviceBehaviors>
        <behavior name="NewBehavior">
            <serviceSecurityAudit auditLogLocation="Application"
            suppressAuditFailure="true"
            serviceAuthorizationAuditLevel="Success"
            messageAuthenticationAuditLevel="Success" />
        </behavior>
    </serviceBehaviors>
</behaviors>
```

Configure WCF to notify the program whenever it is unable to write to an audit log. The program should have an alternative notification scheme in place to alert the organization that audit trails are not being maintained.

Ensure that auditing and logging is enforced on Web API

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Enable auditing and logging on Web APIs. Audit logs should capture user context. Identify all important events and log those events. Implement centralized logging

Ensure that appropriate auditing and logging is enforced on Field Gateway

TITLE	DETAILS
Component	IoT Field Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	When multiple devices connect to a Field Gateway, ensure that connection attempts and authentication status (success or failure) for individual devices are logged and maintained on the Field Gateway. Also, in cases where Field Gateway is maintaining the IoT Hub credentials for individual devices, ensure that auditing is performed when these credentials are retrieved.Develop a process to periodically upload the logs to Azure IoT Hub/storage for long term retention.

Ensure that appropriate auditing and logging is enforced on Cloud Gateway

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Introduction to IoT Hub operations monitoring

TITLE	DETAILS
Steps	Design for collecting and storing audit data gathered through IoT Hub Operations Monitoring. Enable the following monitoring categories: • Device identity operations • Device-to-cloud communications • Cloud-to-device communications • Connections • File uploads

Security Frame: Authentication | Mitigations

1/24/2018 • 25 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Web Application	• Consider using a standard authentication mechanism to authenticate to Web Application • Applications must handle failed authentication scenarios securely • Enable step up or adaptive authentication • Ensure that administrative interfaces are appropriately locked down • Implement forgot password functionalities securely • Ensure that password and account policy are implemented • Implement controls to prevent username enumeration
Database	• When possible, use Windows Authentication for connecting to SQL Server • When possible use Azure Active Directory Authentication for Connecting to SQL Database • When SQL authentication mode is used, ensure that account and password policy are enforced on SQL server • Do not use SQL Authentication in contained databases
Azure Event Hub	• Use per device authentication credentials using SaS tokens
Azure Trust Boundary	• Enable Azure Multi-Factor Authentication for Azure Administrators
Service Fabric Trust Boundary	• Restrict anonymous access to Service Fabric Cluster • Ensure that Service Fabric client-to-node certificate is different from node-to-node certificate • Use AAD to authenticate clients to service fabric clusters • Ensure that service fabric certificates are obtained from an approved Certificate Authority (CA)
Identity Server	• Use standard authentication scenarios supported by Identity Server • Override the default Identity Server token cache with a scalable alternative
Machine Trust Boundary	• Ensure that deployed application's binaries are digitally signed

PRODUCT/SERVICE	ARTICLE
WCF	• Enable authentication when connecting to MSMQ queues in WCF • WCF-Do not set Message clientCredentialType to none • WCF-Do not set Transport clientCredentialType to none
Web API	• Ensure that standard authentication techniques are used to secure Web APIs
Azure AD	• Use standard authentication scenarios supported by Azure Active Directory • Override the default ADAL token cache with a scalable alternative • Ensure that TokenReplayCache is used to prevent the replay of ADAL authentication tokens • Use ADAL libraries to manage token requests from OAuth2 clients to AAD (or on-premises AD)
IoT Field Gateway	• Authenticate devices connecting to the Field Gateway
IoT Cloud Gateway	• Ensure that devices connecting to Cloud gateway are authenticated • Use per-device authentication credentials
Azure Storage	• Ensure that only the required containers and blobs are given anonymous read access • Grant limited access to objects in Azure storage using SAS or SAP

Consider using a standard authentication mechanism to authenticate to Web Application

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Details	Authentication is the process where an entity proves its identity, typically through credentials, such as a user name and password. There are multiple authentication protocols available which may be considered. Some of them are listed below: • Client certificates • Windows based • Forms based • Federation - ADFS • Federation - Azure AD • Federation - Identity Server Consider using a standard authentication mechanism to identify the source process

Applications must handle failed authentication scenarios securely

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Details	Applications that explicitly authenticate users must handle failed authentication scenarios securely.The authentication mechanism must: • Deny access to privileged resources when authentication fails • Display a generic error message after failed authentication and access denied occurs Test for: • Protection of privileged resources after failed logins • A generic error message is displayed on failed authentication and access denied event(s) • Accounts are disabled after an excessive number of failed attempts

Enable step up or adaptive authentication

TITLE	DETAILS
Component	Web Application
SDL Phase	Build

TITLE	DETAILS
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Details	Verify the application has additional authorization (such as step up or adaptive authentication, via multi-factor authentication such as sending OTP in SMS, email etc. or prompting for re-authentication) so the user is challenged before being granted access to sensitive information. This rule also applies for making critical changes to an account or action This also means that the adaptation of authentication has to be implemented in such a manner that the application correctly enforces context-sensitive authorization so as to not allow unauthorized manipulation by means of in example, parameter tampering

Ensure that administrative interfaces are appropriately locked down

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Details	The first solution is to grant access only from a certain source IP range to the administrative interface. If that solution would not be possible than it is always recommended to enforce a step-up or adaptive authentication for logging in into the administrative interface

Implement forgot password functionalities securely

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Details	The first thing is to verify that forgot password and other recovery paths send a link including a time-limited activation token rather than the password itself. Additional authentication based on soft-tokens (e.g. SMS token, native mobile applications, etc.) can be required as well before the link is sent over. Second, you should not lock out the users account whilst the process of getting a new password is in progress. This could lead to a Denial of service attack whenever an attacker decides to intentionally lock out the users with an automated attack. Third, whenever the new password request was set in progress, the message you display should be generalized in order to prevent username enumeration. Fourth, always disallow the use of old passwords and implement a strong password policy.

Ensure that password and account policy are implemented

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Details	Password and account policy in compliance with organizational policy and best practices should be implemented. To defend against brute-force and dictionary based guessing: Strong password policy must be implemented to ensure that users create complex password (e.g., 12 characters minimum length, alphanumeric and special characters). Account lockout policies may be implemented in the following manner: • **Soft lock-out:** This can be a good option for protecting your users against brute force attacks. For example, whenever the user enters a wrong password three times the application could lock down the account for a minute in order to slow down the process of brute forcing his password making it less profitable for the attacker to proceed. If u were to implement hard lock-out countermeasures for this example you would achieve a "Dos" by permanently locking out accounts. Alternatively, application may generate an OTP (One Time Password) and send it out-of-band (through email, sms etc.) to the user. Another approach may be to implement CAPTCHA after a threshold number of failed attempts is reached. • **Hard lock-out:** This type of lockout should be applied whenever you detect a user attacking your application and counter him by means of permanently locking out his account until a response team had time to do their forensics. After this process you can decide to give the user back his account or take further legal actions against him. This type of approach prevents the attacker from further penetrating your application and infrastructure. To defend against attacks on default and predictable accounts, verify that all keys and passwords are replaceable, and are generated or replaced after installation time. If the application has to auto-generate passwords, ensure that the generated passwords are random and have high entropy.

Implement controls to prevent username enumeration

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	All error messages should be generalized in order to prevent username enumeration. Also sometimes you cannot avoid information leaking in functionalities such as a registration page. Here you need to use rate-limiting methods like CAPTCHA to prevent an automated attack by an attacker.

When possible, use Windows Authentication for connecting to SQL Server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	OnPrem
Attributes	SQL Version - All
References	SQL Server - Choose an Authentication Mode
Steps	Windows Authentication uses Kerberos security protocol, provides password policy enforcement with regard to complexity validation for strong passwords, provides support for account lockout, and supports password expiration.

When possible use Azure Active Directory Authentication for Connecting to SQL Database

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure
Attributes	SQL Version - V12
References	Connecting to SQL Database By Using Azure Active Directory Authentication
Steps	**Minimum version:** Azure SQL Database V12 required to allow Azure SQL Database to use AAD Authentication against the Microsoft Directory

When SQL authentication mode is used, ensure that account and

password policy are enforced on SQL server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	SQL Server password policy
Steps	When using SQL Server Authentication, logins are created in SQL Server that are not based on Windows user accounts. Both the user name and the password are created by using SQL Server and stored in SQL Server. SQL Server can use Windows password policy mechanisms. It can apply the same complexity and expiration policies used in Windows to passwords used inside SQL Server.

Do not use SQL Authentication in contained databases

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	OnPrem, SQL Azure
Attributes	SQL Version - MSSQL2012, SQL Version - V12
References	Security Best Practices with Contained Databases
Steps	The absence of an enforced password policy may increase the likelihood of a weak credential being established in a contained database. Leverage Windows Authentication.

Use per device authentication credentials using SaS tokens

TITLE	DETAILS
Component	Azure Event Hub
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	Event Hubs authentication and security model overview
Steps	The Event Hubs security model is based on a combination of Shared Access Signature (SAS) tokens and event publishers. The publisher name represents the DeviceID that receives the token. This would help associate the tokens generated with the respective devices. All messages are tagged with originator on service side allowing detection of in-payload origin spoofing attempts. When authenticating devices, generate a per device SaS token scoped to a unique publisher.

Enable Azure Multi-Factor Authentication for Azure Administrators

TITLE	DETAILS
Component	Azure Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	What is Azure Multi-Factor Authentication?
Steps	Multi-factor authentication (MFA) is a method of authentication that requires more than one verification method and adds a critical second layer of security to user sign-ins and transactions. It works by requiring any two or more of the following verification methods: • Something you know (typically a password) • Something you have (a trusted device that is not easily duplicated, like a phone) • Something you are (biometrics)

Restrict anonymous access to Service Fabric Cluster

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure
References	Service Fabric cluster security scenarios

TITLE	DETAILS
Steps	Clusters should always be secured to prevent unauthorized users from connecting to your cluster, especially when it has production workloads running on it. While creating a service fabric cluster, ensure that the security mode is set to "secure" and configure the required X.509 server certificate. Creating an "insecure" cluster will allow any anonymous user to connect to it if it exposes management endpoints to the public Internet.

Ensure that Service Fabric client-to-node certificate is different from node-to-node certificate

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure, Environment - Stand alone
References	Service Fabric Client-to-node certificate security, Connect to a secure cluster using client certificate
Steps	Client-to-node certificate security is configured while creating the cluster either through the Azure portal, Resource Manager templates or a standalone JSON template by specifying an admin client certificate and/or a user client certificate. The admin client and user client certificates you specify should be different than the primary and secondary certificates you specify for Node-to-node security.

Use AAD to authenticate clients to service fabric clusters

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure
References	Cluster security scenarios - Security Recommendations

TITLE	DETAILS
Steps	Clusters running on Azure can also secure access to the management endpoints using Azure Active Directory (AAD), apart from client certificates. For Azure clusters, it is recommended that you use AAD security to authenticate clients and certificates for node-to-node security.

Ensure that service fabric certificates are obtained from an approved Certificate Authority (CA)

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure
References	X.509 certificates and Service Fabric
Steps	Service Fabric uses X.509 server certificates for authenticating nodes and clients. Some important things to consider while using certificates in service fabrics: • Certificates used in clusters running production workloads should be created by using a correctly configured Windows Server certificate service or obtained from an approved Certificate Authority (CA). The CA can be an approved external CA or a properly managed internal Public Key Infrastructure (PKI) • Never use any temporary or test certificates in production that are created with tools such as MakeCert.exe • You can use a self-signed certificate, but should only do so for test clusters and not in production

Use standard authentication scenarios supported by Identity Server

TITLE	DETAILS
Component	Identity Server
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	IdentityServer3 - The Big Picture

TITLE	DETAILS
Steps	Below are the typical interactions supported by Identity Server: • Browsers communicate with web applications • Web applications communicate with web APIs (sometimes on their own, sometimes on behalf of a user) • Browser-based applications communicate with web APIs • Native applications communicate with web APIs • Server-based applications communicate with web APIs • Web APIs communicate with web APIs (sometimes on their own, sometimes on behalf of a user)

Override the default Identity Server token cache with a scalable alternative

TITLE	DETAILS
Component	Identity Server
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Identity Server Deployment - Caching
Steps	IdentityServer has a simple built-in in-memory cache. While this is good for small scale native apps, it does not scale for mid tier and backend applications for the following reasons: • These applications are accessed by many users at once. Saving all access tokens in the same store creates isolation issues and presents challenges when operating at scale: many users, each with as many tokens as the resources the app accesses on their behalf, can mean huge numbers and very expensive lookup operations • These applications are typically deployed on distributed topologies, where multiple nodes must have access to the same cache • Cached tokens must survive process recycles and deactivations • For all the above reasons, while implementing web apps, it is recommended to override the default Identity Server's token cache with a scalable alternative such as Azure Redis cache

Ensure that deployed application's binaries are digitally signed

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that deployed application's binaries are digitally signed so that the integrity of the binaries can be verified

Enable authentication when connecting to MSMQ queues in WCF

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, NET Framework 3
Attributes	N/A
References	MSDN
Steps	Program fails to enable authentication when connecting to MSMQ queues, an attacker can anonymously submit messages to the queue for processing. If authentication is not used to connect to an MSMQ queue used to deliver a message to another program, an attacker could submit an anonymous message that is malicious.

Example

The `<netMsmqBinding/>` element of the WCF configuration file below instructs WCF to disable authentication when connecting to an MSMQ queue for message delivery.

```
<bindings>
    <netMsmqBinding>
        <binding>
            <security>
                <transport msmqAuthenticationMode=""None"" />
            </security>
        </binding>
    </netMsmqBinding>
</bindings>
```

Configure MSMQ to require Windows Domain or Certificate authentication at all times for any incoming or outgoing messages.

Example

The `<netMsmqBinding/>` element of the WCF configuration file below instructs WCF to enable certificate authentication when connecting to an MSMQ queue. The client is authenticated using X.509 certificates. The client certificate must be present in the certificate store of the server.

```
<bindings>
    <netMsmqBinding>
        <binding>
            <security>
                <transport msmqAuthenticationMode=""Certificate"" />
            </security>
        </binding>
    </netMsmqBinding>
</bindings>
```

WCF-Do not set Message clientCredentialType to none

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework 3
Attributes	Client Credential Type - None
References	MSDN, Fortify
Steps	The absence of authentication means everyone is able to access this service. A service that does not authenticate its clients allows access to all users. Configure the application to authenticate against client credentials. This can be done by setting the message clientCredentialType to Windows or Certificate.

Example

```
<message clientCredentialType=""Certificate""/>
```

WCF-Do not set Transport clientCredentialType to none

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, .NET Framework 3
Attributes	Client Credential Type - None
References	MSDN, Fortify

TITLE	DETAILS
Steps	The absence of authentication means everyone is able to access this service. A service that does not authenticate its clients allows all users to access its functionality. Configure the application to authenticate against client credentials. This can be done by setting the transport clientCredentialType to Windows or Certificate.

Example

```
<transport clientCredentialType=""Certificate""/>
```

Ensure that standard authentication techniques are used to secure Web APIs

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Authentication and Authorization in ASP.NET Web API, External Authentication Services with ASP.NET Web API (C#)
Steps	Authentication is the process where an entity proves its identity, typically through credentials, such as a user name and password. There are multiple authentication protocols available which may be considered. Some of them are listed below: • Client certificates • Windows based • Forms based • Federation - ADFS • Federation - Azure AD • Federation - Identity Server Links in the references section provide low-level details on how each of the authentication schemes can be implemented to secure a Web API.

Use standard authentication scenarios supported by Azure Active Directory

TITLE	DETAILS
Component	Azure AD

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Authentication Scenarios for Azure AD, Azure Active Directory Code Samples, Azure Active Directory developer's guide
Steps	Azure Active Directory (Azure AD) simplifies authentication for developers by providing identity as a service, with support for industry-standard protocols such as OAuth 2.0 and OpenID Connect. Below are the five primary application scenarios supported by Azure AD: • Web Browser to Web Application: A user needs to sign in to a web application that is secured by Azure AD • Single Page Application (SPA): A user needs to sign in to a single page application that is secured by Azure AD • Native Application to Web API: A native application that runs on a phone, tablet, or PC needs to authenticate a user to get resources from a web API that is secured by Azure AD • Web Application to Web API: A web application needs to get resources from a web API secured by Azure AD • Daemon or Server Application to Web API: A daemon application or a server application with no web user interface needs to get resources from a web API secured by Azure AD Please refer to the links in the references section for low-level implementation details

Override the default ADAL token cache with a scalable alternative

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Modern Authentication with Azure Active Directory for Web Applications, Using Redis as ADAL token cache

TITLE	DETAILS
Steps	The default cache that ADAL (Active Directory Authentication Library) uses is an in-memory cache that relies on a static store, available process-wide. While this works for native applications, it does not scale for mid tier and backend applications for the following reasons: • These applications are accessed by many users at once. Saving all access tokens in the same store creates isolation issues and presents challenges when operating at scale: many users, each with as many tokens as the resources the app accesses on their behalf, can mean huge numbers and very expensive lookup operations • These applications are typically deployed on distributed topologies, where multiple nodes must have access to the same cache • Cached tokens must survive process recycles and deactivations For all the above reasons, while implementing web apps, it is recommended to override the default ADAL token cache with a scalable alternative such as Azure Redis cache.

Ensure that TokenReplayCache is used to prevent the replay of ADAL authentication tokens

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Modern Authentication with Azure Active Directory for Web Applications

TITLE	DETAILS
Steps	The TokenReplayCache property allows developers to define a token replay cache, a store that can be used for saving tokens for the purpose of verifying that no token can be used more than once. This is a measure against a common attack, the aptly called token replay attack: an attacker intercepting the token sent at sign-in might try to send it to the app again ("replay" it) for establishing a new session. E.g., In OIDC code-grant flow, after successful user authentication, a request to "/signin-oidc" endpoint of the relying party is made with "id_token", "code" and "state" parameters. The relying party validates this request and establishes a new session. If an adversary captures this request and replays it, he/she can establish a successful session and spoof the user. The presence of the nonce in OpenID Connect can limit but not fully eliminate the circumstances in which the attack can be successfully enacted. To protect their applications, developers can provide an implementation of ITokenReplayCache and assign an instance to TokenReplayCache.

Example

```
// ITokenReplayCache defined in ADAL
public interface ITokenReplayCache
{
bool TryAdd(string securityToken, DateTime expiresOn);
bool TryFind(string securityToken);
}
```

Example

Here is a sample implementation of the ITokenReplayCache interface. (Please customize and implement your project-specific caching framework)

```
public class TokenReplayCache : ITokenReplayCache
{
    private readonly ICacheProvider cache; // Your project-specific cache provider
    public TokenReplayCache(ICacheProvider cache)
    {
        this.cache = cache;
    }
    public bool TryAdd(string securityToken, DateTime expiresOn)
    {
        if (this.cache.Get<string>(securityToken) == null)
        {
            this.cache.Set(securityToken, securityToken);
            return true;
        }
        return false;
    }
    public bool TryFind(string securityToken)
    {
        return this.cache.Get<string>(securityToken) != null;
    }
}
```

The implemented cache has to be referenced in OIDC options via the "TokenValidationParameters" property.

follows.

```
OpenIdConnectOptions openIdConnectOptions = new OpenIdConnectOptions
{
    AutomaticAuthenticate = true,
    ... // other configuration properties follow..
    TokenValidationParameters = new TokenValidationParameters
    {
        TokenReplayCache = new TokenReplayCache(/*Inject your cache provider*/);
    }
}
```

Please note that to test the effectiveness of this configuration, login into your local OIDC-protected application and capture the request to `"/signin-oidc"` endpoint in fiddler. When the protection is not in place, replaying this request in fiddler will set a new session cookie. When the request is replayed after the TokenReplayCache protection is added, the application will throw an exception as follows:

```
SecurityTokenReplayDetectedException: IDX10228: The securityToken has previously been validated, securityToken:
'eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6Ik1uQ19WWmNBVGZNNXBPWWlKSE1iYTlnb0VLWSIsImtpZCI6Ik1uQ1.....
```

Use ADAL libraries to manage token requests from OAuth2 clients to AAD (or on-premises AD)

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	ADAL
Steps	The Azure AD authentication Library (ADAL) enables client application developers to easily authenticate users to cloud or on-premises Active Directory (AD), and then obtain access tokens for securing API calls. ADAL has many features that make authentication easier for developers, such as asynchronous support, a configurable token cache that stores access tokens and refresh tokens, automatic token refresh when an access token expires and a refresh token is available, and more. By handling most of the complexity, ADAL can help a developer focus on business logic in their application and easily secure resources without being an expert on security. Separate libraries are available for .NET, JavaScript (client and Node.js), iOS, Android and Java.

Authenticate devices connecting to the Field Gateway

TITLE	DETAILS
Component	IoT Field Gateway

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that each device is authenticated by the Field Gateway before accepting data from them and before facilitating upstream communications with the Cloud Gateway. Also, ensure that devices connect with a per device credential so that individual devices can be uniquely identified.

Ensure that devices connecting to Cloud gateway are authenticated

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic, C#, NodeJS,
Attributes	N/A, Gateway choice - Azure IoT Hub
References	N/A, Azure IoT hub with .NET, Getting Started with IoT hub and Node JS, Securing IoT with SAS and certificates, Git repository
Steps	• **Generic:** Authenticate the device using Transport Layer Security (TLS) or IPSec. Infrastructure should support using pre-shared key (PSK) on those devices that cannot handle full asymmetric cryptography. Leverage Azure AD, Oauth. • **C#:** When creating a DeviceClient instance, by default, the Create method creates a DeviceClient instance that uses the AMQP protocol to communicate with IoT Hub. To use the HTTPS protocol, use the override of the Create method that enables you to specify the protocol. If you use the HTTPS protocol, you should also add the `Microsoft.AspNet.WebApi.Client` NuGet package to your project to include the `System.Net.Http.Formatting` namespace.

Example

```
static DeviceClient deviceClient;

static string deviceKey = "{device key}";
static string iotHubUri = "{iot hub hostname}";

var messageString = "{message in string format}";
var message = new Message(Encoding.ASCII.GetBytes(messageString));

deviceClient = DeviceClient.Create(iotHubUri, new DeviceAuthenticationWithRegistrySymmetricKey("myFirstDevice",
deviceKey));

await deviceClient.SendEventAsync(message);
```

Example

Node.JS: Authentication

Symmetric key

- Create a IoT hub on azure
- Create an entry in the device identity registry

```
javascript var device = new iothub.Device(null); device.deviceId = <DeviceId > registry.create(device,
function(err, deviceInfo, res) {})
```

- Create a simulated device

```
javascript var clientFromConnectionString = require('azure-iot-device-amqp').clientFromConnectionString; var
Message = require('azure-iot-device').Message; var connectionString = 'HostName=<HostName>DeviceId=
<DeviceId>SharedAccessKey=<SharedAccessKey>'; var client = clientFromConnectionString(connectionString);
```

SAS Token

- Gets internally generated when using symmetric key but we can generate and use it explicitly as well
- Define a protocol : `var Http = require('azure-iot-device-http').Http;`
- Create a sas token : ```javascript resourceUri = encodeURIComponent(resourceUri.toLowerCase()).toLowerCase(); var deviceName = ""; var expires = (Date.now() / 1000) + expiresInMins * 60; var toSign = resourceUri + '\n' + expires; // using crypto var decodedPassword = new Buffer(signingKey, 'base64').toString('binary'); const hmac = crypto.createHmac('sha256', decodedPassword); hmac.update(toSign); var base64signature = hmac.digest('base64'); var base64UriEncoded = encodeURIComponent(base64signature); // construct authorization string var token = "SharedAccessSignature sr=" + resourceUri + "%2fdevices%2f"+deviceName+"&sig="
  - base64UriEncoded + "&se=" + expires; if (policyName) token += "&skn="+policyName; return token; ```
- Connect using sas token: `javascript Client.fromSharedAccessSignature(sas, Http);` #### Certificates
- Generate a self signed X509 certificate using any tool such as OpenSSL to generate a .cert and .key files to store the certificate and the key respectively
- Provision a device that accepts secured connection using certificates.

```
javascript var connectionString = '<connectionString>'; var registry =
iothub.Registry.fromConnectionString(connectionString); var deviceJSON = {deviceId:"<deviceId>",
authentication: { x509Thumbprint: { primaryThumbprint: "<PrimaryThumbprint>", secondaryThumbprint: "
<SecondaryThumbprint>" } }} var device = deviceJSON; registry.create(device, function (err) {});
```

- Connect a device using a certificate

```
javascript var Protocol = require('azure-iot-device-http').Http; var Client = require('azure-iot-
device').Client; var connectionString = 'HostName=<HostName>DeviceId=<DeviceId>x509=true'; var client =
Client.fromConnectionString(connectionString, Protocol); var options = { key: fs.readFileSync('./key.pem',
'utf8'), cert: fs.readFileSync('./server.crt', 'utf8') }; // Calling setOptions with the x509 certificate and
key (and optionally, passphrase) will configure the client //transport to use x509 when connecting to IoT Hub
client.setOptions(options); //call fn to execute after the connection is set up client.open(fn);
```

Use per-device authentication credentials

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	Gateway choice - Azure IoT Hub
References	Azure IoT Hub Security Tokens
Steps	Use per device authentication credentials using SaS tokens based on Device key or Client Certificate, instead of IoT Hub-level shared access policies. This prevents the reuse of authentication tokens of one device or field gateway by another

Ensure that only the required containers and blobs are given anonymous read access

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	StorageType - Blob
References	Manage anonymous read access to containers and blobs, Shared Access Signatures, Part 1: Understanding the SAS model

TITLE	DETAILS
Steps	By default, a container and any blobs within it may be accessed only by the owner of the storage account. To give anonymous users read permissions to a container and its blobs, one can set the container permissions to allow public access. Anonymous users can read blobs within a publicly accessible container without authenticating the request. Containers provide the following options for managing container access: • Full public read access: Container and blob data can be read via anonymous request. Clients can enumerate blobs within the container via anonymous request, but cannot enumerate containers within the storage account. • Public read access for blobs only: Blob data within this container can be read via anonymous request, but container data is not available. Clients cannot enumerate blobs within the container via anonymous request • No public read access: Container and blob data can be read by the account owner only Anonymous access is best for scenarios where certain blobs should always be available for anonymous read access. For finer-grained control, one can create a shared access signature, which enables to delegate restricted access using different permissions and over a specified time interval. Ensure that containers and blobs, which may potentially contain sensitive data, are not given anonymous access accidentally

Grant limited access to objects in Azure storage using SAS or SAP

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Shared Access Signatures, Part 1: Understanding the SAS model, Shared Access Signatures, Part 2: Create and use a SAS with Blob storage, How to delegate access to objects in your account using Shared Access Signatures and Stored Access Policies

TITLE	DETAILS
Steps	Using a shared access signature (SAS) is a powerful way to grant limited access to objects in a storage account to other clients, without having to expose account access key. The SAS is a URI that encompasses in its query parameters all of the information necessary for authenticated access to a storage resource. To access storage resources with the SAS, the client only needs to pass in the SAS to the appropriate constructor or method. You can use a SAS when you want to provide access to resources in your storage account to a client that can't be trusted with the account key. Your storage account keys include both a primary and secondary key, both of which grant administrative access to your account and all of the resources in it. Exposing either of your account keys opens your account to the possibility of malicious or negligent use. Shared access signatures provide a safe alternative that allows other clients to read, write, and delete data in your storage account according to the permissions you've granted, and without need for the account key. If you have a logical set of parameters that are similar each time, using a Stored Access Policy (SAP) is a better idea. Because using a SAS derived from a Stored Access Policy gives you the ability to revoke that SAS immediately, it is the recommended best practice to always use Stored Access Policies when possible.

Security Frame: Authorization | Mitigations

1/24/2018 • 17 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Machine Trust Boundary	• Ensure that proper ACLs are configured to restrict unauthorized access to data on the device • Ensure that sensitive user-specific application content is stored in user-profile directory • Ensure that the deployed applications are run with least privileges
Web Application	• Enforce sequential step order when processing business logic flows • Implement rate limiting mechanism to prevent enumeration • Ensure that proper authorization is in place and principle of least privileges is followed • Business logic and resource access authorization decisions should not be based on incoming request parameters • Ensure that content and resources are not enumerable or accessible via forceful browsing
Database	• Ensure that least-privileged accounts are used to connect to Database server • Implement Row Level Security RLS to prevent tenants from accessing each other's data • Sysadmin role should only have valid necessary users
IoT Cloud Gateway	• Connect to Cloud Gateway using least-privileged tokens
Azure Event Hub	• Use a send-only permissions SAS Key for generating device tokens • Do not use access tokens that provide direct access to the Event Hub • Connect to Event Hub using SAS keys that have the minimum permissions required
Azure Document DB	• Use resource tokens to connect to Azure Cosmos DB whenever possible
Azure Trust Boundary	• Enable fine-grained access management to Azure Subscription using RBAC
Service Fabric Trust Boundary	• Restrict client's access to cluster operations using RBAC

PRODUCT/SERVICE	ARTICLE
Dynamics CRM	• Perform security modeling and use Field Level Security where required
Dynamics CRM Portal	• Perform security modeling of portal accounts keeping in mind that the security model for the portal differs from the rest of CRM
Azure Storage	• Grant fine-grained permission on a range of entities in Azure Table Storage • Enable Role-Based Access Control (RBAC) to Azure storage account using Azure Resource Manager
Mobile Client	• Implement implicit jailbreak or rooting detection
WCF	• Weak Class Reference in WCF • WCF-Implement Authorization control
Web API	• Implement proper authorization mechanism in ASP.NET Web API
IoT Device	• Perform authorization checks in the device if it supports various actions that require different permission levels
IoT Field Gateway	• Perform authorization checks in the Field Gateway if it supports various actions that require different permission levels

Ensure that proper ACLs are configured to restrict unauthorized access to data on the device

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that proper ACLs are configured to restrict unauthorized access to data on the device

Ensure that sensitive user-specific application content is stored in user-profile directory

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that sensitive user-specific application content is stored in user-profile directory. This is to prevent multiple users of the machine from accessing each other's data.

Ensure that the deployed applications are run with least privileges

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that the deployed application is run with least privileges.

Enforce sequential step order when processing business logic flows

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	In order to verify that this stage was run through by a genuine user you want to enforce the application to only process business logic flows in sequential step order, with all steps being processed in realistic human time, and not process out of order, skipped steps, processed steps from another user, or too quickly submitted transactions.

Implement rate limiting mechanism to prevent enumeration

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that sensitive identifiers are random. Implement CAPTCHA control on anonymous pages. Ensure that error and exception should not reveal specific data

Ensure that proper authorization is in place and principle of least privileges is followed

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	The principle means giving a user account only those privileges which are essential to that users work. For example, a backup user does not need to install software: hence, the backup user has rights only to run backup and backup-related applications. Any other privileges, such as installing new software, are blocked. The principle applies also to a personal computer user who usually does work in a normal user account, and opens a privileged, password protected account (that is, a superuser) only when the situation absolutely demands it. This principle can also be applied to your web-applications. Instead of solely depending on role-based authentication methods using sessions, we rather want to assign privileges to users by means of a Database-Based Authentication system. We still use sessions in order to identify if the user was logged in correctly, only now instead of assigning that user with a specific role we assign him with privileges to verify which actions he is privileged to perform on the system. Also a big pro of this method is, whenever a user has to be assigned fewer privileges your changes will be applied on the fly since the assigning does not depend on the session which otherwise had to expire first.

Business logic and resource access authorization decisions should not be based on incoming request parameters

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Whenever you are checking whether a user is restricted to review certain data, the access restrictions should be processed server-side. The userID should be stored inside of a session variable on login and should be used to retrieve user data from the database

Example

```
SELECT data
FROM personaldata
WHERE userID=:id < - session var
```

Now an possible attacker can not tamper and change the application operation since the identifier for retrieving the data is handled server-side.

Ensure that content and resources are not enumerable or accessible via forceful browsing

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Sensitive static and configuration files should not be kept in the web-root. For content not required to be public, either proper access controls should be applied or removal of the content itself. Also, forceful browsing is usually combined with Brute Force techniques to gather information by attempting to access as many URLs as possible to enumerate directories and files on a server. Attackers may check for all variations of commonly existing files. For example, a password file search would encompass files including psswd.txt, password.htm, password.dat, and other variations. To mitigate this, capabilities for detection of brute force attempts should be included.

Ensure that least-privileged accounts are used to connect to Database server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	SQL Database permissions hierarchy, SQL database securables
Steps	Least-privileged accounts should be used to connect to the database. Application login should be restricted in the database and should only execute selected stored procedures. Application's login should have no direct table access.

Implement Row Level Security RLS to prevent tenants from accessing each other's data

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Sql Azure, OnPrem
Attributes	SQL Version - V12, SQL Version - MsSQL2016
References	SQL Server Row-Level Security (RLS)
Steps	Row-Level Security enables customers to control access to rows in a database table based on the characteristics of the user executing a query (e.g., group membership or execution context). Row-Level Security (RLS) simplifies the design and coding of security in your application. RLS enables you to implement restrictions on data row access. For example ensuring that workers can access only those data rows that are pertinent to their department, or restricting a customer's data access to only the data relevant to their company. The access restriction logic is located in the database tier rather than away from the data in another application tier. The database system applies the access restrictions every time that data access is attempted from any tier. This makes the security system more reliable and robust by reducing the surface area of the security system.

Please note that RLS as an out-of-the-box database feature is applicable only to SQL Server starting 2016 and Azure SQL database. If the out-of-the-box RLS feature is not implemented, it should be ensured that data access is restricted Using Views and Procedures

Sysadmin role should only have valid necessary users

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	SQL Database permissions hierarchy, SQL database securables
Steps	Members of the SysAdmin fixed server role should be very limited and never contain accounts used by applications. Please review the list of users in the role and remove any unnecessary accounts

Connect to Cloud Gateway using least-privileged tokens

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Gateway choice - Azure IoT Hub
References	Iot Hub Access Control
Steps	Provide least privilege permissions to various components that connect to Cloud Gateway (IoT Hub). Typical example is -- Device management/provisioning component uses registryread/write, Event Processor (ASA) uses Service Connect. Individual devices connect using Device credentials

Use a send-only permissions SAS Key for generating device tokens

TITLE	DETAILS
Component	Azure Event Hub
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Event Hubs authentication and security model overview
Steps	A SAS key is used to generate individual device tokens. Use a send-only permissions SAS key while generating the device token for a given publisher

Do not use access tokens that provide direct access to the Event Hub

TITLE	DETAILS
Component	Azure Event Hub
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Event Hubs authentication and security model overview

TITLE	DETAILS
Steps	A token that grants direct access to the event hub should not be given to the device. Using a least privileged token for the device that gives access only to a publisher would help identify and blacklist it if found to be a rogue or compromised device.

Connect to Event Hub using SAS keys that have the minimum permissions required

TITLE	DETAILS
Component	Azure Event Hub
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Event Hubs authentication and security model overview
Steps	Provide least privilege permissions to various back-end applications that connect to the Event Hub. Generate separate SAS keys for each back-end application and only provide the required permissions - Send, Receive or Manage to them.

Use resource tokens to connect to Cosmos DB whenever possible

TITLE	DETAILS
Component	Azure Document DB
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	A resource token is associated with an Azure Cosmos DB permission resource and captures the relationship between the user of a database and the permission that user has for a specific Azure Cosmos DB application resource (e.g. collection, document). Always use a resource token to access the Azure Cosmos DB if the client cannot be trusted with handling master or read-only keys - like an end user application like a mobile or desktop client.Use Master key or read-only keys from backend applications which can store these keys securely.

Enable fine-grained access management to Azure Subscription using

RBAC

TITLE	DETAILS
Component	Azure Trust Boundary
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Use role assignments to manage access to your Azure subscription resources
Steps	Azure Role-Based Access Control (RBAC) enables fine-grained access management for Azure. Using RBAC, you can grant only the amount of access that users need to perform their jobs.

Restrict client's access to cluster operations using RBAC

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure
References	Role-based access control for Service Fabric clients
Steps	Azure Service Fabric supports two different access control types for clients that are connected to a Service Fabric cluster: administrator and user. Access control allows the cluster administrator to limit access to certain cluster operations for different groups of users, making the cluster more secure. Administrators have full access to management capabilities (including read/write capabilities). Users, by default, have only read access to management capabilities (for example, query capabilities), and the ability to resolve applications and services. You specify the two client roles (administrator and client) at the time of cluster creation by providing separate certificates for each.

Perform security modeling and use Field Level Security where required

TITLE	DETAILS
Component	Dynamics CRM
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Perform security modeling and use Field Level Security where required

Perform security modeling of portal accounts keeping in mind that the security model for the portal differs from the rest of CRM

TITLE	DETAILS
Component	Dynamics CRM Portal
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Perform security modeling of portal accounts keeping in mind that the security model for the portal differs from the rest of CRM

Grant fine-grained permission on a range of entities in Azure Table Storage

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	StorageType - Table
References	How to delegate access to objects in your Azure storage account using SAS

TITLE	DETAILS
Steps	In certain business scenarios, Azure Table Storage may be required to store sensitive data that caters to different parties. E.g., sensitive data pertaining to different countries. In such cases, SAS signatures can be constructed by specifying the partition and row key ranges, such that a user can access data specific to a particular country.

Enable Role-Based Access Control (RBAC) to Azure storage account using Azure Resource Manager

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	How to secure your storage account with Role-Based Access Control (RBAC)
Steps	When you create a new storage account, you select a deployment model of Classic or Azure Resource Manager. The Classic model of creating resources in Azure only allows all-or-nothing access to the subscription, and in turn, the storage account. With the Azure Resource Manager model, you put the storage account in a resource group and control access to the management plane of that specific storage account using Azure Active Directory. For example, you can give specific users the ability to access the storage account keys, while other users can view information about the storage account, but cannot access the storage account keys.

Implement implicit jailbreak or rooting detection

TITLE	DETAILS
Component	Mobile Client
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Application should safeguard its own configuration and user data in case if phone is rooted or jail broken. Rooting/jail breaking implies unauthorized access, which normal users won't do on their own phones. Therefore application should have implicit detection logic on application startup, to detect if the phone has been rooted. The detection logic can be simply accessing files which normally only root user can access, for example: • /system/app/Superuser.apk • /sbin/su • /system/bin/su • /system/xbin/su • /data/local/xbin/su • /data/local/bin/su • /system/sd/xbin/su • /system/bin/failsafe/su • /data/local/su If the application can access any of these files, it denotes that the application is running as root user.

Weak Class Reference in WCF

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, NET Framework 3
Attributes	N/A
References	MSDN, Fortify Kingdom

TITLE	DETAILS
Steps	The system uses a weak class reference, which might allow an attacker to execute unauthorized code. The program references a user-defined class that is not uniquely identified. When .NET loads this weakly identified class, the CLR type loader searches for the class in the following locations in the specified order: 1. If the assembly of the type is known, the loader searches the configuration file's redirect locations, GAC, the current assembly using configuration information, and the application base directory 2. If the assembly is unknown, the loader searches the current assembly, mscorlib, and the location returned by the TypeResolve event handler 3. This CLR search order can be modified with hooks such as the Type Forwarding mechanism and the AppDomain.TypeResolve event If an attacker exploits the CLR search order by creating an alternative class with the same name and placing it in an alternative location that the CLR will load first, the CLR will unintentionally execute the attacker-supplied code

Example

The `<behaviorExtensions/>` element of the WCF configuration file below instructs WCF to add a custom behavior class to a particular WCF extension.

```
<system.serviceModel>
    <extensions>
        <behaviorExtensions>
            <add name=""myBehavior"" type=""MyBehavior"" />
        </behaviorExtensions>
    </extensions>
</system.serviceModel>
```

Using fully qualified (strong) names uniquely identifies a type and further increases security of your system. Use fully qualified assembly names when registering types in the machine.config and app.config files.

Example

The `<behaviorExtensions/>` element of the WCF configuration file below instructs WCF to add strongly-referenced custom behavior class to a particular WCF extension.

```
<system.serviceModel>
    <extensions>
        <behaviorExtensions>
            <add name=""myBehavior"" type=""Microsoft.ServiceModel.Samples.MyBehaviorSection, MyBehavior,
            Version=1.0.0.0, Culture=neutral, PublicKeyToken=null"" />
        </behaviorExtensions>
    </extensions>
</system.serviceModel>
```

WCF-Implement Authorization control

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, NET Framework 3
Attributes	N/A
References	MSDN, Fortify Kingdom
Steps	This service does not use an authorization control. When a client calls a particular WCF service, WCF provides various authorization schemes that verify that the caller has permission to execute the service method on the server. If authorization controls are not enabled for WCF services, an authenticated user can achieve privilege escalation.

Example

The following configuration instructs WCF to not check the authorization level of the client when executing the service:

```
<behaviors>
    <serviceBehaviors>
        <behavior>
            ...
            <serviceAuthorization principalPermissionMode=""None"" />
        </behavior>
    </serviceBehaviors>
</behaviors>
```

Use a service authorization scheme to verify that the caller of the service method is authorized to do so. WCF provides two modes and allows the definition of a custom authorization scheme. The UseWindowsGroups mode uses Windows roles and users and the UseAspNetRoles mode uses an ASP.NET role provider, such as SQL Server, to authenticate.

Example

The following configuration instructs WCF to make sure that the client is part of the Administrators group before executing the Add service:

```
<behaviors>
    <serviceBehaviors>
        <behavior>
            ...
            <serviceAuthorization principalPermissionMode=""UseWindowsGroups"" />
        </behavior>
    </serviceBehaviors>
</behaviors>
```

The service is then declared as the following:

```
[PrincipalPermission(SecurityAction.Demand,
Role = ""Builtin\\Administrators"")]
public double Add(double n1, double n2)
{
double result = n1 + n2;
return result;
}
```

Implement proper authorization mechanism in ASP.NET Web API

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	Generic, MVC5
Attributes	N/A, Identity Provider - ADFS, Identity Provider - Azure AD
References	Authentication and Authorization in ASP.NET Web API
Steps	Role information for the application users can be derived from Azure AD or ADFS claims if the application relies on them as Identity provider or the application itself might provided it. In any of these cases, the custom authorization implementation should validate the user role information. Role information for the application users can be derived from Azure AD or ADFS claims if the application relies on them as Identity provider or the application itself might provided it. In any of these cases, the custom authorization implementation should validate the user role information.

Example

```
[AttributeUsage(AttributeTargets.Class | AttributeTargets.Method, Inherited = true, AllowMultiple = true)]
public class ApiAuthorizeAttribute : System.Web.Http.AuthorizeAttribute
{
        public async override Task OnAuthorizationAsync(HttpActionContext actionContext, CancellationToken
cancellationToken)
        {
            if (actionContext == null)
            {
                throw new Exception();
            }

            if (!string.IsNullOrEmpty(base.Roles))
            {
                bool isAuthorized = ValidateRoles(actionContext);
                if (!isAuthorized)
                {
                    HandleUnauthorizedRequest(actionContext);
                }
            }

            base.OnAuthorization(actionContext);
        }

public bool ValidateRoles(actionContext)
{
   //Authorization logic here; returns true or false
}

}
```

All the controllers and action methods which needs to protected should be decorated with above attribute.

```
[ApiAuthorize]
public class CustomController : ApiController
{
     //Application code goes here
}
```

Perform authorization checks in the device if it supports various actions that require different permission levels

TITLE	DETAILS
Component	IoT Device
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	The Device should authorize the caller to check if the caller has the required permissions to perform the action requested. For e.g. Lets say the device is a Smart Door Lock that can be monitored from the cloud, plus it provides functionalities like Remotely locking the door. The Smart Door Lock provides unlocking functionality only when someone physically comes near the door with a Card. In this case, the implementation of the remote command and control should be done in such a way that it does not provide any functionality to unlock the door as the cloud gateway is not authorized to send a command to unlock the door.

Perform authorization checks in the Field Gateway if it supports various actions that require different permission levels

TITLE	DETAILS
Component	IoT Field Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	The Field Gateway should authorize the caller to check if the caller has the required permissions to perform the action requested. For e.g. there should be different permissions for an admin user interface/API used to configure a field gateway v/s devices that connect to it.

Security Frame: Communication Security | Mitigations

1/24/2018 • 14 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Azure Event Hub	• Secure communication to Event Hub using SSL/TLS
Dynamics CRM	• Check service account privileges and check that the custom Services or ASP.NET Pages respect CRM's security
Azure Data Factory	• Use Data management gateway while connecting On Prem SQL Server to Azure Data Factory
Identity Server	• Ensure that all traffic to Identity Server is over HTTPS connection
Web Application	• Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections • Configure SSL certificate for custom domain in Azure App Service • Force all traffic to Azure App Service over HTTPS connection • Enable HTTP Strict Transport Security (HSTS)
Database	• Ensure SQL server connection encryption and certificate validation • Force Encrypted communication to SQL server
Azure Storage	• Ensure that communication to Azure Storage is over HTTPS • Validate MD5 hash after downloading blob if HTTPS cannot be enabled • Use SMB 3.0 compatible client to ensure in-transit data encryption to Azure File Shares
Mobile Client	• Implement Certificate Pinning
WCF	• Enable HTTPS - Secure Transport channel • WCF: Set Message security Protection level to EncryptAndSign • WCF: Use a least-privileged account to run your WCF service
Web API	• Force all traffic to Web APIs over HTTPS connection

PRODUCT/SERVICE	ARTICLE
Azure Redis Cache	• Ensure that communication to Azure Redis Cache is over SSL
IoT Field Gateway	• Secure Device to Field Gateway communication
IoT Cloud Gateway	• Secure Device to Cloud Gateway communication using SSL/TLS

Secure communication to Event Hub using SSL/TLS

TITLE	DETAILS
Component	Azure Event Hub
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Event Hubs authentication and security model overview
Steps	Secure AMQP or HTTP connections to Event Hub using SSL/TLS

Check service account privileges and check that the custom Services or ASP.NET Pages respect CRM's security

TITLE	DETAILS
Component	Dynamics CRM
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Check service account privileges and check that the custom Services or ASP.NET Pages respect CRM's security

Use Data management gateway while connecting On Prem SQL Server to Azure Data Factory

TITLE	DETAILS
Component	Azure Data Factory
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Linked Service Types - Azure and On Prem
References	Moving data between On Prem and Azure Data Factory, Data management gateway
Steps	The Data Management Gateway (DMG) tool is required to connect to data sources which are protected behind corpnet or a firewall. 1. Locking down the machine isolates the DMG tool and prevents malfunctioning programs from damaging or snooping on the data source machine. (E.g. latest updates must be installed, enable minimum required ports, controlled accounts provisioning, auditing enabled, disk encryption enabled etc.) 2. Data Gateway key must be rotated at frequent intervals or whenever the DMG service account password renews 3. Data transits through Link Service must be encrypted

Ensure that all traffic to Identity Server is over HTTPS connection

TITLE	DETAILS
Component	Identity Server
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	IdentityServer3 - Keys, Signatures and Cryptography, IdentityServer3 - Deployment
Steps	By default, IdentityServer requires all incoming connections to come over HTTPS. It is absolutely mandatory that communication with IdentityServer is done over secured transports only. There are certain deployment scenarios like SSL offloading where this requirement can be relaxed. See the Identity Server deployment page in the references for more information.

Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Applications that use SSL, TLS, or DTLS must fully verify the X.509 certificates of the entities they connect to. This includes verification of the certificates for: • Domain name • Validity dates (both beginning and expiration dates) • Revocation status • Usage (for example, Server Authentication for servers, Client Authentication for clients) • Trust chain. Certificates must chain to a root certification authority (CA) that is trusted by the platform or explicitly configured by the administrator • Key length of certificate's public key must be >2048 bits • Hashing algorithm must be SHA256 and above

Configure SSL certificate for custom domain in Azure App Service

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	EnvironmentType - Azure
References	Enable HTTPS for an app in Azure App Service
Steps	By default, Azure already enables HTTPS for every app with a wildcard certificate for the *.azurewebsites.net domain. However, like all wildcard domains, it is not as secure as using a custom domain with own certificate Refer. It is recommended to enable SSL for the custom domain which the deployed app will be accessed through

Force all traffic to Azure App Service over HTTPS connection

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	EnvironmentType - Azure
References	Enforce HTTPS on Azure App Service
Steps	Though Azure already enables HTTPS for Azure app services with a wildcard certificate for the domain *.azurewebsites.net, it do not enforce HTTPS. Visitors may still access the app using HTTP, which may compromise the app's security and hence HTTPS has to be enforced explicitly. ASP.NET MVC applications should use the RequireHttps filter that forces an unsecured HTTP request to be re-sent over HTTPS. Alternatively, the URL Rewrite module, which is included with Azure App Service can be used to enforce HTTPS. URL Rewrite module enables developers to define rules that are applied to incoming requests before the requests are handed to your application. URL Rewrite rules are defined in a web.config file stored in the root of the application

Example

The following example contains a basic URL Rewrite rule that forces all incoming traffic to use HTTPS

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <system.webServer>
    <rewrite>
      <rules>
        <rule name="Force HTTPS" enabled="true">
          <match url="(.*)" ignoreCase="false" />
          <conditions>
            <add input="{HTTPS}" pattern="off" />
          </conditions>
          <action type="Redirect" url="https://{HTTP_HOST}/{R:1}" appendQueryString="true"
redirectType="Permanent" />
        </rule>
      </rules>
    </rewrite>
  </system.webServer>
</configuration>
```

This rule works by returning an HTTP status code of 301 (permanent redirect) when the user requests a page using HTTP. The 301 redirects the request to the same URL as the visitor requested, but replaces the HTTP portion of the request with HTTPS. For example, HTTP://contoso.com would be redirected to HTTPS://contoso.com.

Enable HTTP Strict Transport Security (HSTS)

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	OWASP HTTP Strict Transport Security Cheat Sheet
Steps	HTTP Strict Transport Security (HSTS) is an opt-in security enhancement that is specified by a web application through the use of a special response header. Once a supported browser receives this header that browser will prevent any communications from being sent over HTTP to the specified domain and will instead send all communications over HTTPS. It also prevents HTTPS click through prompts on browsers. To implement HSTS, the following response header has to be configured for a website globally, either in code or in config. Strict-Transport-Security: max-age=300; includeSubDomains HSTS addresses the following threats: • User bookmarks or manually types http://example.com and is subject to a man-in-the-middle attacker: HSTS automatically redirects HTTP requests to HTTPS for the target domain • Web application that is intended to be purely HTTPS inadvertently contains HTTP links or serves content over HTTP: HSTS automatically redirects HTTP requests to HTTPS for the target domain • A man-in-the-middle attacker attempts to intercept traffic from a victim user using an invalid certificate and hopes the user will accept the bad certificate: HSTS does not allow a user to override the invalid certificate message

Ensure SQL server connection encryption and certificate validation

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure
Attributes	SQL Version - V12
References	Best Practices on Writing Secure Connection Strings for SQL Database

TITLE	DETAILS
Steps	All communications between SQL Database and a client application are encrypted using Secure Sockets Layer (SSL) at all times. SQL Database doesn't support unencrypted connections. To validate certificates with application code or tools, explicitly request an encrypted connection and do not trust the server certificates. If your application code or tools do not request an encrypted connection, they will still receive encrypted connections However, they may not validate the server certificates and thus will be susceptible to "man in the middle" attacks. To validate certificates with ADO.NET application code, set `Encrypt=True` and `TrustServerCertificate=False` in the database connection string. To validate certificates via SQL Server Management Studio, open the Connect to Server dialog box. Click Encrypt connection on the Connection Properties tab

Force Encrypted communication to SQL server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	OnPrem
Attributes	SQL Version - MsSQL2016, SQL Version - MsSQL2012, SQL Version - MsSQL2014
References	Enable Encrypted Connections to the Database Engine
Steps	Enabling SSL encryption increases the security of data transmitted across networks between instances of SQL Server and applications.

Ensure that communication to Azure Storage is over HTTPS

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Azure Storage Transport-Level Encryption – Using HTTPS

TITLE	DETAILS
Steps	To ensure the security of Azure Storage data in-transit, always use the HTTPS protocol when calling the REST APIs or accessing objects in storage. Also, Shared Access Signatures, which can be used to delegate access to Azure Storage objects, include an option to specify that only the HTTPS protocol can be used when using Shared Access Signatures, ensuring that anybody sending out links with SAS tokens will use the proper protocol.

Validate MD5 hash after downloading blob if HTTPS cannot be enabled

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	StorageType - Blob
References	Windows Azure Blob MD5 Overview
Steps	Windows Azure Blob service provides mechanisms to ensure data integrity both at the application and transport layers. If for any reason you need to use HTTP instead of HTTPS and you are working with block blobs, you can use MD5 checking to help verify the integrity of the blobs being transferred This will help with protection from network/transport layer errors, but not necessarily with intermediary attacks. If you can use HTTPS, which provides transport level security, then using MD5 checking is redundant and unnecessary.

Use SMB 3.0 compatible client to ensure in-transit data encryption to Azure File shares

TITLE	DETAILS
Component	Mobile Client
SDL Phase	Build
Applicable Technologies	Generic
Attributes	StorageType - File
References	Azure File Storage, Azure File Storage SMB Support for Windows Clients

TITLE	DETAILS
Steps	Azure File Storage supports HTTPS when using the REST API, but is more commonly used as an SMB file share attached to a VM. SMB 2.1 does not support encryption, so connections are only allowed within the same region in Azure. However, SMB 3.0 supports encryption, and can be used with Windows Server 2012 R2, Windows 8, Windows 8.1, and Windows 10, allowing cross-region access and even access on the desktop.

Implement Certificate Pinning

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic, Windows Phone
Attributes	N/A
References	Certificate and Public Key Pinning
Steps	Certificate pinning defends against Man-In-The-Middle (MITM) attacks. Pinning is the process of associating a host with their expected X509 certificate or public key. Once a certificate or public key is known or seen for a host, the certificate or public key is associated or 'pinned' to the host. Thus, when an adversary attempts to do SSL MITM attack, during SSL handshake the key from attacker's server will be different from the pinned certificate's key, and the request will be discarded, thus preventing MITM Certificate pinning can be achieved by implementing ServicePointManager's `ServerCertificateValidationCallback` delegate.

Example

```
using System;
using System.Net;
using System.Net.Security;
using System.Security.Cryptography;

namespace CertificatePinningExample
{
    class CertificatePinningExample
    {
        /* Note: In this example, we're hardcoding a the certificate's public key and algorithm for
           demonstration purposes. In a real-world application, this should be stored in a secure
           configuration area that can be updated as needed. */

        private static readonly string PINNED_ALGORITHM = "RSA";

        private static readonly string PINNED_PUBLIC_KEY = "3082010A0282010100B0E75B7CBE56D31658EF79B3A1" +
            "294D506A88DFCDD603F6EF15E7F5BC8DF32291EC5082B828A158E905FE6A83EE044A48258807FAC3D6356AF09B2" +
            "3EDAB15D00507B70DB08DB9A20C7D1201417B3071A346D663A241061C151B6EC5B5B4ECCCDCDBEA24F051962809" +
            "FEC499BF2D093C06E3BDA7D0BB83CDC1C2C6660B8ECB2EA30A685ADE2DC83C88314010FFC7F4F0F895EDDBE5C02" +
            "A8F78E50B708E0A0E8984A9AA536BCE61A0C31D895425C6FEE5A5648158EE7C4F0693C439AE010EF83CA8155750" +
            "09B17537C29F86071E5DD8CA50EBD8A409494F479B07574D83EDCE6F68A8F7D40447471D05BC3F5EAD7862FA748" +
            "EA3C92A60A128344B1CEF7A0B0D94E50203010001";


        public static void Main(string[] args)
        {
            HttpWebRequest request = (HttpWebRequest)WebRequest.Create("https://azure.microsoft.com");
            request.ServerCertificateValidationCallback = (sender, certificate, chain, sslPolicyErrors) =>
            {
                if (certificate == null || sslPolicyErrors != SslPolicyErrors.None)
                {
                    // Error getting certificate or the certificate failed basic validation
                    return false;
                }

                var targetKeyAlgorithm = new Oid(certificate.GetKeyAlgorithm()).FriendlyName;
                var targetPublicKey = certificate.GetPublicKeyString();

                if (targetKeyAlgorithm == PINNED_ALGORITHM &&
                    targetPublicKey == PINNED_PUBLIC_KEY)
                {
                    // Success, the certificate matches the pinned value.
                    return true;
                }
                // Reject, either the key or the algorithm does not match the expected value.
                return false;
            };

            try
            {
                var response = (HttpWebResponse)request.GetResponse();
                Console.WriteLine($"Success, HTTP status code: {response.StatusCode}");
            }
            catch(Exception ex)
            {
                Console.WriteLine($"Failure, {ex.Message}");
            }
            Console.WriteLine("Press any key to end.");
            Console.ReadKey();
        }
    }
}
```

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	NET Framework 3
Attributes	N/A
References	MSDN, Fortify Kingdom
Steps	The application configuration should ensure that HTTPS is used for all access to sensitive information. • **EXPLANATION:** If an application handles sensitive information and does not use message-level encryption, then it should only be allowed to communicate over an encrypted transport channel. • **RECOMMENDATIONS:** Ensure that HTTP transport is disabled and enable HTTPS transport instead. For example, replace the `<httpTransport/>` with `<httpsTransport/>` tag. Do not rely on a network configuration (firewall) to guarantee that the application can only be accessed over a secure channel. From a philosophical point of view, the application should not depend on the network for its security. From a practical point of view, the people responsible for securing the network do not always track the security requirements of the application as they evolve.

WCF: Set Message security Protection level to EncryptAndSign

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework 3
Attributes	N/A
References	MSDN

TITLE	DETAILS
Steps	• **EXPLANATION:** When Protection level is set to "none" it will disable message protection. Confidentiality and integrity is achieved with appropriate level of setting. • **RECOMMENDATIONS:** ◦ when `Mode=None` - Disables message protection ◦ when `Mode=Sign` - Signs but does not encrypt the message; should be used when data integrity is important ◦ when `Mode=EncryptAndSign` - Signs and encrypts the message Consider turning off encryption and only signing your message when you just need to validate the integrity of the information without concerns of confidentiality. This may be useful for operations or service contracts in which you need to validate the original sender but no sensitive data is transmitted. When reducing the protection level, be careful that the message does not contain any personally identifiable information (PII).

Example

Configuring the service and the operation to only sign the message is shown in the following examples. Service Contract Example of `ProtectionLevel.Sign`: The following is an example of using ProtectionLevel.Sign at the Service Contract level:

```
[ServiceContract(Protection Level=ProtectionLevel.Sign]
public interface IService
  {
  string GetData(int value);
  }
```

Example

Operation Contract Example of `ProtectionLevel.Sign` (for Granular Control): The following is an example of using `ProtectionLevel.Sign` at the OperationContract level:

```
[OperationContract(ProtectionLevel=ProtectionLevel.Sign]
string GetData(int value);
```

WCF: Use a least-privileged account to run your WCF service

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework 3
Attributes	N/A

TITLE	DETAILS
References	MSDN
Steps	• **EXPLANATION:** Do not run WCF services under admin or high privilege account. in case of services compromise it will result in high impact. • **RECOMMENDATIONS:** Use a least-privileged account to host your WCF service because it will reduce your application's attack surface and reduce the potential damage if you are attacked. If the service account requires additional access rights on infrastructure resources such as MSMQ, the event log, performance counters, and the file system, appropriate permissions should be given to these resources so that the WCF service can run successfully. If your service needs to access specific resources on behalf of the original caller, use impersonation and delegation to flow the caller's identity for a downstream authorization check. In a development scenario, use the local network service account, which is a special built-in account that has reduced privileges. In a production scenario, create a least-privileged custom domain service account.

Force all traffic to Web APIs over HTTPS connection

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	N/A
References	Enforcing SSL in a Web API Controller
Steps	If an application has both an HTTPS and an HTTP binding, clients can still use HTTP to access the site. To prevent this, use an action filter to ensure that requests to protected APIs are always over HTTPS.

Example

The following code shows a Web API authentication filter that checks for SSL:

```
public class RequireHttpsAttribute : AuthorizationFilterAttribute
{
    public override void OnAuthorization(HttpActionContext actionContext)
    {
        if (actionContext.Request.RequestUri.Scheme != Uri.UriSchemeHttps)
        {
            actionContext.Response = new HttpResponseMessage(System.Net.HttpStatusCode.Forbidden)
            {
                ReasonPhrase = "HTTPS Required"
            };
        }
        else
        {
            base.OnAuthorization(actionContext);
        }
    }
}
```

Add this filter to any Web API actions that require SSL:

```
public class ValuesController : ApiController
{
    [RequireHttps]
    public HttpResponseMessage Get() { ... }
}
```

Ensure that communication to Azure Redis Cache is over SSL

TITLE	DETAILS
Component	Azure Redis Cache
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Azure Redis SSL support
Steps	Redis server does not support SSL out of the box, but Azure Redis Cache does. If you are connecting to Azure Redis Cache and your client supports SSL, like StackExchange.Redis, then you should use SSL. By default non-SSL port is disabled for new Azure Redis Cache instances. Ensure that the secure defaults are not changed unless there is a dependency on SSL support for redis clients.

Please note that Redis is designed to be accessed by trusted clients inside trusted environments. This means that usually it is not a good idea to expose the Redis instance directly to the internet or, in general, to an environment where untrusted clients can directly access the Redis TCP port or UNIX socket.

Secure Device to Field Gateway communication

TITLE	DETAILS
Component	IoT Field Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	For IP based devices, the communication protocol could typically be encapsulated in a SSL/TLS channel to protect data in transit. For other protocols that do not support SSL/TLS investigate if there are secure versions of the protocol that provide security at transport or message layer.

Secure Device to Cloud Gateway communication using SSL/TLS

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Choose your Communication Protocol
Steps	Secure HTTP/AMQP or MQTT protocols using SSL/TLS.

Security Frame: Configuration Management | Mitigations

1/24/2019 • 20 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Web Application	• Implement Content Security Policy (CSP), and disable inline javascript • Enable browser's XSS filter • ASP.NET applications must disable tracing and debugging prior to deployment • Access third-party javascripts from trusted sources only • Ensure that authenticated ASP.NET pages incorporate UI Redressing or click-jacking defenses • Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web Applications • Enable ValidateRequest attribute on ASP.NET Pages • Use locally-hosted latest versions of JavaScript libraries • Disable automatic MIME sniffing • Remove standard server headers on Windows Azure Web Sites to avoid fingerprinting
Database	• Configure a Windows Firewall for Database Engine Access
Web API	• Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web API • Encrypt sections of Web API's configuration files that contain sensitive data
IoT Device	• Ensure that all admin interfaces are secured with strong credentials • Ensure that unknown code cannot execute on devices • Encrypt OS and additional partitions of IoT Device with bit-locker • Ensure that only the minimum services/features are enabled on devices
IoT Field Gateway	• Encrypt OS and additional partitions of IoT Field Gateway with bit-locker • Ensure that the default login credentials of the field gateway are changed during installation
IoT Cloud Gateway	• Ensure that the Cloud Gateway implements a process to keep the connected devices firmware up to date

PRODUCT/SERVICE	ARTICLE
Machine Trust Boundary	• Ensure that devices have end-point security controls configured as per organizational policies
Azure Storage	• Ensure secure management of Azure storage access keys • Ensure that only trusted origins are allowed if CORS is enabled on Azure storage
WCF	• Enable WCF's service throttling feature • WCF-Information disclosure through metadata

Implement Content Security Policy (CSP), and disable inline javascript

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	An introduction to Content Security Policy, Content Security Policy Reference, Security features, Introduction to content security policy, Can I use CSP?

TITLE	DETAILS
Steps	Content Security Policy (CSP) is a defense-in-depth security mechanism, a W3C standard, that enables web application owners to have control on the content embedded in their site. CSP is added as an HTTP response header on the web server and is enforced on the client side by browsers. It is a whitelist-based policy - a website can declare a set of trusted domains from which active content such as JavaScript can be loaded. CSP provides the following security benefits: • **Protection against XSS:** If a page is vulnerable to XSS, an attacker can exploit it in 2 ways: ◦ Inject `<script>malicious code</script>`. This exploit will not work due to CSP's Base Restriction-1 ◦ Inject `<script src="http://attacker.com/maliciousCode.js"/>`. This exploit will not work since the attacker controlled domain will not be in CSP's whitelist of domains • **Control over data exfiltration:** If any malicious content on a webpage attempts to connect to an external website and steal data, the connection will be aborted by CSP. This is because the target domain will not be in CSP's whitelist • **Defense against click-jacking:** click-jacking is an attack technique using which an adversary can frame a genuine website and force users to click on UI elements. Currently defense against click-jacking is achieved by configuring a response header- X-Frame-Options. Not all browsers respect this header and going forward CSP will be a standard way to defend against click-jacking • **Real-time attack reporting:** If there is an injection attack on a CSP-enabled website, browsers will automatically trigger a notification to an endpoint configured on the webserver. This way, CSP serves as a real-time warning system.

Example

Example policy:

```
Content-Security-Policy: default-src 'self'; script-src 'self' www.google-analytics.com
```

This policy allows scripts to load only from the web application's server and google analytics server. Scripts loaded from any other site will be rejected. When CSP is enabled on a website, the following features are automatically disabled to mitigate XSS attacks.

Example

Inline scripts will not execute. Following are examples of inline scripts

```
<script> some Javascript code </script>
Event handling attributes of HTML tags (e.g., <button onclick="function(){}">
javascript:alert(1);
```

Example

Strings will not be evaluated as code.

```
Example: var str="alert(1)"; eval(str);
```

Enable browser's XSS filter

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	XSS Protection Filter
Steps	X-XSS-Protection response header configuration controls the browser's cross site script filter. This response header can have following values: • `0:` This will disable the filter • `1: Filter enabled` If a cross-site scripting attack is detected, in order to stop the attack, the browser will sanitize the page • `1: mode=block : Filter enabled` . Rather than sanitize the page, when a XSS attack is detected, the browser will prevent rendering of the page • `1: report=http://[YOURDOMAIN]/your_report_URI : Filter enabled` . The browser will sanitize the page and report the violation. This is a Chromium function utilizing CSP violation reports to send details to a URI of your choice. The last 2 options are considered safe values.

ASP.NET applications must disable tracing and debugging prior to deployment

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	ASP.NET Debugging Overview, ASP.NET Tracing Overview, How to: Enable Tracing for an ASP.NET Application, How to: Enable Debugging for ASP.NET Applications
Steps	When tracing is enabled for the page, every browser requesting it also obtains the trace information that contains data about internal server state and workflow. That information could be security sensitive. When debugging is enabled for the page, errors happening on the server result in a full stack trace data presented to the browser. That data may expose security-sensitive information about the server's workflow.

Access third-party javascripts from trusted sources only

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	third-party JavaScripts should be referenced only from trusted sources. The reference endpoints should always be on SSL.

Ensure that authenticated ASP.NET pages incorporate UI Redressing or click-jacking defenses

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	OWASP click-jacking Defense Cheat Sheet, IE Internals - Combating click-jacking With X-Frame-Options

TITLE	DETAILS
Steps	click-jacking, also known as a "UI redress attack", is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they were intending to click on the top-level page. This layering is achieved by crafting a malicious page with an iframe, which loads the victim's page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to another page, most likely owned by another application, domain, or both. To prevent click-jacking attacks, set the proper X-Frame-Options HTTP response headers that instruct the browser to not allow framing from other domains

Example

The X-FRAME-OPTIONS header can be set via IIS web.config. Web.config code snippet for sites that should never be framed:

```
<system.webServer>
    <httpProtocol>
        <customHeader>
            <add name="X-FRAME-OPTIONS" value="DENY"/>
        </customHeaders>
    </httpProtocol>
</system.webServer>
```

Example

Web.config code for sites that should only be framed by pages in the same domain:

```
<system.webServer>
    <httpProtocol>
        <customHeader>
            <add name="X-FRAME-OPTIONS" value="SAMEORIGIN"/>
        </customHeaders>
    </httpProtocol>
</system.webServer>
```

Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web Applications

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Web Forms, MVC5
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Browser security prevents a web page from making AJAX requests to another domain. This restriction is called the same-origin policy, and prevents a malicious site from reading sensitive data from another site. However, sometimes it might be required to expose APIs securely which other sites can consume. Cross Origin Resource Sharing (CORS) is a W3C standard that allows a server to relax the same-origin policy. Using CORS, a server can explicitly allow some cross-origin requests while rejecting others. CORS is safer and more flexible than earlier techniques such as JSONP. At its core, enabling CORS translates to adding a few HTTP response headers (Access-Control-*) to the web application and this can be done in a couple of ways.

Example

If access to Web.config is available, then CORS can be added through the following code:

```
<system.webServer>
    <httpProtocol>
      <customHeaders>
        <clear />
        <add name="Access-Control-Allow-Origin" value="http://example.com" />
      </customHeaders>
    </httpProtocol>
```

Example

If access to web.config is not available, then CORS can be configured by adding the following CSharp code:

```
HttpContext.Response.AppendHeader("Access-Control-Allow-Origin", "http://example.com")
```

Please note that it is critical to ensure that the list of origins in "Access-Control-Allow-Origin" attribute is set to a finite and trusted set of origins. Failing to configure this inappropriately (e.g., setting the value as '*') will allow malicious sites to trigger cross origin requests to the web application >without any restrictions, thereby making the application vulnerable to CSRF attacks.

Enable ValidateRequest attribute on ASP.NET Pages

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Web Forms, MVC5
Attributes	N/A
References	Request Validation - Preventing Script Attacks

TITLE	DETAILS
Steps	Request validation, a feature of ASP.NET since version 1.1, prevents the server from accepting content containing un-encoded HTML. This feature is designed to help prevent some script-injection attacks whereby client script code or HTML can be unknowingly submitted to a server, stored, and then presented to other users. We still strongly recommend that you validate all input data and HTML encode it when appropriate. Request validation is performed by comparing all input data to a list of potentially dangerous values. If a match occurs, ASP.NET raises an `HttpRequestValidationException`. By default, Request Validation feature is enabled.

Example

However, this feature can be disabled at page level:

```
<%@ Page validateRequest="false" %>
```

or, at application level

```
<configuration>
   <system.web>
      <pages validateRequest="false" />
   </system.web>
</configuration>
```

Please note that Request Validation feature is not supported, and is not part of MVC6 pipeline.

Use locally-hosted latest versions of JavaScript libraries

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Developers using standard JavaScript libraries like JQuery must use approved versions of common JavaScript libraries that do not contain known security flaws. A good practice is to use the most latest version of the libraries, since they contain security fixes for known vulnerabilities in their older versions. If the most recent release cannot be used due to compatibility reasons, the below minimum versions should be used. Acceptable minimum versions: • **JQuery** ◦ JQuery 1.7.1 ◦ JQueryUI 1.10.0 ◦ JQuery Validate 1.9 ◦ JQuery Mobile 1.0.1 ◦ JQuery Cycle 2.99 ◦ JQuery DataTables 1.9.0 • **Ajax Control Toolkit** ◦ Ajax Control Toolkit 40412 • **ASP.NET Web Forms and Ajax** ◦ ASP.NET Web Forms and Ajax 4 ◦ ASP.NET Ajax 3.5 • **ASP.NET MVC** ◦ ASP.NET MVC 3.0 Never load any JavaScript library from external sites such as public CDNs

Disable automatic MIME sniffing

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	IE8 Security Part V: Comprehensive Protection, MIME type
Steps	The X-Content-Type-Options header is an HTTP header that allows developers to specify that their content should not be MIME-sniffed. This header is designed to mitigate MIME-Sniffing attacks. For each page that could contain user controllable content, you must use the HTTP Header X-Content-Type-Options:nosniff. To enable the required header globally for all pages in the application, you can do one of the following

Example

Add the header in the web.config file if the application is hosted by Internet Information Services (IIS) 7 onwards.

```
<system.webServer>
<httpProtocol>
<customHeaders>
<add name="X-Content-Type-Options" value="nosniff"/>
</customHeaders>
</httpProtocol>
</system.webServer>
```

Example

Add the header through the global Application_BeginRequest

```
void Application_BeginRequest(object sender, EventArgs e)
{
this.Response.Headers["X-Content-Type-Options"] = "nosniff";
}
```

Example

Implement custom HTTP module

```
public class XContentTypeOptionsModule : IHttpModule
{
#region IHttpModule Members
public void Dispose()
{
}
public void Init(HttpApplication context)
{
context.PreSendRequestHeaders += newEventHandler(context_PreSendRequestHeaders);
}
#endregion
void context_PreSendRequestHeaders(object sender, EventArgs e)
{
HttpApplication application = sender as HttpApplication;
if (application == null)
  return;
if (application.Response.Headers["X-Content-Type-Options "] != null)
  return;
application.Response.Headers.Add("X-Content-Type-Options ", "nosniff");
}
}
```

Example

You can enable the required header only for specific pages by adding it to individual responses:

```
this.Response.Headers["X-Content-Type-Options"] = "nosniff";
```

Remove standard server headers on Windows Azure Web Sites to avoid fingerprinting

TITLE	DETAILS
Component	Web Application
SDL Phase	Build

TITLE	DETAILS
Applicable Technologies	Generic
Attributes	EnvironmentType - Azure
References	Removing standard server headers on Windows Azure Web Sites
Steps	Headers such as Server, X-Powered-By, X-AspNet-Version reveal information about the server and the underlying technologies. It is recommended to suppress these headers thereby preventing fingerprinting the application

Configure a Windows Firewall for Database Engine Access

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure, OnPrem
Attributes	N/A, SQL Version - V12
References	How to configure an Azure SQL database firewall, Configure a Windows Firewall for Database Engine Access
Steps	Firewall systems help prevent unauthorized access to computer resources. To access an instance of the SQL Server Database Engine through a firewall, you must configure the firewall on the computer running SQL Server to allow access

Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web API

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC 5
Attributes	N/A
References	Enabling Cross-Origin Requests in ASP.NET Web API 2, ASP.NET Web API - CORS Support in ASP.NET Web API 2

TITLE	DETAILS
Steps	Browser security prevents a web page from making AJAX requests to another domain. This restriction is called the same-origin policy, and prevents a malicious site from reading sensitive data from another site. However, sometimes it might be required to expose APIs securely which other sites can consume. Cross Origin Resource Sharing (CORS) is a W3C standard that allows a server to relax the same-origin policy. Using CORS, a server can explicitly allow some cross-origin requests while rejecting others. CORS is safer and more flexible than earlier techniques such as JSONP.

Example

In the App_Start/WebApiConfig.cs, add the following code to the WebApiConfig.Register method

```
using System.Web.Http;
namespace WebService
{
    public static class WebApiConfig
    {
        public static void Register(HttpConfiguration config)
        {
            // New code
            config.EnableCors();

            config.Routes.MapHttpRoute(
                name: "DefaultApi",
                routeTemplate: "api/{controller}/{id}",
                defaults: new { id = RouteParameter.Optional }
            );
        }
    }
}
```

Example

EnableCors attribute can be applied to action methods in a controller as follows:

```
public class ResourcesController : ApiController
{
  [EnableCors("http://localhost:55912", // Origin
              null,                     // Request headers
              "GET",                    // HTTP methods
              "bar",                    // Response headers
              SupportsCredentials=true  // Allow credentials
  )]
  public HttpResponseMessage Get(int id)
  {
    var resp = Request.CreateResponse(HttpStatusCode.NoContent);
    resp.Headers.Add("bar", "a bar value");
    return resp;
  }
  [EnableCors("http://localhost:55912",       // Origin
              "Accept, Origin, Content-Type", // Request headers
              "PUT",                          // HTTP methods
              PreflightMaxAge=600             // Preflight cache duration
  )]
  public HttpResponseMessage Put(Resource data)
  {
    return Request.CreateResponse(HttpStatusCode.OK, data);
  }
  [EnableCors("http://localhost:55912",       // Origin
              "Accept, Origin, Content-Type", // Request headers
              "POST",                         // HTTP methods
              PreflightMaxAge=600             // Preflight cache duration
  )]
  public HttpResponseMessage Post(Resource data)
  {
    return Request.CreateResponse(HttpStatusCode.OK, data);
  }
}
```

Please note that it is critical to ensure that the list of origins in EnableCors attribute is set to a finite and trusted set of origins. Failing to configure this inappropriately (e.g., setting the value as '*') will allow malicious sites to trigger cross origin requests to the API without any restrictions, >thereby making the API vulnerable to CSRF attacks. EnableCors can be decorated at controller level.

Example

To disable CORS on a particular method in a class, the DisableCors attribute can be used as shown below:

```
[EnableCors("http://example.com", "Accept, Origin, Content-Type", "POST")]
public class ResourcesController : ApiController
{
  public HttpResponseMessage Put(Resource data)
  {
    return Request.CreateResponse(HttpStatusCode.OK, data);
  }
  public HttpResponseMessage Post(Resource data)
  {
    return Request.CreateResponse(HttpStatusCode.OK, data);
  }
  // CORS not allowed because of the [DisableCors] attribute
  [DisableCors]
  public HttpResponseMessage Delete(int id)
  {
    return Request.CreateResponse(HttpStatusCode.NoContent);
  }
}
```

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC 6
Attributes	N/A
References	Enabling Cross-Origin Requests (CORS) in ASP.NET Core 1.0
Steps	In ASP.NET Core 1.0, CORS can be enabled either using middleware or using MVC. When using MVC to enable CORS the same CORS services are used, but the CORS middleware is not.

Approach-1 Enabling CORS with middleware: To enable CORS for the entire application add the CORS middleware to the request pipeline using the UseCors extension method. A cross-origin policy can be specified when adding the CORS middleware using the CorsPolicyBuilder class. There are two ways to do this:

Example

The first is to call UseCors with a lambda. The lambda takes a CorsPolicyBuilder object:

```
public void Configure(IApplicationBuilder app)
{
    app.UseCors(builder =>
        builder.WithOrigins("http://example.com")
        .WithMethods("GET", "POST", "HEAD")
        .WithHeaders("accept", "content-type", "origin", "x-custom-header"));
}
```

Example

The second is to define one or more named CORS policies, and then select the policy by name at run time.

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddCors(options =>
    {
        options.AddPolicy("AllowSpecificOrigin",
            builder => builder.WithOrigins("http://example.com"));
    });
}
public void Configure(IApplicationBuilder app)
{
    app.UseCors("AllowSpecificOrigin");
    app.Run(async (context) =>
    {
        await context.Response.WriteAsync("Hello World!");
    });
}
```

Approach-2 Enabling CORS in MVC: Developers can alternatively use MVC to apply specific CORS per action, per controller, or globally for all controllers.

Example

Per action: To specify a CORS policy for a specific action add the [EnableCors] attribute to the action. Specify the policy name.

```
public class HomeController : Controller
{
    [EnableCors("AllowSpecificOrigin")]
    public IActionResult Index()
    {
        return View();
    }
```

Example

Per controller:

```
[EnableCors("AllowSpecificOrigin")]
public class HomeController : Controller
{
```

Example

Globally:

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddMvc();
    services.Configure<MvcOptions>(options =>
    {
        options.Filters.Add(new CorsAuthorizationFilterFactory("AllowSpecificOrigin"));
    });
}
```

Please note that it is critical to ensure that the list of origins in EnableCors attribute is set to a finite and trusted set of origins. Failing to configure this inappropriately (e.g., setting the value as '*') will allow malicious sites to trigger cross origin requests to the API without any restrictions, >thereby making the API vulnerable to CSRF attacks.

Example

To disable CORS for a controller or action, use the [DisableCors] attribute.

```
[DisableCors]
    public IActionResult About()
    {
        return View();
    }
```

Encrypt sections of Web API's configuration files that contain sensitive data

TITLE	DETAILS
Component	Web API
SDL Phase	Deployment
Applicable Technologies	Generic

TITLE	DETAILS
Attributes	N/A
References	How To: Encrypt Configuration Sections in ASP.NET 2.0 Using DPAPI, Specifying a Protected Configuration Provider, Using Azure Key Vault to protect application secrets
Steps	Configuration files such as the Web.config, appsettings.json are often used to hold sensitive information, including user names, passwords, database connection strings, and encryption keys. If you do not protect this information, your application is vulnerable to attackers or malicious users obtaining sensitive information such as account user names and passwords, database names and server names. Based on the deployment type (azure/on-prem), encrypt the sensitive sections of config files using DPAPI or services like Azure Key Vault.

Ensure that all admin interfaces are secured with strong credentials

TITLE	DETAILS
Component	IoT Device
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Any administrative interfaces that the device or field gateway exposes should be secured using strong credentials. Also, any other exposed interfaces like WiFi, SSH, File shares, FTP should be secured with strong credentials. Default weak passwords should not be used.

Ensure that unknown code cannot execute on devices

TITLE	DETAILS
Component	IoT Device
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Enabling Secure Boot and bit-locker Device Encryption on Windows 10 IoT Core

TITLE	DETAILS
Steps	UEFI Secure Boot restricts the system to only allow execution of binaries signed by a specified authority. This feature prevents unknown code from being executed on the platform and potentially weakening the security posture of it. Enable UEFI Secure Boot and restrict the list of certificate authorities that are trusted for signing code. Sign all code that is deployed on the device using one of the trusted authorities.

Encrypt OS and additional partitions of IoT Device with bit-locker

TITLE	DETAILS
Component	IoT Device
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Windows 10 IoT Core implements a lightweight version of bit-locker Device Encryption, which has a strong dependency on the presence of a TPM on the platform, including the necessary preOS protocol in UEFI that conducts the necessary measurements. These preOS measurements ensure that the OS later has a definitive record of how the OS was launched.Encrypt OS partitions using bit-locker and any additional partitions also in case they store any sensitive data.

Ensure that only the minimum services/features are enabled on devices

TITLE	DETAILS
Component	IoT Device
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Do not enable or turn off any features or services in the OS that is not required for the functioning of the solution. For e.g. if the device does not require a UI to be deployed, install Windows IoT Core in headless mode.

Encrypt OS and additional partitions of IoT Field Gateway with bit-

IriusRisk
Ex. 1002 - Page 250

locker

TITLE	DETAILS
Component	IoT Field Gateway
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Windows 10 IoT Core implements a lightweight version of bit-locker Device Encryption, which has a strong dependency on the presence of a TPM on the platform, including the necessary preOS protocol in UEFI that conducts the necessary measurements. These preOS measurements ensure that the OS later has a definitive record of how the OS was launched.Encrypt OS partitions using bit-locker and any additional partitions also in case they store any sensitive data.

Ensure that the default login credentials of the field gateway are changed during installation

TITLE	DETAILS
Component	IoT Field Gateway
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that the default login credentials of the field gateway are changed during installation

Ensure that the Cloud Gateway implements a process to keep the connected devices firmware up to date

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic

TITLE	DETAILS
Attributes	Gateway choice - Azure IoT Hub
References	IoT Hub Device Management Overview, How to update Device Firmware
Steps	LWM2M is a protocol from the Open Mobile Alliance for IoT Device Management. Azure IoT device management allows to interact with physical devices using device jobs. Ensure that the Cloud Gateway implements a process to routinely keep the device and other configuration data up to date using Azure IoT Hub Device Management.

Ensure that devices have end-point security controls configured as per organizational policies

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that devices have end-point security controls such as bit-locker for disk-level encryption, anti-virus with updated signatures, host based firewall, OS upgrades, group policies etc. are configured as per organizational security policies.

Ensure secure management of Azure storage access keys

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Azure Storage security guide - Managing Your Storage Account Keys

TITLE	DETAILS
Steps	Key Storage: It is recommended to store the Azure Storage access keys in Azure Key Vault as a secret and have the applications retrieve the key from key vault. This is recommended due to the following reasons: • The application will never have the storage key hardcoded in a configuration file, which removes that avenue of somebody getting access to the keys without specific permission • Access to the keys can be controlled using Azure Active Directory. This means an account owner can grant access to the handful of applications that need to retrieve the keys from Azure Key Vault. Other applications will not be able to access the keys without granting them permission specifically • Key Regeneration: It is recommended to have a process in place to regenerate Azure storage access keys for security reasons. Details on why and how to plan for key regeneration are documented in the Azure Storage Security Guide reference article

Ensure that only trusted origins are allowed if CORS is enabled on Azure storage

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	CORS Support for the Azure Storage Services
Steps	Azure Storage allows you to enable CORS -- Cross Origin Resource Sharing. For each storage account, you can specify domains that can access the resources in that storage account. By default, CORS is disabled on all services. You can enable CORS by using the REST API or the storage client library to call one of the methods to set the service policies.

Enable WCF's service throttling feature

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework 3

TITLE	DETAILS
Attributes	N/A
References	MSDN, Fortify Kingdom
Steps	Not placing a limit on the use of system resources could result in resource exhaustion and ultimately a denial of service. • **EXPLANATION:** Windows Communication Foundation (WCF) offers the ability to throttle service requests. Allowing too many client requests can flood a system and exhaust its resources. On the other hand, allowing only a small number of requests to a service can prevent legitimate users from using the service. Each service should be individually tuned to and configured to allow the appropriate amount of resources. • **RECOMMENDATIONS** Enable WCF's service throttling feature and set limits appropriate for your application.

Example

The following is an example configuration with throttling enabled:

```
<system.serviceModel>
  <behaviors>
    <serviceBehaviors>
    <behavior name="Throttled">
    <serviceThrottling maxConcurrentCalls="[YOUR SERVICE VALUE]" maxConcurrentSessions="[YOUR SERVICE VALUE]"
maxConcurrentInstances="[YOUR SERVICE VALUE]" />
  ...
</system.serviceModel>
```

WCF-Information disclosure through metadata

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework 3
Attributes	N/A
References	MSDN, Fortify Kingdom
Steps	Metadata can help attackers learn about the system and plan a form of attack. WCF services can be configured to expose metadata. Metadata gives detailed service description information and should not be broadcast in production environments. The `HttpGetEnabled` / `HttpsGetEnabled` properties of the ServiceMetaData class defines whether a service will expose the metadata

Example

The code below instructs WCF to broadcast a service's metadata

```
ServiceMetadataBehavior smb = new ServiceMetadataBehavior();
smb.HttpGetEnabled = true;
smb.HttpGetUrl = new Uri(EndPointAddress);
Host.Description.Behaviors.Add(smb);
```

Do not broadcast service metadata in a production environment. Set the HttpGetEnabled / HttpsGetEnabled properties of the ServiceMetaData class to false.

Example

The code below instructs WCF to not broadcast a service's metadata.

```
ServiceMetadataBehavior smb = new ServiceMetadataBehavior();
smb.HttpGetEnabled = false;
smb.HttpGetUrl = new Uri(EndPointAddress);
Host.Description.Behaviors.Add(smb);
```

Security Frame: Cryptography | Mitigations

8/22/2017 • 14 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Web Application	• Use only approved symmetric block ciphers and key lengths • Use approved block cipher modes and initialization vectors for symmetric ciphers • Use approved asymmetric algorithms, key lengths, and padding • Use approved random number generators • Do not use symmetric stream ciphers • Use approved MAC/HMAC/keyed hash algorithms • Use only approved cryptographic hash functions
Database	• Use strong encryption algorithms to encrypt data in the database • SSIS packages should be encrypted and digitally signed • Add digital signature to critical database securables • Use SQL server EKM to protect encryption keys • Use AlwaysEncrypted feature if encryption keys should not be revealed to Database engine
IoT Device	• Store Cryptographic Keys securely on IoT Device
IoT Cloud Gateway	• Generate a random symmetric key of sufficient length for authentication to IoT Hub
Dynamics CRM Mobile Client	• Ensure a device management policy is in place that requires a use PIN and allows remote wiping
Dynamics CRM Outlook Client	• Ensure a device management policy is in place that requires a PIN/password/auto lock and encrypts all data (e.g. Bitlocker)
Identity Server	• Ensure that signing keys are rolled over when using Identity Server • Ensure that cryptographically strong client ID, client secret are used in Identity Server

Use only approved symmetric block ciphers and key lengths

TITLE	DETAILS
Component	Web Application

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Products must use only those symmetric block ciphers and associated key lengths which have been explicitly approved by the Crypto Advisor in your organization. Approved symmetric algorithms at Microsoft include the following block ciphers: • For new code AES-128, AES-192, and AES-256 are acceptable • For backward compatibility with existing code, three-key 3DES is acceptable • For products using symmetric block ciphers: ◦ Advanced Encryption Standard (AES) is required for new code ◦ Three-key triple Data Encryption Standard (3DES) is permissible in existing code for backward compatibility ◦ All other block ciphers, including RC2, DES, 2 Key 3DES, DESX, and Skipjack, may only be used for decrypting old data, and must be replaced if used for encryption • For symmetric block encryption algorithms, a minimum key length of 128 bits is required. The only block encryption algorithm recommended for new code is AES (AES-128, AES-192 and AES-256 are all acceptable) • Three-key 3DES is currently acceptable if already in use in existing code; transition to AES is recommended. DES, DESX, RC2, and Skipjack are no longer considered secure. These algorithms may only be used for decrypting existing data for the sake of backward-compatibility, and data should be re-encrypted using a recommended block cipher Please note that all symmetric block ciphers must be used with an approved cipher mode, which requires use of an appropriate initialization vector (IV). An appropriate IV, is typically a random number and never a constant value The use of legacy or otherwise unapproved crypto algorithms and smaller key lengths for reading existing data (as opposed to writing new data) may be permitted after your organization's Crypto Board review. However, you must file for an exception against this requirement. Additionally, in enterprise deployments, products should consider warning administrators when weak crypto is used to read data. Such warnings should be explanatory and actionable. In some cases, it may be appropriate to have Group Policy control the use of weak crypto Allowed .NET algorithms for managed crypto agility (in order of preference) • AesCng (FIPS compliant)

IriusRisk
Ex. 1002 - Page 257

TITLE	DETAILS
	• AuthenticatedAesCng (FIPS compliant) • AESCryptoServiceProvider (FIPS compliant) • AESManaged (non-FIPS-compliant) Please note that none of these algorithms can be specified via the `SymmetricAlgorithm.Create` or `CryptoConfig.CreateFromName` methods without making changes to the machine.config file. Also, note that AES in versions of .NET prior to .NET 3.5 is named `RijndaelManaged`, and `AesCng` and `AuthenticatedAesCng` are >available through CodePlex and require CNG in the underlying OS

Use approved block cipher modes and initialization vectors for symmetric ciphers

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	All symmetric block ciphers must be used with an approved symmetric cipher mode. The only approved modes are CBC and CTS. In particular, the electronic code book (ECB) mode of operation should be avoided; use of ECB requires your organization's Crypto Board review. All usage of OFB, CFB, CTR, CCM, and GCM or any other encryption mode must be reviewed by your organization's Crypto Board. Reusing the same initialization vector (IV) with block ciphers in "streaming ciphers modes," such as CTR, may cause encrypted data to be revealed. All symmetric block ciphers must also be used with an appropriate initialization vector (IV). An appropriate IV is a cryptographically strong, random number and never a constant value.

Use approved asymmetric algorithms, key lengths, and padding

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	The use of banned cryptographic algorithms introduces significant risk to product security and must be avoided. Products must use only those cryptographic algorithms and associated key lengths and padding that have been explicitly approved by your organization's Crypto Board. • **RSA**- may be used for encryption, key exchange and signature. RSA encryption must use only the OAEP or RSA-KEM padding modes. Existing code may use PKCS #1 v1.5 padding mode for compatibility only. Use of null padding is explicitly banned. Keys >= 2048 bits is required for new code. Existing code may support keys < 2048 bits only for backwards compatibility after a review by your organization's Crypto Board. Keys < 1024 bits may only be used for decrypting/verifying old data, and must be replaced if used for encryption or signing operations • **ECDSA**- may be used for signature only. ECDSA with >=256-bit keys is required for new code. ECDSA-based signatures must use one of the three NIST approved curves (P-256, P-384, or P521). Curves that have been thoroughly analyzed may be used only after a review with your organization's Crypto Board. • **ECDH**- may be used for key exchange only. ECDH with >=256-bit keys is required for new code. ECDH-based key exchange must use one of the three NIST approved curves (P-256, P-384, or P521). Curves that have been thoroughly analyzed may be used only after a review with your organization's Crypto Board. • **DSA**- may be acceptable after review and approval from your organization's Crypto Board. Contact your security advisor to schedule your organization's Crypto Board review. If your use of DSA is approved, note that you will need to prohibit use of keys less than 2048 bits in length. CNG supports 2048-bit and greater key lengths as of Windows 8. • **Diffie-Hellman**- may be used for session key management only. Key length >= 2048 bits is required for new code. Existing code may support key lengths < 2048 bits only for backwards compatibility after a review by your organization's Crypto Board. Keys < 1024 bits may not be used.

Use approved random number generators

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	Products must use approved random number generators. Pseudorandom functions such as the C runtime function rand, the .NET Framework class System.Random, or system functions such as GetTickCount must, therefore, never be used in such code. Use of the dual elliptic curve random number generator (DUAL_EC_DRBG) algorithm is prohibited • **CNG**- BCryptGenRandom(use of the BCRYPT_USE_SYSTEM_PREFERRED_RNG flag recommended unless the caller might run at any IRQL greater than 0 [that is, PASSIVE_LEVEL]) • **CAPI**- cryptGenRandom • **Win32/64**- RtlGenRandom (new implementations should use BCryptGenRandom or CryptGenRandom) * rand_s * SystemPrng (for kernel mode) • **.NET**- RNGCryptoServiceProvider or RNGCng • **Windows Store Apps**- Windows.Security.Cryptography.CryptographicBuffer.GenerateRandom or .GenerateRandomNumber • **Apple OS X (10.7+)/iOS(2.0+)**- int SecRandomCopyBytes (SecRandomRef random, size_t count, uint8_t *bytes)* • *Apple OS X (< 10.7)*-* Use /dev/random to retrieve random numbers • **Java(including Google Android Java code)**- java.security.SecureRandom class. Note that for Android 4.3 (Jelly Bean), developers must follow the Android recommended workaround and update their applications to explicitly initialize the PRNG with entropy from /dev/urandom or /dev/random

Do not use symmetric stream ciphers

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Symmetric stream ciphers, such as RC4, must not be used. Instead of symmetric stream ciphers, products should use a block cipher, specifically AES with a key length of at least 128 bits.

Use approved MAC/HMAC/keyed hash algorithms

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Products must use only approved message authentication code (MAC) or hash-based message authentication code (HMAC) algorithms. A message authentication code (MAC) is a piece of information attached to a message that allows its recipient to verify both the authenticity of the sender and the integrity of the message using a secret key. The use of either a hash-based MAC (HMAC) or block-cipher-based MAC is permissible as long as all underlying hash or symmetric encryption algorithms are also approved for use; currently this includes the HMAC-SHA2 functions (HMAC-SHA256, HMAC-SHA384 and HMAC-SHA512) and the CMAC/OMAC1 and OMAC2 block cipher-based MACs (these are based on AES). Use of HMAC-SHA1 may be permissible for platform compatibility, but you will be required to file an exception to this procedure and undergo your organization's Crypto review. Truncation of HMACs to less than 128 bits is not permitted. Using customer methods to hash a key and data is not approved, and must undergo your organization's Crypto Board review prior to use.

Use only approved cryptographic hash functions

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Products must use the SHA-2 family of hash algorithms (SHA256, SHA384, and SHA512). If a shorter hash is needed, such as a 128-bit output length in order to fit a data structure designed with the shorter MD5 hash in mind, product teams may truncate one of the SHA2 hashes (typically SHA256). Note that SHA384 is a truncated version of SHA512. Truncation of cryptographic hashes for security purposes to less than 128 bits is not permitted. New code must not use the MD2, MD4, MD5, SHA-0, SHA-1, or RIPEMD hash algorithms. Hash collisions are computationally feasible for these algorithms, which effectively breaks them. Allowed .NET hash algorithms for managed crypto agility (in order of preference): • SHA512Cng (FIPS compliant) • SHA384Cng (FIPS compliant) • SHA256Cng (FIPS compliant) • SHA512Managed (non-FIPS-compliant) (use SHA512 as algorithm name in calls to HashAlgorithm.Create or CryptoConfig.CreateFromName) • SHA384Managed (non-FIPS-compliant) (use SHA384 as algorithm name in calls to HashAlgorithm.Create or CryptoConfig.CreateFromName) • SHA256Managed (non-FIPS-compliant) (use SHA256 as algorithm name in calls to HashAlgorithm.Create or CryptoConfig.CreateFromName) • SHA512CryptoServiceProvider (FIPS compliant) • SHA256CryptoServiceProvider (FIPS compliant) • SHA384CryptoServiceProvider (FIPS compliant)

Use strong encryption algorithms to encrypt data in the database

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Choosing an encryption algorithm
Steps	Encryption algorithms define data transformations that cannot be easily reversed by unauthorized users. SQL Server allows administrators and developers to choose from among several algorithms, including DES, Triple DES, TRIPLE_DES_3KEY, RC2, RC4, 128-bit RC4, DESX, 128-bit AES, 192-bit AES, and 256-bit AES

SSIS packages should be encrypted and digitally signed

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Identify the Source of Packages with Digital Signatures, Threat and Vulnerability Mitigation (Integration Services)
Steps	The source of a package is the individual or organization that created the package. Running a package from an unknown or untrusted source might be risky. To prevent unauthorized tampering of SSIS packages, digital signatures should be used. Also, to ensure the confidentiality of the packages during storage/transit, SSIS packages have to be encrypted

Add digital signature to critical database securables

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	ADD SIGNATURE (Transact-SQL)
Steps	In cases where the integrity of a critical database securable has to be verified, digital signatures should be used. Database securables such as a stored procedure, function, assembly, or trigger can be digitally signed. Below is an example of when this can be useful: Let us say an ISV (Independent Software Vendor) has provided support to a software delivered to one of their customers. Before providing support, the ISV would want to ensure that a database securable in the software was not tampered either by mistake or by a malicious attempt. If the securable is digitally signed, the ISV can verify its digital signature and validate its integrity.

Use SQL server EKM to protect encryption keys

TITLE	DETAILS
Component	Database
SDL Phase	Build

TITLE	DETAILS
Applicable Technologies	Generic
Attributes	N/A
References	SQL Server Extensible Key Management (EKM), Extensible Key Management Using Azure Key Vault (SQL Server)
Steps	SQL Server Extensible Key Management enables the encryption keys that protect the database files to be stored in an off-box device such as a smartcard, USB device, or EKM/HSM module. This also enables data protection from database administrators (except members of the sysadmin group). Data can be encrypted by using encryption keys that only the database user has access to on the external EKM/HSM module.

Use AlwaysEncrypted feature if encryption keys should not be revealed to Database engine

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure, OnPrem
Attributes	SQL Version - V12, MsSQL2016
References	Always Encrypted (Database Engine)
Steps	Always Encrypted is a feature designed to protect sensitive data, such as credit card numbers or national identification numbers (e.g. U.S. social security numbers), stored in Azure SQL Database or SQL Server databases. Always Encrypted allows clients to encrypt sensitive data inside client applications and never reveal the encryption keys to the Database Engine (SQL Database or SQL Server). As a result, Always Encrypted provides a separation between those who own the data (and can view it) and those who manage the data (but should have no access)

Store Cryptographic Keys securely on IoT Device

TITLE	DETAILS
Component	IoT Device
SDL Phase	Build
Applicable Technologies	Generic

TITLE	DETAILS
Attributes	Device OS - Windows IoT Core, Device Connectivity - Azure IoT device SDKs
References	TPM on Windows IoT Core, Set up TPM on Windows IoT Core, Azure IoT Device SDK TPM
Steps	Symmetric or Certificate Private keys securely in a hardware protected storage like TPM or Smart Card chips. Windows 10 IoT Core supports the user of a TPM and there are several compatible TPMs that can be used: https://developer.microsoft.com/windows/iot/win10/tpm. It is recommended to use a Firmware or Discrete TPM. A Software TPM should only be used for development and testing purposes. Once a TPM is available and the keys are provisioned in it, the code that generates the token should be written without hard coding any sensitive information in it.

Example

```
TpmDevice myDevice = new TpmDevice(0);
// Use logical device 0 on the TPM
string hubUri = myDevice.GetHostName();
string deviceId = myDevice.GetDeviceId();
string sasToken = myDevice.GetSASToken();

var deviceClient = DeviceClient.Create( hubUri, AuthenticationMethodFactory.
CreateAuthenticationWithToken(deviceId, sasToken), TransportType.Amqp);
```

As can be seen, the device primary key is not present in the code. Instead, it is stored in the TPM at slot 0. TPM device generates a short-lived SAS token that is then used to connect to the IoT Hub.

Generate a random symmetric key of sufficient length for authentication to IoT Hub

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	Gateway choice - Azure IoT Hub
References	N/A
Steps	IoT Hub contains a device Identity Registry and while provisioning a device, automatically generates a random Symmetric key. It is recommended to use this feature of the Azure IoT Hub Identity Registry to generate the key used for authentication. IoT Hub also allows for a key to be specified while creating the device. If a key is generated outside of IoT Hub during device provisioning, it is recommended to create a random symmetric key or at least 256 bits.

Ensure a device management policy is in place that requires a use PIN and allows remote wiping

TITLE	DETAILS
Component	Dynamics CRM Mobile Client
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure a device management policy is in place that requires a use PIN and allows remote wiping

Ensure a device management policy is in place that requires a PIN/password/auto lock and encrypts all data (e.g. Bitlocker)

TITLE	DETAILS
Component	Dynamics CRM Outlook Client
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure a device management policy is in place that requires a PIN/password/auto lock and encrypts all data (e.g. Bitlocker)

Ensure that signing keys are rolled over when using Identity Server

TITLE	DETAILS
Component	Identity Server
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Identity Server - Keys, Signatures and Cryptography

TITLE	DETAILS
Steps	Ensure that signing keys are rolled over when using Identity Server. The link in the references section explains how this should be planned without causing outages to applications relying on Identity Server.

Ensure that cryptographically strong client ID, client secret are used in Identity Server

TITLE	DETAILS
Component	Identity Server
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that cryptographically strong client ID, client secret are used in Identity Server. The following guidelines should be used while generating a client ID and secret: • Generate a random GUID as the client ID • Generate a cryptographically random 256-bit key as the secret

Security Frame: Exception Management | Mitigations

1/24/2018 • 7 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
WCF	• WCF- Do not include serviceDebug node in configuration file • WCF- Do not include serviceMetadata node in configuration file
Web API	• Ensure that proper exception handling is done in ASP.NET Web API
Web Application	• Do not expose security details in error messages • Implement Default error handling page • Set Deployment Method to Retail in IIS • Exceptions should fail safely

WCF- Do not include serviceDebug node in configuration file

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, NET Framework 3
Attributes	N/A
References	MSDN, Fortify Kingdom
Steps	Windows Communication Framework (WCF) services can be configured to expose debugging information. Debug information should not be used in production environments. The `<serviceDebug>` tag defines whether the debug information feature is enabled for a WCF service. If the attribute includeExceptionDetailInFaults is set to true, exception information from the application will be returned to clients. Attackers can leverage the additional information they gain from debugging output to mount attacks targeted on the framework, database, or other resources used by the application.

Example

The following configuration file includes the `<serviceDebug>` tag:

```
<configuration>
<system.serviceModel>
<behaviors>
<serviceBehaviors>
<behavior name=""MyServiceBehavior"">
<serviceDebug includeExceptionDetailInFaults=""True"" httpHelpPageEnabled=""True""/>
...
```

Disable debugging information in the service. This can be accomplished by removing the `<serviceDebug>` tag from your application's configuration file.

WCF- Do not include serviceMetadata node in configuration file

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic
Attributes	Generic, NET Framework 3
References	MSDN, Fortify Kingdom
Steps	Publicly exposing information about a service can provide attackers with valuable insight into how they might exploit the service. The `<serviceMetadata>` tag enables the metadata publishing feature. Service metadata could contain sensitive information that should not be publicly accessible. At a minimum, only allow trusted users to access the metadata and ensure that unnecessary information is not exposed. Better yet, entirely disable the ability to publish metadata. A safe WCF configuration will not contain the `<serviceMetadata>` tag.

Ensure that proper exception handling is done in ASP.NET Web API

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC 5, MVC 6
Attributes	N/A
References	Exception Handling in ASP.NET Web API, Model Validation in ASP.NET Web API
Steps	By default, most uncaught exceptions in ASP.NET Web API are translated into an HTTP response with status code `500, Internal Server Error`

Example

To control the status code returned by the API, `HttpResponseException` can be used as shown below:

```
public Product GetProduct(int id)
{
    Product item = repository.Get(id);
    if (item == null)
    {
        throw new HttpResponseException(HttpStatusCode.NotFound);
    }
    return item;
}
```

Example

For further control on the exception response, the `HttpResponseMessage` class can be used as shown below:

```
public Product GetProduct(int id)
{
    Product item = repository.Get(id);
    if (item == null)
    {
        var resp = new HttpResponseMessage(HttpStatusCode.NotFound)
        {
            Content = new StringContent(string.Format("No product with ID = {0}", id)),
            ReasonPhrase = "Product ID Not Found"
        }
        throw new HttpResponseException(resp);
    }
    return item;
}
```

To catch unhandled exceptions that are not of the type `HttpResponseException`, Exception Filters can be used. Exception filters implement the `System.Web.Http.Filters.IExceptionFilter` interface. The simplest way to write an exception filter is to derive from the `System.Web.Http.Filters.ExceptionFilterAttribute` class and override the OnException method.

Example

Here is a filter that converts `NotImplementedException` exceptions into HTTP status code `501, Not Implemented`:

```
namespace ProductStore.Filters
{
    using System;
    using System.Net;
    using System.Net.Http;
    using System.Web.Http.Filters;

    public class NotImplExceptionFilterAttribute : ExceptionFilterAttribute
    {
        public override void OnException(HttpActionExecutedContext context)
        {
            if (context.Exception is NotImplementedException)
            {
                context.Response = new HttpResponseMessage(HttpStatusCode.NotImplemented);
            }
        }
    }
}
```

There are several ways to register a Web API exception filter:

- By action
- By controller
- Globally

Example

To apply the filter to a specific action, add the filter as an attribute to the action:

```
public class ProductsController : ApiController
{
    [NotImplExceptionFilter]
    public Contact GetContact(int id)
    {
        throw new NotImplementedException("This method is not implemented");
    }
}
```

Example

To apply the filter to all of the actions on a `controller`, add the filter as an attribute to the `controller` class:

```
[NotImplExceptionFilter]
public class ProductsController : ApiController
{
    // ...
}
```

Example

To apply the filter globally to all Web API controllers, add an instance of the filter to the `GlobalConfiguration.Configuration.Filters` collection. Exception filters in this collection apply to any Web API controller action.

```
GlobalConfiguration.Configuration.Filters.Add(
    new ProductStore.NotImplExceptionFilterAttribute());
```

Example

For model validation, the model state can be passed to CreateErrorResponse method as shown below:

```
public HttpResponseMessage PostProduct(Product item)
{
    if (!ModelState.IsValid)
    {
        return Request.CreateErrorResponse(HttpStatusCode.BadRequest, ModelState);
    }
    // Implementation not shown...
}
```

Check the links in the references section for additional details about exceptional handling and model validation in ASP.Net Web API

Do not expose security details in error messages

TITLE	DETAILS
Component	Web Application

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Generic error messages are provided directly to the user without including sensitive application data. Examples of sensitive data include: • Server names • Connection strings • Usernames • Passwords • SQL procedures • Details of dynamic SQL failures • Stack trace and lines of code • Variables stored in memory • Drive and folder locations • Application install points • Host configuration settings • Other internal application details Trapping all errors within an application and providing generic error messages, as well as enabling custom errors within IIS will help prevent information disclosure. SQL Server database and .NET Exception handling, among other error handling architectures, are especially verbose and extremely useful to a malicious user profiling your application. Do not directly display the contents of a class derived from the .NET Exception class, and ensure that you have proper exception handling so that an unexpected exception isn't inadvertently raised directly to the user. • Provide generic error messages directly to the user that abstract away specific details found directly in the exception/error message • Do not display the contents of a .NET exception class directly to the user • Trap all error messages and if appropriate inform the user via a generic error message sent to the application client • Do not expose the contents of the Exception class directly to the user, especially the return value from `.ToString()`, or the values of the Message or StackTrace properties. Securely log this information and display a more innocuous message to the user

Implement Default error handling page

TITLE	DETAILS
Component	Web Application

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Edit ASP.NET Error Pages Settings Dialog Box
Steps	When an ASP.NET application fails and causes an HTTP/1.x 500 Internal Server Error, or a feature configuration (such as Request Filtering) prevents a page from being displayed, an error message will be generated. Administrators can choose whether or not the application should display a friendly message to the client, detailed error message to the client, or detailed error message to localhost only. The tag in the web.config has three modes: • **On:** Specifies that custom errors are enabled. If no defaultRedirect attribute is specified, users see a generic error. The custom errors are shown to the remote clients and to the local host • **Off:** Specifies that custom errors are disabled. The detailed ASP.NET errors are shown to the remote clients and to the local host • **RemoteOnly:** Specifies that custom errors are shown only to the remote clients, and that ASP.NET errors are shown to the local host. This is the default value Open the `web.config` file for the application/site and ensure that the tag has either `<customErrors mode="RemoteOnly" />` or `<customErrors mode="On" />` defined.

Set Deployment Method to Retail in IIS

TITLE	DETAILS
Component	Web Application
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	deployment Element (ASP.NET Settings Schema)

TITLE	DETAILS
Steps	The `<deployment retail>` switch is intended for use by production IIS servers. This switch is used to help applications run with the best possible performance and least possible security information leakages by disabling the application's ability to generate trace output on a page, disabling the ability to display detailed error messages to end users, and disabling the debug switch. Often times, switches and options that are developer-focused, such as failed request tracing and debugging, are enabled during active development. It is recommended that the deployment method on any production server be set to retail. open the machine.config file and ensure that `<deployment retail="true" />` remains set to true.

Exceptions should fail safely

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Fail securely
Steps	Application should fail safely. Any method that returns a Boolean value, based on which certain decision is made, should have exception block carefully created. There are lot of logical errors due to which security issues creep in, when the exception block is written carelessly.

Example

```
public static bool ValidateDomain(string pathToValidate, Uri currentUrl)
{
    try
    {
        if (!string.IsNullOrWhiteSpace(pathToValidate))
        {
            var domain = RetrieveDomain(currentUrl);
            var replyPath = new Uri(pathToValidate);
            var replyDomain = RetrieveDomain(replyPath);

            if (string.Compare(domain, replyDomain, StringComparison.OrdinalIgnoreCase) != 0)
            {
                //// Adding additional check to enable CMS urls if they are not hosted on same domain.
                if (!string.IsNullOrWhiteSpace(Utilities.CmsBase))
                {
                    var cmsDomain = RetrieveDomain(new Uri(Utilities.Base.Trim()));
                    if (string.Compare(cmDomain, replyDomain, StringComparison.OrdinalIgnoreCase) != 0)
                    {
                        return false;
                    }
                    else
                    {
                        return true;
                    }
                }

                return false;
            }
        }

        return true;
    }
    catch (UriFormatException ex)
    {
        LogHelper.LogException("Utilities:ValidateDomain", ex);
        return true;
    }
}
```

The above method will always return True, if some exception happens. If the end user provides a malformed URL, that the browser respects, but the `Uri()` constructor doesn't, this will throw an exception, and the victim will be taken to the valid but malformed URL.

Security Frame: Input Validation | Mitigations

1/24/2018 • 31 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Web Application	• Disable XSLT scripting for all transforms using untrusted style sheets • Ensure that each page that could contain user controllable content opts out of automatic MIME sniffing • Harden or Disable XML Entity Resolution • Applications utilizing http.sys perform URL canonicalization verification • Ensure appropriate controls are in place when accepting files from users • Ensure that type-safe parameters are used in Web Application for data access • Use separate model binding classes or binding filter lists to prevent MVC mass assignment vulnerability • Encode untrusted web output prior to rendering • Perform input validation and filtering on all string type Model properties • Sanitization should be applied on form fields that accept all characters, e.g. rich text editor • Do not assign DOM elements to sinks that do not have inbuilt encoding • Validate all redirects within the application are closed or done safely • Implement input validation on all string type parameters accepted by Controller methods • Set upper limit timeout for regular expression processing to prevent DoS due to bad regular expressions • Avoid using Html.Raw in Razor views
Database	• Do not use dynamic queries in stored procedures
Web API	• Ensure that model validation is done on Web API methods • Implement input validation on all string type parameters accepted by Web API methods • Ensure that type-safe parameters are used in Web API for data access
Azure Document DB	• Use parametrized SQL queries for Azure Cosmos DB
WCF	• WCF Input validation through Schema binding • WCF - Input validation through Parameter Inspectors

Disable XSLT scripting for all transforms using untrusted style sheets

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	XSLT Security, XsltSettings.EnableScript Property
Steps	XSLT supports scripting inside style sheets using the `<msxml:script>` element. This allows custom functions to be used in an XSLT transformation. The script is executed under the context of the process performing the transform. XSLT script must be disabled when in an untrusted environment to prevent execution of untrusted code. *If using .NET*: XSLT scripting is disabled by default; however, you must ensure that it has not been explicitly enabled through the `XsltSettings.EnableScript` property.

Example

```
XsltSettings settings = new XsltSettings();
settings.EnableScript = true; // WRONG: THIS SHOULD BE SET TO false
```

Example

If you are using using MSXML 6.0, XSLT scripting is disabled by default; however, you must ensure that it has not been explicitly enabled through the XML DOM object property AllowXsltScript.

```
doc.setProperty("AllowXsltScript", true); // WRONG: THIS SHOULD BE SET TO false
```

Example

If you are using MSXML 5 or below, XSLT scripting is enabled by default and you must explicitly disable it. Set the XML DOM object property AllowXsltScript to false.

```
doc.setProperty("AllowXsltScript", false); // CORRECT. Setting to false disables XSLT scripting.
```

Ensure that each page that could contain user controllable content opts out of automatic MIME sniffing

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	IE8 Security Part V - Comprehensive Protection
Steps	For each page that could contain user controllable content, you must use the HTTP Header `X-Content-Type-Options:nosniff`. To comply with this requirement, you can either set the required header page by page for only those pages that might contain user-controllable content, or you can set it globally for all pages in the application. Each type of file delivered from a web server has an associated MIME type (also called a *content-type*) that describes the nature of the content (that is, image, text, application, etc.) The X-Content-Type-Options header is an HTTP header that allows developers to specify that their content should not be MIME-sniffed. This header is designed to mitigate MIME-Sniffing attacks. Support for this header was added in Internet Explorer 8 (IE8) Only users of Internet Explorer 8 (IE8) will benefit from X-Content-Type-Options. Previous versions of Internet Explorer do not currently respect the X-Content-Type-Options header Internet Explorer 8 (and later) are the only major browsers to implement a MIME-sniffing opt-out feature. If and when other major browsers (Firefox, Safari, Chrome) implement similar features, this recommendation will be updated to include syntax for those browsers as well

Example

To enable the required header globally for all pages in the application, you can do one of the following:

- Add the header in the web.config file if the application is hosted by Internet Information Services (IIS) 7

```
<system.webServer>
  <httpProtocol>
    <customHeaders>
      <add name=""X-Content-Type-Options"" value=""nosniff""/>
    </customHeaders>
  </httpProtocol>
</system.webServer>
```

- Add the header through the global Application_BeginRequest

```
void Application_BeginRequest(object sender, EventArgs e)
{
  this.Response.Headers[""X-Content-Type-Options""] = ""nosniff"";
}
```

- Implement custom HTTP module

```
public class XContentTypeOptionsModule : IHttpModule
  {
    #region IHttpModule Members
    public void Dispose()
    {

    }
    public void Init(HttpApplication context)
    {
      context.PreSendRequestHeaders += newEventHandler(context_PreSendRequestHeaders);
    }
    #endregion
    void context_PreSendRequestHeaders(object sender, EventArgs e)
      {
        HttpApplication application = sender as HttpApplication;
        if (application == null)
          return;
        if (application.Response.Headers[""X-Content-Type-Options ""] != null)
          return;
        application.Response.Headers.Add(""X-Content-Type-Options "", ""nosniff"");
      }
  }
```

- You can enable the required header only for specific pages by adding it to individual responses:

```
this.Response.Headers[""X-Content-Type-Options""] = ""nosniff"";
```

Harden or Disable XML Entity Resolution

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	XML Entity Expansion, XML Denial of Service Attacks and Defenses, MSXML Security Overview, Best Practices for Securing MSXML Code, NSXMLParserDelegate Protocol Reference, Resolving External References

TITLE	DETAILS
Steps	Although it is not widely used, there is a feature of XML that allows the XML parser to expand macro entities with values defined either within the document itself or from external sources. For example, the document might define an entity "companyname" with the value "Microsoft," so that every time the text "&companyname;" appears in the document, it is automatically replaced with the text Microsoft. Or, the document might define an entity "MSFTStock" that references an external web service to fetch the current value of Microsoft stock. Then any time "&MSFTStock;" appears in the document, it is automatically replaced with the current stock price. However, this functionality can be abused to create denial of service (DoS) conditions. An attacker can nest multiple entities to create an exponential expansion XML bomb that consumes all available memory on the system. Alternatively, he can create an external reference that streams back an infinite amount of data or that simply hangs the thread. As a result, all teams must disable internal and/or external XML entity resolution entirely if their application does not use it, or manually limit the amount of memory and time that the application can consume for entity resolution if this functionality is absolutely necessary. If entity resolution is not required by your application, then disable it.

Example

For .NET Framework code, you can use the following approaches:

```
XmlTextReader reader = new XmlTextReader(stream);
reader.ProhibitDtd = true;

XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = true;
XmlReader reader = XmlReader.Create(stream, settings);

// for .NET 4
XmlReaderSettings settings = new XmlReaderSettings();
settings.DtdProcessing = DtdProcessing.Prohibit;
XmlReader reader = XmlReader.Create(stream, settings);
```

Note that the default value of `ProhibitDtd` in `XmlReaderSettings` is true, but in `XmlTextReader` it is false. If you are using XmlReaderSettings, you do not need to set ProhibitDtd to true explicitly, but it is recommended for safety sake that you do. Also note that the XmlDocument class allows entity resolution by default.

Example

To disable entity resolution for XmlDocuments, use the `XmlDocument.Load(XmlReader)` overload of the Load method and set the appropriate properties in the XmlReader argument to disable resolution, as illustrated in the following code:

```
XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = true;
XmlReader reader = XmlReader.Create(stream, settings);
XmlDocument doc = new XmlDocument();
doc.Load(reader);
```

Example

If disabling entity resolution is not possible for your application, set the XmlReaderSettings.MaxCharactersFromEntities property to a reasonable value according to your application's needs. This will limit the impact of potential exponential expansion DoS attacks. The following code provides an example of this approach:

```
XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = false;
settings.MaxCharactersFromEntities = 1000;
XmlReader reader = XmlReader.Create(stream, settings);
```

Example

If you need to resolve inline entities but do not need to resolve external entities, set the XmlReaderSettings.XmlResolver property to null. For example:

```
XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = false;
settings.MaxCharactersFromEntities = 1000;
settings.XmlResolver = null;
XmlReader reader = XmlReader.Create(stream, settings);
```

Note that in MSXML6, ProhibitDTD is set to true (disabling DTD processing) by default. For Apple OSX/iOS code, there are two XML parsers you can use: NSXMLParser and libXML2.

Applications utilizing http.sys perform URL canonicalization verification

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Any application that uses http.sys should follow these guidelines: • Limit the URL length to no more than 16,384 characters (ASCII or Unicode). This is the absolute maximum URL length based on the default Internet Information Services (IIS) 6 setting. Websites should strive for a length shorter than this if possible • Use the standard .NET Framework file I/O classes (such as FileStream) as these will take advantage of the canonicalization rules in the .NET FX • Explicitly build an allow-list of known filenames • Explicitly reject known filetypes you will not serve UrlScan rejects: exe, bat, cmd, com, htw, ida, idq, htr, idc, shtm[l], stm, printer, ini, pol, dat files • Catch the following exceptions: ◦ System.ArgumentException (for device names) ◦ System.NotSupportedException (for data streams) ◦ System.IO.FileNotFoundException (for invalid escaped filenames) ◦ System.IO.DirectoryNotFoundException (for invalid escaped dirs) • *Do not* call out to Win32 file I/O APIs. On an invalid URL gracefully return a 400 error to the user, and log the real error.

Ensure appropriate controls are in place when accepting files from users

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Unrestricted File Upload, File Signature Table

TITLE	DETAILS
Steps	Uploaded files represent a significant risk to applications. The first step in many attacks is to get some code to the system to be attacked. Then the attack only needs to find a way to get the code executed. Using a file upload helps the attacker accomplish the first step. The consequences of unrestricted file upload can vary, including complete system takeover, an overloaded file system or database, forwarding attacks to back-end systems, and simple defacement. It depends on what the application does with the uploaded file and especially where it is stored. Server side validation of file uploads is missing. Following security controls should be implemented for File Upload functionality: • File Extension check (only a valid set of allowed file type should be accepted) • Maximum file size limit • File should not be uploaded to webroot; the location should be a directory on non-system drive • Naming convention should be followed, such that the uploaded file name have some randomness, so as to prevent file overwrites • Files should be scanned for anti-virus before writing to the disk • Ensure that the file name and any other metadata (e.g., file path) are validated for malicious characters • File format signature should be checked, to prevent a user from uploading a masqueraded file (e.g., uploading an exe file by changing extension to txt)

Example

For the last point regarding file format signature validation, refer to the class below for details:

```
        private static Dictionary<string, List<byte[]>> fileSignature = new Dictionary<string, List<byte[]>>
                    {
                    { ".DOC", new List<byte[]> { new byte[] { 0xD0, 0xCF, 0x11, 0xE0, 0xA1, 0xB1, 0x1A, 0xE1 }
} },
                    { ".DOCX", new List<byte[]> { new byte[] { 0x50, 0x4B, 0x03, 0x04 } } },
                    { ".PDF", new List<byte[]> { new byte[] { 0x25, 0x50, 0x44, 0x46 } } },
                    { ".ZIP", new List<byte[]>
                                        {
                                          new byte[] { 0x50, 0x4B, 0x03, 0x04 },
                                          new byte[] { 0x50, 0x4B, 0x4C, 0x49, 0x54, 0x55 },
                                          new byte[] { 0x50, 0x4B, 0x53, 0x70, 0x58 },
                                          new byte[] { 0x50, 0x4B, 0x05, 0x06 },
                                          new byte[] { 0x50, 0x4B, 0x07, 0x08 },
                                          new byte[] { 0x57, 0x69, 0x6E, 0x5A, 0x69, 0x70 }
                                            }
                                        },
                    { ".PNG", new List<byte[]> { new byte[] { 0x89, 0x50, 0x4E, 0x47, 0x0D, 0x0A, 0x1A, 0x0A }
} },
                    { ".JPG", new List<byte[]>
                                        {
                                          new byte[] { 0xFF, 0xD8, 0xFF, 0xE0 },
                                          new byte[] { 0xFF, 0xD8, 0xFF, 0xE1 },
                                          new byte[] { 0xFF, 0xD8, 0xFF, 0xE8 }
                                        }
                                        },
                    { ".JPEG", new List<byte[]>
```

```
                {
                    new byte[] { 0xFF, 0xD8, 0xFF, 0xE0 },
                    new byte[] { 0xFF, 0xD8, 0xFF, 0xE2 },
                    new byte[] { 0xFF, 0xD8, 0xFF, 0xE3 }
                }
                },
            { ".XLS", new List<byte[]>
                {
                  new byte[] { 0xD0, 0xCF, 0x11, 0xE0, 0xA1, 0xB1, 0x1A, 0xE1 },
                  new byte[] { 0x09, 0x08, 0x10, 0x00, 0x00, 0x06, 0x05, 0x00 },
                  new byte[] { 0xFD, 0xFF, 0xFF, 0xFF }
                }
                },
            { ".XLSX", new List<byte[]> { new byte[] { 0x50, 0x4B, 0x03, 0x04 } } },
            { ".GIF", new List<byte[]> { new byte[] { 0x47, 0x49, 0x46, 0x38 } } }
        };

public static bool IsValidFileExtension(string fileName, byte[] fileData, byte[] allowedChars)
{
    if (string.IsNullOrEmpty(fileName) || fileData == null || fileData.Length == 0)
    {
        return false;
    }

    bool flag = false;
    string ext = Path.GetExtension(fileName);
    if (string.IsNullOrEmpty(ext))
    {
        return false;
    }

    ext = ext.ToUpperInvariant();

    if (ext.Equals(".TXT") || ext.Equals(".CSV") || ext.Equals(".PRN"))
    {
        foreach (byte b in fileData)
        {
            if (b > 0x7F)
            {
                if (allowedChars != null)
                {
                    if (!allowedChars.Contains(b))
                    {
                        return false;
                    }
                }
                else
                {
                    return false;
                }
            }
        }

        return true;
    }

    if (!fileSignature.ContainsKey(ext))
    {
        return true;
    }

    List<byte[]> sig = fileSignature[ext];
    foreach (byte[] b in sig)
    {
        var curFileSig = new byte[b.Length];
        Array.Copy(fileData, curFileSig, b.Length);
        if (curFileSig.SequenceEqual(b))
        {
```

```
                flag = true;
                break;
            }
        }

        return flag;
    }
```

Ensure that type-safe parameters are used in Web Application for data access

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	If you use the Parameters collection, SQL treats the input is as a literal value rather then as executable code. The Parameters collection can be used to enforce type and length constraints on input data. Values outside of the range trigger an exception. If type-safe SQL parameters are not used, attackers might be able to execute injection attacks that are embedded in the unfiltered input. Use type safe parameters when constructing SQL queries to avoid possible SQL injection attacks that can occur with unfiltered input. You can use type safe parameters with stored procedures and with dynamic SQL statements. Parameters are treated as literal values by the database and not as executable code. Parameters are also checked for type and length.

Example

The following code shows how to use type safe parameters with the SqlParameterCollection when calling a stored procedure.

```
using System.Data;
using System.Data.SqlClient;

using (SqlConnection connection = new SqlConnection(connectionString))
{
DataSet userDataset = new DataSet();
SqlDataAdapter myCommand = new SqlDataAdapter(LoginStoredProcedure", connection);
myCommand.SelectCommand.CommandType = CommandType.StoredProcedure;
myCommand.SelectCommand.Parameters.Add("@au_id", SqlDbType.VarChar, 11);
myCommand.SelectCommand.Parameters["@au_id"].Value = SSN.Text;
myCommand.Fill(userDataset);
}
```

In the preceding code example, the input value cannot be longer than 11 characters. If the data does not conform to

the type or length defined by the parameter, the SqlParameter class throws an exception.

Use separate model binding classes or binding filter lists to prevent MVC mass assignment vulnerability

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	N/A
References	Metadata Attributes, Public Key Security Vulnerability And Mitigation, Complete Guide to Mass Assignment in ASP.NET MVC, Getting Started with EF using MVC
Steps	• **When should I look for over-posting vulnerabilities?** - Over-posting vulnerabilities can occur any place you bind model classes from user input. Frameworks like MVC can represent user data in custom .NET classes, including Plain Old CLR Objects (POCOs). MVC automatically populates these model classes with data from the request, providing a convenient representation for dealing with user input. When these classes include properties that should not be set by the user, the application can be vulnerable to over-posting attacks, which allow user control of data that the application never intended. Like MVC model binding, database access technologies such as object/relational mappers like Entity Framework often also support using POCO objects to represent database data. These data model classes provide the same convenience in dealing with database data as MVC does in dealing with user input. Because both MVC and the database support similar models, like POCO objects, it seems easy to reuse the same classes for both purposes. This practice fails to preserve separation of concerns, and it is one common area where unintended properties are exposed to model binding, enabling over-posting attacks. • **Why shouldn't I use my unfiltered database model classes as parameters to my MVC actions?** - Because MVC model binding will bind anything in that class. Even if the data does not appear in your view, a malicious user can send an HTTP request with this data included, and MVC will gladly bind it because your action says that database class is the shape of data it should accept for user input. • **Why should I care about the shape used for model binding?** - Using ASP.NET MVC model binding with overly broad models exposes an application to over-posting attacks. Over-posting could enable attackers to change application data beyond what the developer intended, such as overriding the price for an item or the security privileges for an account. Applications should use action-specific binding models (or specific allowed property filter lists) to prevent [illegible]

TITLE	DETAILS
	contract for what untrusted input to allow via model binding. • **Is having separate binding models just duplicating code?** - No, it is a matter of separation of concerns. If you reuse database models in action methods, you are saying any property (or sub-property) in that class can be set by the user in an HTTP request. If that is not what you want MVC to do, you need a filter list or a separate class shape to show MVC what data can come from user input instead. • **If I have separate binding models for user input, do I have to duplicate all my data annotation attributes?** - Not necessarily. You can use MetadataTypeAttribute on the database model class to link to the metadata on a model binding class. Just note that the type referenced by the MetadataTypeAttribute must be a subset of the referencing type (it can have fewer properties, but not more). • **Moving data back and forth between user input models and database models is tedious. Can I just copy over all properties using reflection?** - Yes. The only properties that appear in the binding models are the ones you have determined to be safe for user input. There is no security reason that prevents using reflection to copy over all properties that exist in common between these two models. • **What about [Bind(Exclude ="â€¦")]. Can I use that instead of having separate binding models?** - This approach is not recommended. Using [Bind(Exclude ="â€¦")] means that any new property is bindable by default. When a new property is added, there is an extra step to remember to keep things secure, rather than having the design be secure by default. Depending on the developer checking this list every time a property is added is risky. • **Is [Bind(Include ="â€¦")] useful for Edit operations?** - No. [Bind(Include ="â€¦")] is only suitable for INSERT-style operations (adding new data). For UPDATE-style operations (revising existing data), use another approach, like having separate binding models or passing an explicit list of allowed properties to UpdateModel or TryUpdateModel. Adding a [Bind(Include ="â€¦")] attribute on an Edit operation means that MVC will create an object instance and set only the listed properties, leaving all others at their default values. When the data is persisted, it will entirely replace the existing entity, resetting the values for any omitted properties to their defaults. For example, if IsAdmin was omitted from a [Bind(Include ="â€¦")] attribute on an Edit operation, any user whose name was edited via this action would be reset to IsAdmin = false (any edited user would lose administrator status). If you want to prevent updates to certain properties, use one of the other approaches above. Note that some versions of MVC tooling generate controller classes with [Bind(Include ="â€¦")] on Edit actions and imply that removing a property from that list will prevent over-posting attacks. However, as described above, that approach does not work as intended and instead will reset any data in the omitted properties to their default values. • **For Create operations, are there any caveats using [Bind(Include ="â€¦")] rather than separate**

TrustRisk
Ex. 1002 - Page 287

TITLE	DETAILS
	binding models? - Yes. First this approach does not work for Edit scenarios, requiring maintaining two separate approaches for mitigating all over-posting vulnerabilities. Second, separate binding models enforce separation of concerns between the shape used for user input and the shape used for persistence, something [Bind(Include ="â€¦")] does not do. Third, note that [Bind(Include ="â€¦")] can only handle top-level properties; you cannot allow only portions of sub-properties (such as "Details.Name") in the attribute. Finally, and perhaps most importantly, using [Bind(Include ="â€¦")] adds an extra step that must be remembered any time the class is used for model binding. If a new action method binds to the data class directly and forgets to include a [Bind(Include ="â€¦")] attribute, it can be vulnerable to over-posting attacks, so the [Bind(Include ="â€¦")] approach is somewhat less secure by default. If you use [Bind(Include ="â€¦")], take care always to remember to specify it every time your data classes appear as action method parameters. • **For Create operations, what about putting the [Bind(Include ="â€¦")] attribute on the model class itself? Does not this approach avoid the need to remember putting the attribute on every action method?** - This approach works in some cases. Using [Bind(Include ="â€¦")] on the model type itself (rather than on action parameters using this class), does avoid the need to remember to include the [Bind(Include ="â€¦")] attribute on every action method. Using the attribute directly on the class effectively creates a separate surface area of this class for model binding purposes. However, this approach only allows for one model binding shape per model class. If one action method needs to allow model binding of a field (for example, an administrator-only action that updates user roles) and other actions need to prevent model binding of this field, this approach will not work. Each class can only have one model binding shape; if different actions need different model binding shapes, they need to represent these separate shapes using either separate model binding classes or separate [Bind(Include ="â€¦")] attributes on the action methods. • **What are binding models? Are they the same thing as view models?** - These are two related concepts. The term binding model refers to a model class used in an action is parameter list (the shape passed from MVC model binding to the action method). The term view model refers to a model class passed from an action method to a view. Using a view-specific model is a common approach for passing data from an action method to a view. Often, this shape is also suitable for model binding, and the term view model can be used to refer the same model used in both places. To be precise, this procedure talks specifically about binding models, focusing on the shape passed to the action, which is what matters for mass assignment purposes.

Encode untrusted web output prior to rendering

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic, Web Forms, MVC5, MVC6
Attributes	N/A
References	How to prevent Cross-site scripting in ASP.NET, Cross-site Scripting, XSS (Cross Site Scripting) Prevention Cheat Sheet
Steps	Cross-site scripting (commonly abbreviated as XSS) is an attack vector for online services or any application/component that consumes input from the web. XSS vulnerabilities may allow an attacker to execute script on another user's machine through a vulnerable web application. Malicious scripts can be used to steal cookies and otherwise tamper with a victim's machine through JavaScript. XSS is prevented by validating user input, ensuring it is well formed and encoding before it is rendered in a web page. Input validation and output encoding can be done by using Web Protection Library. For Managed code (C#, VB.net, etc.), use one or more appropriate encoding methods from the Web Protection (Anti-XSS) Library, depending on the context where the user input gets manifested:

Example

```
* Encoder.HtmlEncode
* Encoder.HtmlAttributeEncode
* Encoder.JavaScriptEncode
* Encoder.UrlEncode
* Encoder.VisualBasicScriptEncode
* Encoder.XmlEncode
* Encoder.XmlAttributeEncode
* Encoder.CssEncode
* Encoder.LdapEncode
```

Perform input validation and filtering on all string type Model properties

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic, MVC5, MVC6
Attributes	N/A

TITLE	DETAILS
References	Adding Validation, Validating Model Data in an MVC Application, Guiding Principles For Your ASP.NET MVC Applications
Steps	All the input parameters must be validated before they are used in the application to ensure that the application is safeguarded against malicious user inputs. Validate the input values using regular expression validations on server side with a whitelist validation strategy. Unsanitized user inputs / parameters passed to the methods can cause code injection vulnerabilities. For web applications, entry points can also include form fields, QueryStrings, cookies, HTTP headers, and web service parameters. The following input validation checks must be performed upon model binding: • The model properties should be annotated with RegularExpression annotation, for accepting allowed characters and maximum permissible length • The controller methods should perform ModelState validity

Sanitization should be applied on form fields that accept all characters, e.g, rich text editor

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Encode Unsafe Input, HTML Sanitizer

TITLE	DETAILS
Steps	Identify all static markup tags that you want to use. A common practice is to restrict formatting to safe HTML elements, such as `<b>` (bold) and `<i>` (italic). Before writing the data, HTML-encode it. This makes any malicious script safe by causing it to be handled as text, not as executable code. 1. Disable ASP.NET request validation by the adding the ValidateRequest="false" attribute to the @ Page directive 2. Encode the string input with the HtmlEncode method 3. Use a StringBuilder and call its Replace method to selectively remove the encoding on the HTML elements that you want to permit The page-in the references disables ASP.NET request validation by setting `ValidateRequest="false"`. It HTML-encodes the input and selectively allows the `<b>` and `<i>` Alternatively, a .NET library for HTML sanitization may also be used. HtmlSanitizer is a .NET library for cleaning HTML fragments and documents from constructs that can lead to XSS attacks. It uses AngleSharp to parse, manipulate, and render HTML and CSS. HtmlSanitizer can be installed as a NuGet package, and the user input can be passed through relevant HTML or CSS sanitization methods, as applicable, on the server side. Please note that Sanitization as a security control should be considered only as a last option. Input validation and Output Encoding are considered better security controls.

Do not assign DOM elements to sinks that do not have inbuilt encoding

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Many javascript functions don't do encoding by default. When assigning untrusted input to DOM elements via such functions, may result in cross site script (XSS) executions.

Example

Following are insecure examples:

```
document.getElementByID("div1").innerHtml = value;
$("#userName").html(res.Name);
return $('<div/>').html(value)
$('body').append(resHTML);
```

Don't use `innerHtml`; instead use `innerText`. Similarly, instead of `$("#elm").html()`, use `$("#elm").text()`

Validate all redirects within the application are closed or done safely

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	The OAuth 2.0 Authorization Framework - Open Redirectors
Steps	Application design requiring redirection to a user-supplied location must constrain the possible redirection targets to a predefined "safe" list of sites or domains. All redirects in the application must be closed/safe. To do this: • Identify all redirects • Implement an appropriate mitigation for each redirect. Appropriate mitigations include redirect whitelist or user confirmation. If a web site or service with an open redirect vulnerability uses Facebook/OAuth/OpenID identity providers, an attacker can steal a user's logon token and impersonate that user. This is an inherent risk when using OAuth, which is documented in RFC 6749 "The OAuth 2.0 Authorization Framework", Section 10.15 "Open Redirects" Similarly, users' credentials can be compromised by spear phishing attacks using open redirects

Implement input validation on all string type parameters accepted by Controller methods

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic, MVC5, MVC6
Attributes	N/A

TITLE	DETAILS
References	Validating Model Data in an MVC Application, Guiding Principles For Your ASP.NET MVC Applications
Steps	For methods that just accept primitive data type, and not models as argument,input validation using Regular Expression should be done. Here Regex.IsMatch should be used with a valid regex pattern. If the input doesn't match the specified Regular Expression, control should not proceed further, and an adequate warning regarding validation failure should be displayed.

Set upper limit timeout for regular expression processing to prevent DoS due to bad regular expressions

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic, Web Forms, MVC5, MVC6
Attributes	N/A
References	DefaultRegexMatchTimeout Property
Steps	To ensure denial of service attacks against badly created regular expressions, that cause a lot of backtracking, set the global default timeout. If the processing time takes longer than the defined upper limit, it would throw a Timeout exception. If nothing is configured, the timeout would be infinite.

Example

For example, the following configuration will throw a RegexMatchTimeoutException, if the processing takes more than 5 seconds:

```
<httpRuntime targetFramework="4.5" defaultRegexMatchTimeout="00:00:05" />
```

Avoid using Html.Raw in Razor views

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	N/A

TITLE	DETAILS
References	N/A
Step	ASP.Net WebPages (Razor) perform automatic HTML encoding. All strings printed by embedded code nuggets (@ blocks) are automatically HTML-encoded. However, when `HtmlHelper.Raw` Method is invoked, it returns markup that is not HTML encoded. If `Html.Raw()` helper method is used, it bypasses the automatic encoding protection that Razor provides.

Example

Following is an insecure example:

```
<div class="form-group">
            @Html.Raw(Model.AccountConfirmText)
        </div>
        <div class="form-group">
            @Html.Raw(Model.PaymentConfirmText)
        </div>
</div>
```

Do not use `Html.Raw()` unless you need to display markup. This method does not perform output encoding implicitly. Use other ASP.NET helpers e.g., `@Html.DisplayFor()`

Do not use dynamic queries in stored procedures

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	A SQL injection attack exploits vulnerabilities in input validation to run arbitrary commands in the database. It can occur when your application uses input to construct dynamic SQL statements to access the database. It can also occur if your code uses stored procedures that are passed strings that contain raw user input. Using the SQL injection attack, the attacker can execute arbitrary commands in the database. All SQL statements (including the SQL statements in stored procedures) must be parameterized. Parameterized SQL statements will accept characters that have special meaning to SQL (like single quote) without problems because they are strongly typed.

Example

Following is an example of insecure dynamic Stored Procedure:

```
CREATE PROCEDURE [dbo].[uspGetProductsByCriteria]
(
  @productName nvarchar(200) = NULL,
  @startPrice float = NULL,
  @endPrice float = NULL
)
AS
 BEGIN
  DECLARE @sql nvarchar(max)
  SELECT @sql = ' SELECT ProductID, ProductName, Description, UnitPrice, ImagePath' +
       ' FROM dbo.Products WHERE 1 = 1 '
       PRINT @sql
  IF @productName IS NOT NULL
     SELECT @sql = @sql + ' AND ProductName LIKE ''%' + @productName + '%'''
  IF @startPrice IS NOT NULL
     SELECT @sql = @sql + ' AND UnitPrice > ''' + CONVERT(VARCHAR(10),@startPrice) + ''''
  IF @endPrice IS NOT NULL
     SELECT @sql = @sql + ' AND UnitPrice < ''' + CONVERT(VARCHAR(10),@endPrice) + ''''

  PRINT @sql
  EXEC(@sql)
 END
```

Example

Following is the same stored procedure implemented securely:

```
CREATE PROCEDURE [dbo].[uspGetProductsByCriteriaSecure]
(
            @productName nvarchar(200) = NULL,
            @startPrice float = NULL,
            @endPrice float = NULL
)
AS
       BEGIN
            SELECT ProductID, ProductName, Description, UnitPrice, ImagePath
            FROM dbo.Products where
            (@productName IS NULL or ProductName like '%'+ @productName +'%')
            AND
            (@startPrice IS NULL or UnitPrice > @startPrice)
            AND
            (@endPrice IS NULL or UnitPrice < @endPrice)
       END
```

Ensure that model validation is done on Web API methods

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	N/A
References	Model Validation in ASP.NET Web API

TITLE	DETAILS
Steps	When a client sends data to a web API, it is mandatory to validate the data before doing any processing. For ASP.NET Web APIs which accept models as input, use data annotations on models to set validation rules on the properties of the model.

Example

The following code demonstrates the same:

```
using System.ComponentModel.DataAnnotations;

namespace MyApi.Models
{
    public class Product
    {
        public int Id { get; set; }
        [Required]
        [RegularExpression(@"^[a-zA-Z0-9]*$", ErrorMessage="Only alphanumeric characters are allowed.")]
        public string Name { get; set; }
        public decimal Price { get; set; }
        [Range(0, 999)]
        public double Weight { get; set; }
    }
}
```

Example

In the action method of the API controllers, validity of the model has to be explicitly checked as shown below:

```
namespace MyApi.Controllers
{
    public class ProductsController : ApiController
    {
        public HttpResponseMessage Post(Product product)
        {
            if (ModelState.IsValid)
            {
                // Do something with the product (not shown).

                return new HttpResponseMessage(HttpStatusCode.OK);
            }
            else
            {
                return Request.CreateErrorResponse(HttpStatusCode.BadRequest, ModelState);
            }
        }
    }
}
```

Implement input validation on all string type parameters accepted by Web API methods

TITLE	DETAILS
Component	Web API

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic, MVC 5, MVC 6
Attributes	N/A
References	Validating Model Data in an MVC Application, Guiding Principles For Your ASP.NET MVC Applications
Steps	For methods that just accept primitive data type, and not models as argument,input validation using Regular Expression should be done. Here Regex.IsMatch should be used with a valid regex pattern. If the input doesn't match the specified Regular Expression, control should not proceed further, and an adequate warning regarding validation failure should be displayed.

Ensure that type-safe parameters are used in Web API for data access

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	If you use the Parameters collection, SQL treats the input is as a literal value rather then as executable code. The Parameters collection can be used to enforce type and length constraints on input data. Values outside of the range trigger an exception. If type-safe SQL parameters are not used, attackers might be able to execute injection attacks that are embedded in the unfiltered input. Use type safe parameters when constructing SQL queries to avoid possible SQL injection attacks that can occur with unfiltered input. You can use type safe parameters with stored procedures and with dynamic SQL statements. Parameters are treated as literal values by the database and not as executable code. Parameters are also checked for type and length.

Example

The following code shows how to use type safe parameters with the SqlParameterCollection when calling a stored procedure.

```
using System.Data;
using System.Data.SqlClient;

using (SqlConnection connection = new SqlConnection(connectionString))
{
DataSet userDataset = new DataSet();
SqlDataAdapter myCommand = new SqlDataAdapter(LoginStoredProcedure", connection);
myCommand.SelectCommand.CommandType = CommandType.StoredProcedure;
myCommand.SelectCommand.Parameters.Add("@au_id", SqlDbType.VarChar, 11);
myCommand.SelectCommand.Parameters["@au_id"].Value = SSN.Text;
myCommand.Fill(userDataset);
}
```

In the preceding code example, the input value cannot be longer than 11 characters. If the data does not conform to the type or length defined by the parameter, the SqlParameter class throws an exception.

Use parametrized SQL queries for Cosmos DB

TITLE	DETAILS
Component	Azure Document DB
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Announcing SQL Parameterization in Azure Cosmos DB
Steps	Although Azure Cosmos DB only supports read-only queries, SQL injection is still possible if queries are constructed by concatenating with user input. It might be possible for a user to gain access to data they shouldn't be accessing within the same collection by crafting malicious SQL queries. Use parameterized SQL queries if queries are constructed based on user input.

WCF Input validation through Schema binding

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, NET Framework 3
Attributes	N/A
References	MSDN

TITLE	DETAILS
Steps	Lack of validation leads to different type injection attacks. Message validation represents one line of defense in the protection of your WCF application. With this approach, you validate messages using schemas to protect WCF service operations from attack by a malicious client. Validate all messages received by the client to protect the client from attack by a malicious service. Message validation makes it possible to validate messages when operations consume message contracts or data contracts, which cannot be done using parameter validation. Message validation allows you to create validation logic inside schemas, thereby providing more flexibility and reducing development time. Schemas can be reused across different applications inside the organization, creating standards for data representation. Additionally, message validation allows you to protect operations when they consume more complex data types involving contracts representing business logic. To perform message validation, you first build a schema that represents the operations of your service and the data types consumed by those operations. You then create a .NET class that implements a custom client message inspector and custom dispatcher message inspector to validate the messages sent/received to/from the service. Next, you implement a custom endpoint behavior to enable message validation on both the client and the service. Finally, you implement a custom configuration element on the class that allows you to expose the extended custom endpoint behavior in the configuration file of the service or the client"

WCF- Input validation through Parameter Inspectors

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, NET Framework 3
Attributes	N/A
References	MSDN

TITLE	DETAILS
Steps	Input and data validation represents one important line of defense in the protection of your WCF application. You should validate all parameters exposed in WCF service operations to protect the service from attack by a malicious client. Conversely, you should also validate all return values received by the client to protect the client from attack by a malicious service WCF provides different extensibility points that allow you to customize the WCF runtime behavior by creating custom extensions. Message Inspectors and Parameter Inspectors are two extensibility mechanisms used to gain greater control over the data passing between a client and a service. You should use parameter inspectors for input validation and use message inspectors only when you need to inspect the entire message flowing in and out of a service. To perform input validation, you will build a .NET class and implement a custom parameter inspector in order to validate parameters on operations in your service. You will then implement a custom endpoint behavior to enable validation on both the client and the service. Finally, you will implement a custom configuration element on the class that allows you to expose the extended custom endpoint behavior in the configuration file of the service or the client

Security Frame: Sensitive Data | Mitigations

1/24/2018 • 16 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Machine Trust Boundary	• Ensure that binaries are obfuscated if they contain sensitive information • Consider using Encrypted File System (EFS) is used to protect confidential user-specific data • Ensure that sensitive data stored by the application on the file system is encrypted
Web Application	• Ensure that sensitive content is not cached on the browser • Encrypt sections of Web App's configuration files that contain sensitive data • Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs • Ensure that sensitive data displayed on the user screen is masked
Database	• Implement dynamic data masking to limit sensitive data exposure non privileged users • Ensure that passwords are stored in salted hash format • Ensure that sensitive data in database columns is encrypted • Ensure that database-level encryption (TDE) is enabled • Ensure that database backups are encrypted
Web API	• Ensure that sensitive data relevant to Web API is not stored in browser's storage
Azure Document DB	• Encrypt sensitive data stored in Azure Cosmos DB
Azure IaaS VM Trust Boundary	• Use Azure Disk Encryption to encrypt disks used by Virtual Machines
Service Fabric Trust Boundary	• Encrypt secrets in Service Fabric applications
Dynamics CRM	• Perform security modeling and use Business Units/Teams where required • Minimize access to share feature on critical entities • Train users on the risks associated with the Dynamics CRM Share feature and good security practices • Include a development standards rule proscribing showing config details in exception management

PRODUCT/SERVICE	ARTICLE
Azure Storage	• Use Azure Storage Service Encryption (SSE) for Data at Rest (Preview) • Use Client-Side Encryption to store sensitive data in Azure Storage
Mobile Client	• Encrypt sensitive or PII data written to phones local storage • Obfuscate generated binaries before distributing to end users
WCF	• Set clientCredentialType to Certificate or Windows • WCF-Security Mode is not enabled

Ensure that binaries are obfuscated if they contain sensitive information

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that binaries are obfuscated if they contain sensitive information such as trade secrets, sensitive business logic that should not reversed. This is to stop reverse engineering of assemblies. Tools like `CryptoObfuscator` may be used for this purpose.

Consider using Encrypted File System (EFS) is used to protect confidential user-specific data

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Consider using Encrypted File System (EFS) is used to protect confidential user-specific data from adversaries with physical access to the computer.

Ensure that sensitive data stored by the application on the file system is encrypted

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that sensitive data stored by the application on the file system is encrypted (e.g., using DPAPI), if EFS cannot be enforced

Ensure that sensitive content is not cached on the browser

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic, Web Forms, MVC5, MVC6
Attributes	N/A
References	N/A
Steps	Browsers can store information for purposes of caching and history. These cached files are stored in a folder, like the Temporary Internet Files folder in the case of Internet Explorer. When these pages are referred again, the browser displays them from its cache. If sensitive information is displayed to the user (such as their address, credit card details, Social Security Number, or username), then this information could be stored in browser's cache, and therefore retrievable through examining the browser's cache or by simply pressing the browser's "Back" button. Set cache-control response header value to "no-store" for all pages.

Example

```
<configuration>
  <system.webServer>
   <httpProtocol>
    <customHeaders>
        <add name="Cache-Control" value="no-cache" />
        <add name="Pragma" value="no-cache" />
        <add name="Expires" value="-1" />
    </customHeaders>
  </httpProtocol>
 </system.webServer>
</configuration>
```

Example

This may be implemented through a filter. Following example may be used:

```
public override void OnActionExecuting(ActionExecutingContext filterContext)
        {
            if (filterContext == null || (filterContext.HttpContext != null &&
filterContext.HttpContext.Response != null && filterContext.HttpContext.Response.IsRequestBeingRedirected))
            {
                //// Since this is MVC pipeline, this should never be null.
                return;
            }

            var attributes =
filterContext.ActionDescriptor.GetCustomAttributes(typeof(System.Web.Mvc.OutputCacheAttribute), false);
            if (attributes == null || **Attributes**.Count() == 0)
            {
                filterContext.HttpContext.Response.Cache.SetNoStore();
                filterContext.HttpContext.Response.Cache.SetCacheability(HttpCacheability.NoCache);
                filterContext.HttpContext.Response.Cache.SetExpires(DateTime.UtcNow.AddHours(-1));
                if (!filterContext.IsChildAction)
                {
                    filterContext.HttpContext.Response.AppendHeader("Pragma", "no-cache");
                }
            }

            base.OnActionExecuting(filterContext);
        }
```

Encrypt sections of Web App's configuration files that contain sensitive data

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	How To: Encrypt Configuration Sections in ASP.NET 2.0 Using DPAPI, Specifying a Protected Configuration Provider, Using Azure Key Vault to protect application secrets

TITLE	DETAILS
Steps	Configuration files such as the Web.config, appsettings.json are often used to hold sensitive information, including user names, passwords, database connection strings, and encryption keys. If you do not protect this information, your application is vulnerable to attackers or malicious users obtaining sensitive information such as account user names and passwords, database names and server names. Based on the deployment type (azure/on-prem), encrypt the sensitive sections of config files using DPAPI or services like Azure Key Vault.

Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	MSDN: autocomplete attribute, Using AutoComplete in HTML, HTML Sanitization Vulnerability, Autocomplete..again?!
Steps	The autocomplete attribute specifies whether a form should have autocomplete on or off. When autocomplete is on, the browser automatically complete values based on values that the user has entered before. For example, when a new name and password is entered in a form and the form is submitted, the browser asks if the password should be saved.Thereafter when the form is displayed, the name and password are filled in automatically or are completed as the name is entered. An attacker with local access could obtain the clear text password from the browser cache. By default autocomplete is enabled, and it must explicitly be disabled.

Example

```
<form action="Login.aspx" method="post " autocomplete="off" >
      Social Security Number: <input type="text" name="ssn" />
      <input type="submit" value="Submit" />
</form>
```

Ensure that sensitive data displayed on the user screen is masked

TITLE	DETAILS
Component	Web Application

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Sensitive data such as passwords, credit card numbers, SSN etc. should be masked when displayed on the screen. This is to prevent unauthorized personnel from accessing the data (e.g., shoulder-surfing passwords, support personnel viewing SSN numbers of users) . Ensure that these data elements are not visible in plain text and are appropriately masked. This has to be taken care while accepting them as input (e.g,. input type="password") as well as displaying back on the screen (e.g., display only the last 4 digits of the credit card number).

Implement dynamic data masking to limit sensitive data exposure non privileged users

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Sql Azure, OnPrem
Attributes	SQL Version - V12, SQL Version - MsSQL2016
References	Dynamic Data Masking
Steps	The purpose of dynamic data masking is to limit exposure of sensitive data, preventing users who should not have access to the data from viewing it. Dynamic data masking does not aim to prevent database users from connecting directly to the database and running exhaustive queries that expose pieces of the sensitive data. Dynamic data masking is complementary to other SQL Server security features (auditing, encryption, row level security...) and it is highly recommended to use this feature in conjunction with them in addition in order to better protect the sensitive data in the database. Please note that this feature is supported only by SQL Server starting with 2016 and Azure SQL Database.

Ensure that passwords are stored in salted hash format

TITLE	DETAILS
Component	Database

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Password Hashing using .NET Crypto APIs
Steps	Passwords should not be stored in custom user store databases. Password hashes should be stored with salt values instead. Make sure the salt for the user is always unique and you apply b-crypt, s-crypt or PBKDF2 before storing the password, with a minimum work factor iteration count of 150,000 loops to eliminate the possibility of brute forcing.

Ensure that sensitive data in database columns is encrypted

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	SQL Version - All
References	Encrypting sensitive data in SQL server, How to: Encrypt a Column of Data in SQL Server, Encrypt by Certificate
Steps	Sensitive data such as credit card numbers has to be encrypted in the database. Data can be encrypted using column-level encryption or by an application function using the encryption functions.

Ensure that database-level encryption (TDE) is enabled

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Understanding SQL Server Transparent Data Encryption (TDE)

TITLE	DETAILS
Steps	Transparent Data Encryption (TDE) feature in SQL server helps in encrypting sensitive data in a database and protect the keys that are used to encrypt the data with a certificate. This prevents anyone without the keys from using the data. TDE protects data "at rest", meaning the data and log files. It provides the ability to comply with many laws, regulations, and guidelines established in various industries.

Ensure that database backups are encrypted

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure, OnPrem
Attributes	SQL Version - V12, SQL Version - MsSQL2014
References	SQL database backup encryption
Steps	SQL Server has the ability to encrypt the data while creating a backup. By specifying the encryption algorithm and the encryptor (a Certificate or Asymmetric Key) when creating a backup, one can create an encrypted backup file.

Ensure that sensitive data relevant to Web API is not stored in browser's storage

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC 5, MVC 6
Attributes	Identity Provider - ADFS, Identity Provider - Azure AD
References	N/A

TITLE	DETAILS
Steps	In certain implementations, sensitive artifacts relevant to Web API's authentication are stored in browser's local storage. E.g., Azure AD authentication artifacts like adal.idtoken, adal.nonce.idtoken, adal.access.token.key, adal.token.keys, adal.state.login, adal.session.state, adal.expiration.key etc. All these artifacts are available even after sign out or browser is closed. If an adversary gets access to these artifacts, he/she can reuse them to access the protected resources (APIs). Ensure that all sensitive artifacts related to Web API is not stored in browser's storage. In cases where client-side storage is unavoidable (e.g., Single Page Applications (SPA) that leverage Implicit OpenIdConnect/OAuth flows need to store access tokens locally), use storage choices with do not have persistence. e.g., prefer SessionStorage to LocalStorage.

Example

The below JavaScript snippet is from a custom authentication library which stores authentication artifacts in local storage. Such implementations should be avoided.

```
ns.AuthHelper.Authenticate = function () {
window.config = {
instance: 'https://login.microsoftonline.com/',
tenant: ns.Configurations.Tenant,
clientId: ns.Configurations.AADApplicationClientID,
postLogoutRedirectUri: window.location.origin,
cacheLocation: 'localStorage', // enable this for IE, as sessionStorage does not work for localhost.
};
```

Encrypt sensitive data stored in Cosmos DB

TITLE	DETAILS
Component	Azure Document DB
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Encrypt sensitive data at application level before storing in document DB or store any sensitive data in other storage solutions like Azure Storage or Azure SQL

Use Azure Disk Encryption to encrypt disks used by Virtual Machines

TITLE	DETAILS
Component	Azure IaaS VM Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Using Azure Disk Encryption to encrypt disks used by your virtual machines
Steps	Azure Disk Encryption is a new feature that is currently in preview. This feature allows you to encrypt the OS disks and Data disks used by an IaaS Virtual Machine. For Windows, the drives are encrypted using industry-standard BitLocker encryption technology. For Linux, the disks are encrypted using the DM-Crypt technology. This is integrated with Azure Key Vault to allow you to control and manage the disk encryption keys. The Azure Disk Encryption solution supports the following three customer encryption scenarios: • Enable encryption on new IaaS VMs created from customer-encrypted VHD files and customer-provided encryption keys, which are stored in Azure Key Vault. • Enable encryption on new IaaS VMs created from the Azure Marketplace. • Enable encryption on existing IaaS VMs already running in Azure.

Encrypt secrets in Service Fabric applications

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Build
Applicable Technologies	Generic
Attributes	Environment - Azure
References	Managing secrets in Service Fabric applications
Steps	Secrets can be any sensitive information, such as storage connection strings, passwords, or other values that should not be handled in plain text. Use Azure Key Vault to manage keys and secrets in service fabric applications.

Perform security modeling and use Business Units/Teams where required

TITLE	DETAILS
Component	Dynamics CRM
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Perform security modeling and use Business Units/Teams where required

Minimize access to share feature on critical entities

TITLE	DETAILS
Component	Dynamics CRM
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Minimize access to share feature on critical entities

Train users on the risks associated with the Dynamics CRM Share feature and good security practices

TITLE	DETAILS
Component	Dynamics CRM
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Train users on the risks associated with the Dynamics CRM Share feature and good security practices

Include a development standards rule proscribing showing config

details in exception management

TITLE	DETAILS
Component	Dynamics CRM
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Include a development standards rule proscribing showing config details in exception management outside development. Test for this as part of code reviews or periodic inspection.

Use Azure Storage Service Encryption (SSE) for Data at Rest (Preview)

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	StorageType - Blob
References	Azure Storage Service Encryption for Data at Rest (Preview)

TITLE	DETAILS
Steps	Azure Storage Service Encryption (SSE) for Data at Rest helps you protect and safeguard your data to meet your organizational security and compliance commitments. With this feature, Azure Storage automatically encrypts your data prior to persisting to storage and decrypts prior to retrieval. The encryption, decryption and key management is totally transparent to users. SSE applies only to block blobs, page blobs, and append blobs. The other types of data, including tables, queues, and files, will not be encrypted. Encryption and Decryption Workflow: • The customer enables encryption on the storage account • When the customer writes new data (PUT Blob, PUT Block, PUT Page, etc.) to Blob storage; every write is encrypted using 256-bit AES encryption, one of the strongest block ciphers available • When the customer needs to access data (GET Blob, etc.), data is automatically decrypted before returning to the user • If encryption is disabled, new writes are no longer encrypted and existing encrypted data remains encrypted until rewritten by the user. While encryption is enabled, writes to Blob storage will be encrypted. The state of data does not change with the user toggling between enabling/disabling encryption for the storage account • All encryption keys are stored, encrypted, and managed by Microsoft Please note that at this time, the keys used for the encryption are managed by Microsoft. Microsoft generates the keys originally, and manage the secure storage of the keys as well as the regular rotation as defined by internal Microsoft policy. In the future, customers will get the ability to manage their own >encryption keys, and provide a migration path from Microsoft-managed keys to customer-managed keys.

Use Client-Side Encryption to store sensitive data in Azure Storage

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Client-Side Encryption and Azure Key Vault for Microsoft Azure Storage, Tutorial: Encrypt and decrypt blobs in Microsoft Azure Storage using Azure Key Vault, Storing Data Securely in Azure Blob Storage with Azure Encryption Extension

IriusRisk
Ex. 1002 - Page 313

TITLE	DETAILS
Steps	The Azure Storage Client Library for .NET Nuget package supports encrypting data within client applications before uploading to Azure Storage, and decrypting data while downloading to the client. The library also supports integration with Azure Key Vault for storage account key management. Here is a brief description of how client side encryption works: • The Azure Storage client SDK generates a content encryption key (CEK), which is a one-time-use symmetric key • Customer data is encrypted using this CEK • The CEK is then wrapped (encrypted) using the key encryption key (KEK). The KEK is identified by a key identifier and can be an asymmetric key pair or a symmetric key and can be managed locally or stored in Azure Key Vault. The Storage client itself never has access to the KEK. It just invokes the key wrapping algorithm that is provided by Key Vault. Customers can choose to use custom providers for key wrapping/unwrapping if they want • The encrypted data is then uploaded to the Azure Storage service. Check the links in the references section for low-level implementation details.

Encrypt sensitive or PII data written to phones local storage

TITLE	DETAILS
Component	Mobile Client
SDL Phase	Build
Applicable Technologies	Generic, Xamarin
Attributes	N/A
References	Manage settings and features on your devices with Microsoft Intune policies, Keychain Valet
Steps	If the application writes sensitive information like user's PII (email, phone number, first name, last name, preferences etc.)- on mobile's file system, then it should be encrypted before writing to the local file system. If the application is an enterprise application, then explore the possibility of publishing application using Windows Intune.

Example

Intune can be configured with following security policies to safeguard sensitive data:

```
Require encryption on mobile device
Require encryption on storage cards
Allow screen capture
```

Example

If the application is not an enterprise application, then use platform provided keystore, keychains to store encryption keys, using which cryptographic operation may be performed on the file system. Following code snippet shows how to access key from keychain using xamarin:

```
protected static string EncryptionKey
{
    get
    {
        if (String.IsNullOrEmpty(_Key))
        {
            var query = new SecRecord(SecKind.GenericPassword);
            query.Service = NSBundle.MainBundle.BundleIdentifier;
            query.Account = "UniqueID";

            NSData uniqueId = SecKeyChain.QueryAsData(query);
            if (uniqueId == null)
            {
                query.ValueData = NSData.FromString(System.Guid.NewGuid().ToString());
                var err = SecKeyChain.Add(query);
                _Key = query.ValueData.ToString();
            }
            else
            {
                _Key = uniqueId.ToString();
            }
        }

        return _Key;
    }
}
```

Obfuscate generated binaries before distributing to end users

TITLE	DETAILS
Component	Mobile Client
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Crypto Obfuscation For .Net
Steps	Generated binaries (assemblies within apk) should be obfuscated to stop reverse engineering of assemblies.Tools like `CryptoObfuscator` may be used for this purpose.

Set clientCredentialType to Certificate or Windows

TITLE	DETAILS
Component	WCF

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	.NET Framework 3
Attributes	N/A
References	Fortify
Steps	Using a UsernameToken with a plaintext password over an unencrypted channel exposes the password to attackers who can sniff the SOAP messages. Service Providers that use the UsernameToken might accept passwords sent in plaintext. Sending plaintext passwords over an unencrypted channel can expose the credential to attackers who can sniff the SOAP message.

Example

The following WCF service provider configuration uses the UsernameToken:

```
<security mode="Message">
<message clientCredentialType="UserName" />
```

Set clientCredentialType to Certificate or Windows.

WCF-Security Mode is not enabled

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, .NET Framework 3
Attributes	Security Mode - Transport, Security Mode - Message
References	MSDN, Fortify Kingdom, Fundamentals of WCF Security CoDe Magazine
Steps	No transport or message security has been defined. Applications that transmit messages without transport or message security cannot guarantee the integrity or confidentiality of the messages. When a WCF security binding is set to None, both transport and message security are disabled.

Example

The following configuration sets the security mode to None.

```
<system.serviceModel>
  <bindings>
    <wsHttpBinding>
      <binding name=""MyBinding"">
        <security mode=""None""/>
      </binding>
  </bindings>
</system.serviceModel>
```

Example

Security Mode Across all service bindings there are five possible security modes:

- None. Turns security off.
- Transport. Uses transport security for mutual authentication and message protection.
- Message. Uses message security for mutual authentication and message protection.
- Both. Allows you to supply settings for transport and message-level security (only MSMQ supports this).
- TransportWithMessageCredential. Credentials are passed with the message and message protection and server authentication are provided by the transport layer.
- TransportCredentialOnly. Client credentials are passed with the transport layer and no message protection is applied. Use transport and message security to protect the integrity and confidentiality of messages. The configuration below tells the service to use transport security with message credentials.

```
<system.serviceModel> <bindings> <wsHttpBinding> <binding name=""MyBinding""> <security
mode=""TransportWithMessageCredential""/> <message clientCredentialType=""Windows""/> </binding> </bindings>
</system.serviceModel>
```

Security Frame: Session Management | Articles

1/24/2018 • 14 min to read • Edit Online

PRODUCT/SERVICE	ARTICLE
Azure AD	• Implement proper logout using ADAL methods when using Azure AD
IoT Device	• Use finite lifetimes for generated SaS tokens
Azure Document DB	• Use minimum token lifetimes for generated Resource tokens
ADFS	• Implement proper logout using WsFederation methods when using ADFS
Identity Server	• Implement proper logout when using Identity Server
Web Application	• Applications available over HTTPS must use secure cookies • All http based application should specify http only for cookie definition • Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET web pages • Set up session for inactivity lifetime • Implement proper logout from the application
Web API	• Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET Web APIs

Implement proper logout using ADAL methods when using Azure AD

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	If the application relies on access token issued by Azure AD, the logout event handler should call

Example

```
HttpContext.GetOwinContext().Authentication.SignOut(OpenIdConnectAuthenticationDefaults.AuthenticationType,
CookieAuthenticationDefaults.AuthenticationType)
```

Example

It should also destroy user's session by calling Session.Abandon() method. Following method shows secure implementation of user logout:

```
    [HttpPost]
        [ValidateAntiForgeryToken]
        public void LogOff()
        {
            string userObjectID =
ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/objectidentifier").Value;
            AuthenticationContext authContext = new AuthenticationContext(Authority + TenantId, new
NaiveSessionCache(userObjectID));
            authContext.TokenCache.Clear();
            Session.Clear();
            Session.Abandon();
            Response.SetCookie(new HttpCookie("ASP.NET_SessionId", string.Empty));
            HttpContext.GetOwinContext().Authentication.SignOut(
                OpenIdConnectAuthenticationDefaults.AuthenticationType,
                CookieAuthenticationDefaults.AuthenticationType);
        }
```

Use finite lifetimes for generated SaS tokens

TITLE	DETAILS
Component	IoT Device
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	SaS tokens generated for authenticating to Azure IoT Hub should have a finite expiry period. Keep the SaS token lifetimes to a minimum to limit the amount of time they can be replayed in case the tokens are compromised.

Use minimum token lifetimes for generated Resource tokens

TITLE	DETAILS
Component	Azure Document DB
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Reduce the timespan of resource token to a minimum value required. Resource tokens have a default valid timespan of 1 hour.

Implement proper logout using WsFederation methods when using ADFS

TITLE	DETAILS
Component	ADFS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	If the application relies on STS token issued by ADFS, the logout event handler should call WSFederationAuthenticationModule.FederatedSignOut() method to log out the user. Also the current session should be destroyed, and the session token value should be reset and nullified.

Example

```
        [HttpPost, ValidateAntiForgeryToken]
        [Authorization]
        public ActionResult SignOut(string redirectUrl)
        {
            if (!this.User.Identity.IsAuthenticated)
            {
                return this.View("LogOff", null);
            }

            // Removes the user profile.
            this.Session.Clear();
            this.Session.Abandon();
            HttpContext.Current.Response.Cookies.Add(new System.Web.HttpCookie("ASP.NET_SessionId",
string.Empty)
                {
                    Expires = DateTime.Now.AddDays(-1D),
                    Secure = true,
                    HttpOnly = true
                });

            // Signs out at the specified security token service (STS) by using the WS-Federation protocol.
            Uri signOutUrl = new Uri(FederatedAuthentication.WSFederationAuthenticationModule.Issuer);
            Uri replyUrl = new Uri(FederatedAuthentication.WSFederationAuthenticationModule.Realm);
            if (!string.IsNullOrEmpty(redirectUrl))
            {
                replyUrl = new Uri(FederatedAuthentication.WSFederationAuthenticationModule.Realm +
redirectUrl);
            }
           //     Signs out of the current session and raises the appropriate events.
            var authModule = FederatedAuthentication.WSFederationAuthenticationModule;
            authModule.SignOut(false);
        //     Signs out at the specified security token service (STS) by using the WS-Federation
        //     protocol.
            WSFederationAuthenticationModule.FederatedSignOut(signOutUrl, replyUrl);
            return new RedirectResult(redirectUrl);
        }
```

Implement proper logout when using Identity Server

TITLE	DETAILS
Component	Identity Server
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	IdentityServer3 - Federated sign out
Steps	IdentityServer supports the ability to federate with external identity providers. When a user signs out of an upstream identity provider, depending upon the protocol used, it might be possible to receive a notification when the user signs out. It allows IdentityServer to notify its clients so they can also sign the user out. Check the documentation in the references section for the implementation details.

Applications available over HTTPS must use secure cookies

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	EnvironmentType - OnPrem
References	httpCookies Element (ASP.NET Settings Schema), HttpCookie.Secure Property
Steps	Cookies are normally only accessible to the domain for which they were scoped. Unfortunately, the definition of "domain" does not include the protocol so cookies that are created over HTTPS are accessible over HTTP. The "secure" attribute indicates to the browser that the cookie should only be made available over HTTPS. Ensure that all cookies set over HTTPS use the **secure** attribute. The requirement can be enforced in the web.config file by setting the requireSSL attribute to true. It is the preferred approach because it will enforce the **secure** attribute for all current and future cookies without the need to make any additional code changes.

Example

```
<configuration>
  <system.web>
    <httpCookies requireSSL="true"/>
  </system.web>
</configuration>
```

The setting is enforced even if HTTP is used to access the application. If HTTP is used to access the application, the setting breaks the application because the cookies are set with the secure attribute and the browser will not send them back to the application.

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Web Forms, MVC5
Attributes	EnvironmentType - OnPrem
References	N/A
Steps	When the web application is the Relying Party, and the IdP is ADFS server, the FedAuth token's secure attribute can be configured by setting requireSSL to True in `system.identityModel.services` section of web.config

Example

```
<system.identityModel.services>
  <federationConfiguration>
    <!-- Set requireSsl=true; domain=application domain name used by FedAuth cookies (Ex: .gdinfra.com); -->
    <cookieHandler requireSsl="true" persistentSessionLifetime="0.0:20:0" />
  ....
  </federationConfiguration>
</system.identityModel.services>
```

All http based application should specify http only for cookie definition

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Secure Cookie Attribute
Steps	To mitigate the risk of information disclosure with a cross-site scripting (XSS) attack, a new attribute - httpOnly - was introduced to cookies and is supported by all major browsers. The attribute specifies that a cookie is not accessible through script. By using HttpOnly cookies, a web application reduces the possibility that sensitive information contained in the cookie can be stolen via script and sent to an attacker's website.

Example

All HTTP-based applications that use cookies should specify HttpOnly in the cookie definition, by implementing following configuration in web.config:

```
<system.web>
.
.
   <httpCookies requireSSL="false" httpOnlyCookies="true"/>
.
.
</system.web>
```

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Web Forms
Attributes	N/A

TITLE	DETAILS
References	FormsAuthentication.RequireSSL Property
Steps	The RequireSSL property value is set in the configuration file for an ASP.NET application by using the requireSSL attribute of the configuration element. You can specify in the Web.config file for your ASP.NET application whether SSL (Secure Sockets Layer) is required to return the forms-authentication cookie to the server by setting the requireSSL attribute.

Example

The following code example sets the requireSSL attribute in the Web.config file.

```
<authentication mode="Forms">
  <forms loginUrl="member_login.aspx" cookieless="UseCookies" requireSSL="true"/>
</authentication>
```

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	MVC5
Attributes	EnvironmentType - OnPrem
References	Windows Identity Foundation (WIF) Configuration – Part II
Steps	To set httpOnly attribute for FedAuth cookies, hideFromCsript attribute value should be set to True.

Example

Following configuration shows the correct configuration:

```
<federatedAuthentication>
<cookieHandler mode="Custom"
                       hideFromScript="true"
                       name="FedAuth"
                       path="/"
                       requireSsl="true"
                       persistentSessionLifetime="25">
</cookieHandler>
</federatedAuthentication>
```

Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET web pages

TITLE	DETAILS
Component	Web Application

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker can carry out actions in the security context of a different user's established session on a web site. The goal is to modify or delete content, if the targeted web site relies exclusively on session cookies to authenticate received request. An attacker could exploit this vulnerability by getting a different user's browser to load a URL with a command from a vulnerable site on which the user is already logged in. There are many ways for an attacker to do that, such as by hosting a different web site that loads a resource from the vulnerable server, or getting the user to click a link. The attack can be prevented if the server sends an additional token to the client, requires the client to include that token in all future requests, and verifies that all future requests include a token that pertains to the current session, such as by using the ASP.NET AntiForgeryToken or ViewState.

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	N/A
References	XSRF/CSRF Prevention in ASP.NET MVC and Web Pages
Steps	Anti-CSRF and ASP.NET MVC forms - Use the `AntiForgeryToken` helper method on Views; put an `Html.AntiForgeryToken()` into the form, for example,

Example

```
@using (Html.BeginForm("UserProfile", "SubmitUpdate")) {
    @Html.ValidationSummary(true)
    @Html.AntiForgeryToken()
    <fieldset>
```

Example

```
<form action="/UserProfile/SubmitUpdate" method="post">
    <input name="__RequestVerificationToken" type="hidden"
value="saTFWpkKN0BYazFtN6c4YbZAmsEwG0srqlUqqloi/fVgeV2ciIFVmelvzwRZpArs" />
    <!-- rest of form goes here -->
</form>
```

Example

At the same time, Html.AntiForgeryToken() gives the visitor a cookie called __RequestVerificationToken, with the same value as the random hidden value shown above. Next, to validate an incoming form post, add the [ValidateAntiForgeryToken] filter to the target action method. For example:

```
[ValidateAntiForgeryToken]
public ViewResult SubmitUpdate()
{
// ... etc.
}
```

Authorization filter that checks that:

- The incoming request has a cookie called __RequestVerificationToken
- The incoming request has a `Request.Form` entry called __RequestVerificationToken
- These cookie and `Request.Form` values match Assuming all is well, the request goes through as normal. But if not, then an authorization failure with message "A required anti-forgery token was not supplied or was invalid".

Example

Anti-CSRF and AJAX: The form token can be a problem for AJAX requests, because an AJAX request might send JSON data, not HTML form data. One solution is to send the tokens in a custom HTTP header. The following code uses Razor syntax to generate the tokens, and then adds the tokens to an AJAX request.

```
<script>
    @functions{
        public string TokenHeaderValue()
        {
            string cookieToken, formToken;
            AntiForgery.GetTokens(null, out cookieToken, out formToken);
            return cookieToken + ":" + formToken;
        }
    }

    $.ajax("api/values", {
        type: "post",
        contentType: "application/json",
        data: {  }, // JSON data goes here
        dataType: "json",
        headers: {
            'RequestVerificationToken': '@TokenHeaderValue()'
        }
    });
</script>
```

Example

When you process the request, extract the tokens from the request header. Then call the AntiForgery.Validate method to validate the tokens. The Validate method throws an exception if the tokens are not valid.

```
void ValidateRequestHeader(HttpRequestMessage request)
{
    string cookieToken = "";
    string formToken = "";

    IEnumerable<string> tokenHeaders;
    if (request.Headers.TryGetValues("RequestVerificationToken", out tokenHeaders))
    {
        string[] tokens = tokenHeaders.First().Split(':');
        if (tokens.Length == 2)
        {
            cookieToken = tokens[0].Trim();
            formToken = tokens[1].Trim();
        }
    }
    AntiForgery.Validate(cookieToken, formToken);
}
```

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Web Forms
Attributes	N/A
References	Take Advantage of ASP.NET Built-in Features to Fend Off Web Attacks
Steps	CSRF attacks in WebForm based applications can be mitigated by setting ViewStateUserKey to a random string that varies for each user - user ID or, better yet, session ID. For a number of technical and social reasons, session ID is a much better fit because a session ID is unpredictable, times out, and varies on a per-user basis.

Example

Here's the code you need to have in all of your pages:

```
void Page_Init (object sender, EventArgs e) {
   ViewStateUserKey = Session.SessionID;
   :
}
```

Set up session for inactivity lifetime

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic

TITLE	DETAILS
Attributes	N/A
References	HttpSessionState.Timeout Property
Steps	Session timeout represents the event occurring when a user does not perform any action on a web site during a interval (defined by web server). The event, on server side, change the status of the user session to 'invalid' (for example "not used anymore") and instruct the web server to destroy it (deleting all data contained into it). The following code example sets the timeout session attribute to 15 minutes in the Web.config file.

Example

```XML code <system.web> </system.web>

```
Enable Threat detection on Azure SQL
```

| TITLE | DETAILS |
| --- | --- |
| **Component** | Web Application |
| **SDL Phase** | Build |
| **Applicable Technologies** | Web Forms |
| **Attributes** | N/A |
| **References** | forms Element for authentication (ASP.NET Settings Schema) |
| **Steps** | Set the Forms Authentication Ticket cookie timeout to 15 minutes |

### Example

```XML code

```
Title	Details
Component	Web Application
SDL Phase	Build
Applicable Technologies	Web Forms, MVC5
Attributes	EnvironmentType - OnPrem
References	[asdeqa](https://skf.azurewebsites.net/Mitigations/Details/wefr)
Steps	When the web application is Relying Party and ADFS is the STS, the lifetime of the authentication cookies - FedAuth tokens - can be set by the following configuration in web.config:

Example
```XML
  <system.identityModel.services>
    <federationConfiguration>
      <!-- Set requireSsl=true; domain=application domain name used by FedAuth cookies (Ex: .gdinfra.com); -->
      <cookieHandler requireSsl="true" persistentSessionLifetime="0.0:15:0" />
      <!-- Set requireHttps=true; -->
      <wsFederation passiveRedirectEnabled="true" issuer="http://localhost:39529/"
realm="https://localhost:44302/" reply="https://localhost:44302/" requireHttps="true"/>
      <!--
      Use the code below to enable encryption-decryption of claims received from ADFS. Thumbprint value varies
based on the certificate being used.
      <serviceCertificate>
        <certificateReference findValue="4FBBBA33A1D11A9022A5BF3492FF83320007686A" storeLocation="LocalMachine"
storeName="My" x509FindType="FindByThumbprint" />
      </serviceCertificate>
      -->
    </federationConfiguration>
  </system.identityModel.services>
```

Example

Also the ADFS issued SAML claim token's lifetime should be set to 15 minutes, by executing the following powershell command on the ADFS server:

```
Set-ADFSRelyingPartyTrust -TargetName "<RelyingPartyWebApp>" -ClaimsProviderName @("Active Directory") -
TokenLifetime 15 -AlwaysRequireAuthentication $true
```

Implement proper logout from the application

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
```

TITLE	DETAILS
Steps	Perform proper Sign Out from the application, when user presses log out button. Upon logout, application should destroy user's session, and also reset and nullify session cookie value, along with resetting and nullifying authentication cookie value. Also, when multiple sessions are tied to a single user identity, they must be collectively terminated on the server side at timeout or logout. Lastly, ensure that Logout functionality is available on every page.

Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET Web APIs

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker can carry out actions in the security context of a different user's established session on a web site. The goal is to modify or delete content, if the targeted web site relies exclusively on session cookies to authenticate received request. An attacker could exploit this vulnerability by getting a different user's browser to load a URL with a command from a vulnerable site on which the user is already logged in. There are many ways for an attacker to do that, such as by hosting a different web site that loads a resource from the vulnerable server, or getting the user to click a link. The attack can be prevented if the server sends an additional token to the client, requires the client to include that token in all future requests, and verifies that all future requests include a token that pertains to the current session, such as by using the ASP.NET AntiForgeryToken or ViewState.

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	N/A
References	Preventing Cross-Site Request Forgery (CSRF) Attacks in ASP.NET Web API

TITLE	DETAILS
Steps	Anti-CSRF and AJAX: The form token can be a problem for AJAX requests, because an AJAX request might send JSON data, not HTML form data. One solution is to send the tokens in a custom HTTP header. The following code uses Razor syntax to generate the tokens, and then adds the tokens to an AJAX request.

Example

```
<script>
    @functions{
        public string TokenHeaderValue()
        {
            string cookieToken, formToken;
            AntiForgery.GetTokens(null, out cookieToken, out formToken);
            return cookieToken + ":" + formToken;
        }
    }
    $.ajax("api/values", {
        type: "post",
        contentType: "application/json",
        data: {  }, // JSON data goes here
        dataType: "json",
        headers: {
            'RequestVerificationToken': '@TokenHeaderValue()'
        }
    });
</script>
```

Example

When you process the request, extract the tokens from the request header. Then call the AntiForgery.Validate method to validate the tokens. The Validate method throws an exception if the tokens are not valid.

```
void ValidateRequestHeader(HttpRequestMessage request)
{
    string cookieToken = "";
    string formToken = "";

    IEnumerable<string> tokenHeaders;
    if (request.Headers.TryGetValues("RequestVerificationToken", out tokenHeaders))
    {
        string[] tokens = tokenHeaders.First().Split(':');
        if (tokens.Length == 2)
        {
            cookieToken = tokens[0].Trim();
            formToken = tokens[1].Trim();
        }
    }
    AntiForgery.Validate(cookieToken, formToken);
}
```

Example

Anti-CSRF and ASP.NET MVC forms - Use the AntiForgeryToken helper method on Views; put an Html.AntiForgeryToken() into the form, for example,

```
@using (Html.BeginForm("UserProfile", "SubmitUpdate")) {
    @Html.ValidationSummary(true)
    @Html.AntiForgeryToken()
    <fieldset>
}
```

Example

The example above will output something like the following:

```
<form action="/UserProfile/SubmitUpdate" method="post">
    <input name="__RequestVerificationToken" type="hidden"
value="saTFWpkKN08YazFtN6c4YbZAmsEwG0srqlUqqloi/fVgeV2ciIFVmelvzwRZpArs" />
    <!-- rest of form goes here -->
</form>
```

Example

At the same time, Html.AntiForgeryToken() gives the visitor a cookie called __RequestVerificationToken, with the same value as the random hidden value shown above. Next, to validate an incoming form post, add the [ValidateAntiForgeryToken] filter to the target action method. For example:

```
[ValidateAntiForgeryToken]
public ViewResult SubmitUpdate()
{
// ... etc.
}
```

Authorization filter that checks that:

- The incoming request has a cookie called __RequestVerificationToken
- The incoming request has a `Request.Form` entry called __RequestVerificationToken
- These cookie and `Request.Form` values match Assuming all is well, the request goes through as normal. But if not, then an authorization failure with message "A required anti-forgery token was not supplied or was invalid".

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	Identity Provider - ADFS, Identity Provider - Azure AD
References	Secure a Web API with Individual Accounts and Local Login in ASP.NET Web API 2.2

TITLE	DETAILS
Steps	If the Web API is secured using OAuth 2.0, then it expects a bearer token in Authorization request header and grants access to the request only if the token is valid. Unlike cookie based authentication, browsers do not attach the bearer tokens to requests. The requesting client needs to explicitly attach the bearer token in the request header. Therefore, for ASP.NET Web APIs protected using OAuth 2.0, bearer tokens are considered as a defense against CSRF attacks. Please note that if the MVC portion of the application uses forms authentication (i.e., uses cookies), anti-forgery tokens have to be used by the MVC web app.

Example

The Web API has to be informed to rely ONLY on bearer tokens and not on cookies. It can be done by the following configuration in `WebApiConfig.Register` method: ```C-Sharp code config.SuppressDefaultHostAuthentication(); config.Filters.Add(new HostAuthenticationFilter(OAuthDefaults.AuthenticationType));

```
The SuppressDefaultHostAuthentication method tells Web API to ignore any authentication that happens before the request reaches the Web API pipeline, either by IIS or by OWIN middleware. That way, we can restrict Web API to authenticate only using bearer tokens.
```

LOCKHEED MARTIN

Seven Ways to Apply the Cyber Kill Chain® with a Threat Intelligence Platform

A White Paper Presented by:
Lockheed Martin Corporation

© 2015 Lockheed Martin Corporation. All rights reserved.

© 2015 Lockheed Martin Corporation. All rights reserved.

ABSTRACT

Threat Intelligence Platforms (TIP) are an emerging technology supporting organizations as they consume and then act on cyber intelligence. Lockheed Martin believes that a TIP helps an organization transition from relying solely on external intelligence sources to producing their own intelligence based on what is observed in their environment. The result is elevated cyber maturity and improved resilience against attackers.

1. INTRODUCTION

Within the past decade, computer network defense has shifted from a culture of sharing minimal information to one of intelligence overload. Previously, information regarding system breaches, malware, or attack attribution was rarely shared between organizations. Today the more common issue is how to sift through all the emails, reports, and indicators to identify actionable intelligence.

Threat intelligence is evidence-based knowledge about a threat that can be used to inform decisions regarding the response to that threat (McMillan, 2013). It includes the details of the motivations, intent, and capabilities of threat actors (Holland, 2014). In order to successfully defend against the multitude of Advanced Persistent Threats (APT) facing an organization, consuming external threat intelligence has become an increasingly important aspect of cybersecurity. However, exactly how to ingest the intelligence and successfully leverage it within an organization's environment often remains a challenge.

In addition to external intelligence, an influx of data from one's own organization can further complicate matters. Alerts from a myriad of technologies including Intrusion Detection Systems (IDS), firewalls, mail scanners, Host Intrusion Prevention Systems (HIPS), and proxies can overwhelm a defender who is trying to respond to and disposition each alert. External intelligence can quickly become an afterthought as there is no time to evaluate and implement countermeasures, making it useless. Conversely, external intelligence fed directly into these tools results only in more noise if it is not properly vetted for one's environment and mitigations are not appropriately tuned.

In order to process all of this internal and external data and have it result in actionable intelligence, a TIP can be employed. A TIP is the central management repository for all external and internal intelligence, and can provide the mechanism to act upon this intelligence.

This paper is organized as follows: section two of this paper documents related work on defining the requisite components of a TIP. Section three introduces an expansion of this definition that is based upon the Intelligence Driven Defense® approach to computer network defense. Section four outlines the seven ways an organization can apply the Cyber Kill Chain® framework in their environment using a TIP. Section five introduces the Palisade™ solution, Lockheed Martin's Threat Intelligence Platform, and section six summarizes the paper.

2. RELATED WORK

The analyst firms Gartner, Inc. and Forrester Research provide insight on the value of employing a Threat Intelligence Platform, and have each offered definitions of its components.

Gartner states that organizations looking for effective, automated methods to ingest multiple sources and formats of threat intelligence, correlate them, and act upon the information more efficiently should investigate using a TIP. They suggest that a TIP has the following six capabilities: 1) Collect intelligence from multiple sources in multiple formats and automatically enrich the intelligence; 2) Correlate intelligence and help determine which sources are the most effective; 3) Categorize intelligence to help perform analytics on threats, recognize tactics, techniques, and

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 336

procedures (TTPs), and understand relationships through modeling and visualizations; 4) Integrate the intelligence with workflow processes to further enrich existing data; 5) Provide action by supporting various integrations via Application Program Interfaces (APIs) and email notifications; and 6) Support the sharing of intelligence among trusted communities (Lawson, 2014).

Forrester states that the specific benefit of a TIP is providing Security Operation Centers with analysis platforms that can orchestrate intelligence activities. The platform should be capable of ingesting and normalizing threat intelligence, rating the value of intelligence sources, providing an analyst workspace, supporting visualization and pivoting, enriching the intelligence, and enabling the internal and external collaboration and sharing of the intelligence (Holland, 2015).

3. EXPANDING THE OBJECTIVE OF A THREAT INTELLIGENCE PLATFORM

Gartner and Forrester laid out the intention and minimal requirements for a TIP. Lockheed Martin expands on those requirements to ensure a TIP provides value in all environments. While a TIP should enable the collection of vast amounts of external intelligence, it should also provide the foundation for an organization to mature into a producer of intelligence.

An Intelligence Driven Defense® approach to computer network defense has been advocated by Lockheed Martin for over five years (Hutchins et al., 2010). The central theme is that you can learn from past intrusion attempts to generate actionable intelligence that can be put back into your defenses to make you more resilient against advanced adversaries. Connecting the dots between seemingly different data points allows a defender to recognize relationships among incidents and identify common characteristics. This allows the analyst to understand the tactics, techniques, and procedures (TTPs) of their adversary. The more complete the profile that is built on an adversary directly targeting your network, the greater the opportunity to be one step ahead of them.

Without a dedicated tool to store all the information and context gleaned from an investigation, identifying those commonalities becomes impractical. In the absence of a TIP, an analyst has few options to capture the data that otherwise could provide keen insights on an attack. Spreadsheets or text files on a SharePoint site do not allow an analyst much flexibility to correlate events by leveraging historical knowledge. For example, if you identify malware that beacons out to an IP address, how can you determine whether you have previously seen that IP address in your environment? A TIP should provide the location for an analyst to input all the indicators, context, notes, intelligence, recommendations, evidence, and adversary profile information discovered during their investigation and ensure everything is easily searchable. Furthermore, a TIP should support the Cyber Kill Chain® framework for this analysis as it is central to an Intelligence Driven Defense® posture. This approach can help an analyst quickly determine the severity of an attack and then identify gaps in either the analysis or the organization's defenses.

External intelligence can be an important facet of computer network defense, and a TIP should support its consumption. However, dependence on those sources is reduced as an organization's cyber posture matures. External intelligence can provide insight into potential threats while an organization works to adopt an intelligence driven approach and develop their own analytic capabilities. As internal capabilities mature, external sources should be used to correlate what is seen in the environment and provide greater context and insight to the adversary's TTPs. The TIP should also provide a mechanism to measure those external sources, making it easy to determine which provide value. Those not beneficial to your organization should be removed – an action that will reduce noise and save money.

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 337

4. SEVEN WAYS TO APPLY THE CYBER KILL CHAIN® WITH A THREAT INTELLIGENCE PLATFORM

The Cyber Kill Chain® framework for computer network defense is not something that can be placed into an enterprise's defenses. However, there are seven ways to apply it within an organization to mitigate risk, build true resilience, better communicate, and meaningfully measure results (Hutchins, 2014). Each of these applications can be achieved or augmented with the use of a TIP that manages threat intelligence and enables an analyst to identify broader campaign activity by leveraging historical data.

4.1 PRIORITIZE SENSOR ALERTS

With so many alerts coming from so many sources, which alerts should be analyzed first? Which sensor is the most important and demands immediate attention? In order to answer these questions, an organization should list the events that are produced by each sensor and map them to the relevant stage of the Cyber Kill Chain®. The further down the attack chain to which an event maps, the higher the priority of that alert. For example, an event produced from a Network Intrusion Detection System (NIDS) maps to either the first stage, Reconnaissance, or the third stage, Delivery. However, an event from a HIPS maps to stage four or five, Installation or Exploitation, or perhaps even stage seven, Action on Objectives. Thus alerts coming from the HIPS should be evaluated first, because they correspond to the potentially more damaging event.

A proper Threat Intelligence Platform can help an organization with this by automatically ingesting alerts from a Security Information and Event Management (SIEM) or SIEM-like device that aggregates and normalizes events. The alerts should be ingested by the TIP and prioritized based on the original source of the alert. Those mapped to later stages of the Cyber Kill Chain® process should be associated with a higher priority and be displayed for an analyst to respond to first. The TIP should also provide the functionality for the analyst to respond to the alert within the tool. That way if it is an intrusion or attempted intrusion, indicators can be documented, and notes recorded to explain how the investigation was conducted. Standard courses of action can also be documented so that similar alerts are dispositioned and addressed in a repeatable and efficient manner.

4.2 PRIORITIZE ESCALATION

Another common problem for analyst teams is knowing with whom to report each attempted intrusion into the network. Investigation of an incident through the Cyber Kill Chain® framework can quickly determine the severity of the attack. Was a malicious email delivered to a recipient's inbox? Is a computer on the network communicating with a known command and control IP address? Once again, the further down the kill chain the attack progresses, the higher the incident should be reported in an organization's management chain. If an attack results in Action on Objectives – meaning the adversary was able to exfiltrate data from the network – the impact could potentially require notification to the organization's CEO or Board of Directors. On the other hand, a "shielded" attack, where malware has been installed on a host computer, but there is no command and control, may only need to be reported up to a Chief Information Security Officer, as this sort of compromise has much less impact on the organization.

A TIP can provide the platform for an analyst to record all results from the investigation of an intrusion. When an alert comes in, an analyst gathers data points and documents them in the TIP. The analyst then studies the data to understand the alert and identify how far the intrusion progressed. The TIP can map key pieces of intelligence to the stage of the Cyber Kill Chain® stage from which it was pulled to determine how far the intrusion advanced and the level of impact to the organization. This can then align directly to the level of escalation required to communicate the impact.

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 338

4.3 PRIORITIZE INVESTMENT

Establishing a return on investment for security products can be an arduous task. Building a business case for future investments can be even more difficult. An organization can visually display where protections exist that detect, deny, disrupt, degrade, and/or deceive an intrusion attempt at each stage of the Cyber Kill Chain® process. A graph showing each stage of the process on one axis and the countermeasure along the other can then identify the tool in place to perform the mitigation at that stage in the box where each intersects. An empty box indicates gaps where investments can be made to further enhance the organization's protections. An example of this type of graph is shown in Figure 1.

A TIP can provide this type of graph as output by associating mitigations for every intrusion attempt to each stage of the attack. A case can then be made to invest in tools that would fill the gap areas in an organization's countermeasures. Not every box would need to be filled for every organization, but the further along the Cyber Kill Chain®, the greater the priority. For example, there may not be a need to degrade Reconnaissance activity, but there would certainly be a need to detect a command and control channel.

Additionally, TIPs could help an organization identify data gaps and source collection requirements. As analysts investigate intrusions and intrusion attempts, those stages lacking data can lead to investments in tools to close the gap.

4.4 MEASURE EFFECTIVENESS

As previously stated, attempted attacks should be investigated, analyzed, and synthesized using the Cyber Kill Chain® framework. The stage at which the attack was stopped should be documented. The goal is that over time attacks would be stopped earlier and earlier in the process.

Once again, a TIP can be the platform for an analyst to record the results of the investigation, analysis, and synthesis of each attempted intrusion. The identification of where in the Cyber Kill Chain® process the intrusion was stopped can also be recorded as a data point within the TIP. Another output of the TIP could then be a graphical representation of these stopping points over a period of time. Ideally the TIP would show a trend of analysts recognizing and preventing adversaries from reaching later stages of the kill chain, and thus becoming more effective. An example of this graph is shown in Figure 2.

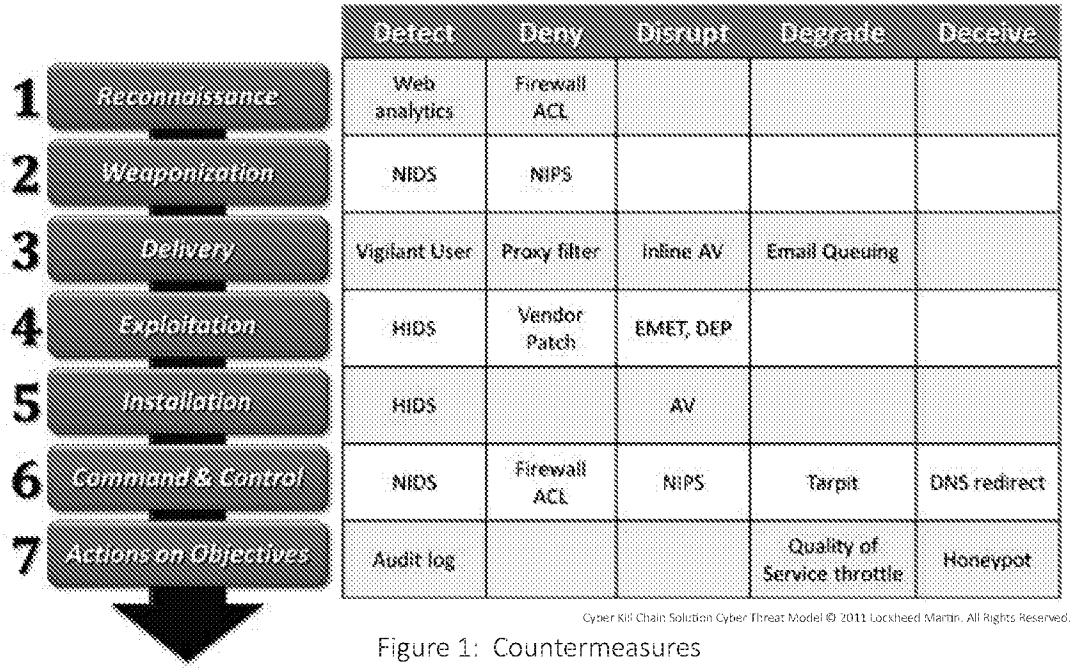


Figure 1: Countermeasures

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 339

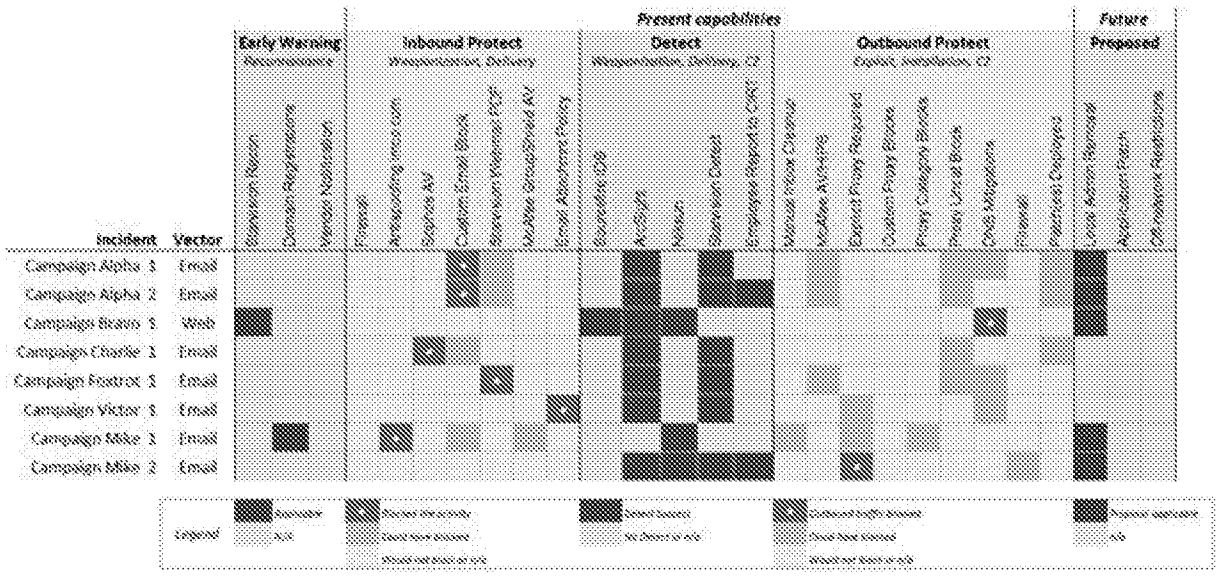


Figure 2: Measuring Effectiveness Scorecard

4.5 MEASURE RESILIENCE

If an intrusion is blocked in an early Cyber Kill Chain® stage and the organization's detections and mitigations are effective, is the organization also resilient? In other words, if it weren't identified and stopped at that point, would it have been caught lower down the kill chain? Were there layers of security in place that would protect an organization in a future attack if the adversary changed that one particular tactic, signature, or indicator? This is determined after synthesizing an attack through the entire Cyber Kill Chain® cycle – regardless of how early it is stopped. Understanding what would have happened had the adversary been successful is just as important as understanding what did happen. Understanding where defenses are in place at every stage of the kill chain – including those past the point of initial detection – is understanding how resilient the organization is. The goal is to be effective and block intrusions early in the Cyber Kill Chain®, but also to be resilient and have layers of countermeasures in place.

Once again a TIP can associate countermeasures to each stage of the Cyber Kill Chain® framework for every intrusion attempt. This includes those later stages that the attempted intrusions did not actually reach. The TIP can reveal the level of resiliency for each investigated incident and could provide the mechanism to compare the resiliency against intrusions over a period of time. As with effectiveness, the trend would hopefully show that the organization is becoming more resilient over time, with multiple countermeasures in place for each intrusion attempt.

4.6 MEASURE ANALYTIC COMPLETENESS

As has been previously discussed, an analyst's investigation does not stop with identifying and stopping an attack. One must also analyze how the attack occurred, and synthesize what would have happened had it not been stopped. This complete understanding of the potential incident can be measured against the Cyber Kill Chain® framework, which can be used as a pseudo checklist to ensure the analyst has mined all possible information. The idea of pulling data from every piece of the investigation is a critical component of the Intelligence Driven Defense® approach, as it is what provides the intelligence that informs one's defenses.

The output of good analysis is a lot of data and intelligence. Managing that data and intelligence is a function of a TIP. Furthermore, a TIP that employs the Cyber Kill Chain® framework

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 340

for an investigation can help ensure analytic completeness. It can force the analyst to think about each step to encourage a complete understanding of the intrusion attempt: Do I understand each of the stages for this attack? Have I fully investigated, analyzed, and synthesized all the information I could possibly collect? A TIP should enable an analyst to become effective at computer network defense by providing a mechanism to produce one's own threat intelligence.

4.7 IDENTIFY AND TRACK CAMPAIGNS

The final way to apply the Cyber Kill Chain® process within an organization is to group intrusions from the same adversary together into campaigns. This broader view provides insight into when a particular adversary attacks, the TTPs they commonly use, and how effective and resilient an organization is against that adversary. An organization can then prioritize and measure themselves against each adversary to further enhance their defenses. Establishing campaigns also provides a standard nomenclature to improve communications. This enables analysts to be more efficient when identifying intrusions, understanding potential next steps, and applying the proper defensive response.

An effective Threat Intelligence Platform enables analysts to determine patterns of malicious behavior learned from previous events to better address future attacks. All the data from previous investigations and the intelligence that is mined from analyzing and synthesizing intrusion attempts must be centrally stored and easy to search to enable identification of these patterns and quick recognition of relationships. Analysts connect intrusions by correlating indicators or TTPs from multiple Cyber Kill Chain® stages to determine that an attack is coming from a previously seen adversary. A powerful example of this is being able to recognize that the adversary scraping your external Web servers looking for documents is the same adversary who then sends a phishing email with a weaponized press release.
The TIP can facilitate this correlation because it holds all the intelligence and previous incident analysis.

Since the TIP is also the central location for all campaign mapping, a Campaign Heat Map or similar measurement of adversary activity should

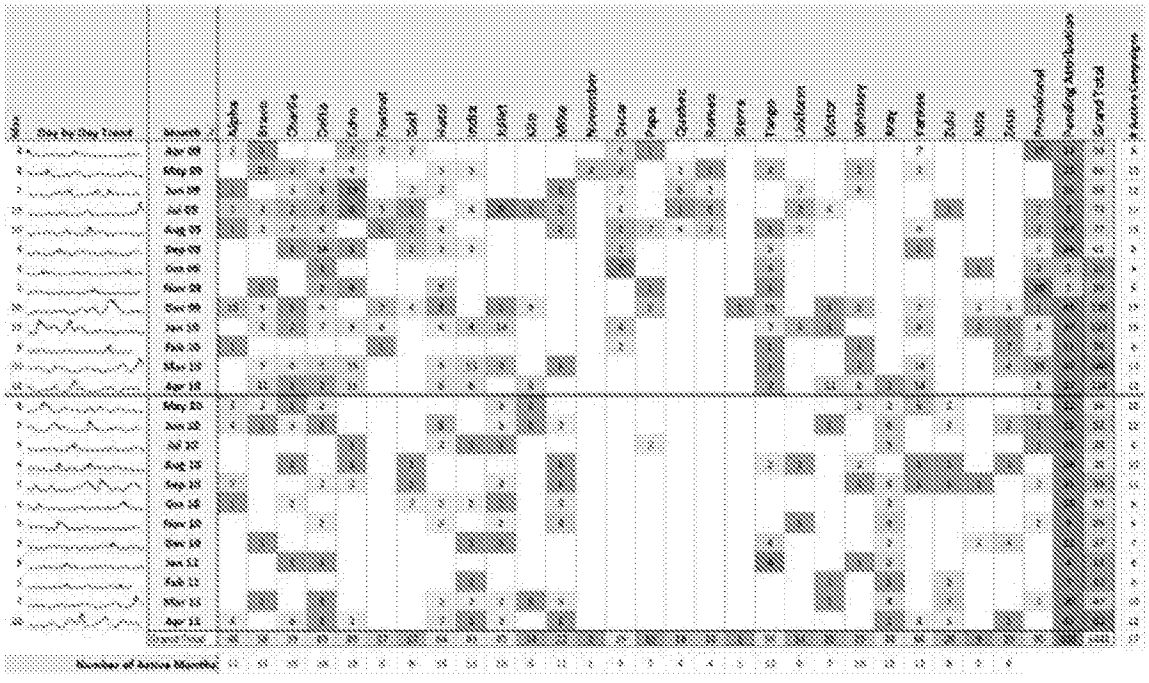


Figure 3. Campaign Heat Map

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 341

be an output of a TIP. This heat map can help organizations build a profile of past and current adversary activity, helping them better understand when, where, and how they will attack again in the future. A sample Campaign Heat Map is shown in Figure 3.

5 . THE PALISADE™ PLATFORM

Lockheed Martin developed the Palisade™ solution to enable analysts to correlate malicious activity. It is not just an intelligence ingest engine or broker; it is a tool for analysts that supports the adoption of the Intelligence Driven Defense® methodology. Palisade™ is a centralized platform that integrates with an organization's SIEM to provide enterprise-wide alerting capability and manage all threat intelligence – internal and external. It also enables the tracking of countermeasures to ensure the organization is properly acting upon the intelligence that has been mined.

The Palisade™ solution empowers analysts to focus their efforts and adopt an Intelligent Driven Defense® approach by providing a platform for incident investigation and response that is aligned to the Cyber Kill Chain® framework. It allows for the identification of broader campaign activity through the connection of disparate events by leveraging historical intelligence from previous incidents. The Palisade™ platform is an enabler for a more effective approach to computer network defense – an approach that is fueled by analyst tradecraft and intelligence.

6 . CONCLUSION

Threat Intelligence Platforms are an emerging technology that help an organization to consume and then act on cyber intelligence. Rather than relying solely on external intelligence, a TIP can also enable an organization to transition to producing their own actionable intelligence. This is a more reliable, sustainable, and cost-effective model for the long-term defenses of an organization. Properly employing a TIP can enhance an organization's cyber maturity and improve resiliency against advanced adversaries through the adoption of an Intelligence Driven Defense® approach to computer network defense. This can mean the difference between a reactive and vulnerable security state and a proactive, or even predictive, computer network defense posture that is capable of identifying and stopping current attacks, and preventing future attacks.

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 342

References

Rick Holland. Threat Intelligence Buyer's Guide, 10 February 2014. URL https://digital-forensics.sans.org/summit-archives/cti_summit2014/Threat_Intelligence_Buyers_Guide_Rick_Holland.pdf

Rick Holland. Threat Intelligence is Like Three Day Potty Training, April 2015. URL https://www.rsaconference.com/writable/presentations/file_upload/cxo-t08r-threat-intelligence-is-like-three-day-potty-training.pdf

Eric Hutchins. Understanding the Cyber Kill Chain™: Applying Intelligence to Computer Network Defense, Rick Holland. Threat Intelligence is Like Three Day Potty Training, April 2015. URL https://www.rsaconference.com/writable/presentations/file_upload/cxo-t08r-threat-intelligence-is-like-three-day-potty-training.pdf

Eric Hutchins, Michael Cloppert, Rohan Amin. Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains, 2010. URL https://isgs-gen.external.lmco.com/sites/ECS/Marketing/White%20Papers%20and%20External%20Publications/Cyber%20Kill%20Chain%20Whitepaper.pdf,

Craig Lawson. Technology Overview for Threat Intelligence Platforms, 10 December 2014. URL https://www.gartner.com/doc/2941522/technology-overview-threat-intelligence-platforms

Rob McMillan. Definition: Threat Intelligence, 16 May 2013. URL https://www.gartner.com/doc/2487216/definition-threat-intelligence

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 343

© 2015 Lockheed Martin Corporation. All rights reserved.

IriusRisk
Ex. 1002 - Page 344

For more information on cybersecurity solutions
Email: cyber.security@lmco.com
Phone: 855-LMCYBER (856) 562-9237
www.lockheedmartin.com/cyber

PIRA# CMK201507003

© 2015 Lockheed Martin Corporation
Cyber Kill Chain Solution Cyber Threat Model © 2011 Lockheed Martin. All Rights Reserved.
LOCKHEED MARTIN, LOCKHEED, the STAR design, CYBER KILL CHAIN, LOCKHEED MARTIN CYBER KILL CHAIN, and INTELLIGENCE DRIVEN DEFENSE trademarks used throughout are registered trademarks in the U.S. Patent and Trademark Office owned by Lockheed Martin Corporation.

IriusRisk
Ex. 1002 - Page 345

Given below are screenshots of a previous version of the system/methods which were implemented in software that was on sale by applicants in the United States at least as early as 12/31/2011. The 2011 version disclosed the following elements associated with the claims as originally presented and, to speed examination, counsel for applicants presents here a listing of the claims as originally presented in the present application, with the items that are known to have been disclosed in the 2011 on-sale version shown in underling and bold and in black font, with the other elements being shown in red font (no bolding and no underlining) which were not included in the 2011 on-sale version.

1. **A threat modeling method, comprising:**

providing one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

providing one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components"); and

in response to receiving one or more user inputs, using one or more user interfaces displayed on one or more computing devices communicatively coupled with the one or more databases:

using the one or more computing devices, reading a data file generated by the third party software application, the data file comprising a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components;

determining using the one or more computing devices, for the subset of third party diagram components, correlated threat model components as defined in the one or more mapping files;

displaying on the one or more user interfaces a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, **the relational diagram defining a threat model; and**

generating, using the one or more computing devices, and displaying, on the one or more user interfaces, a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

2. The method of claim 1, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

3. The method of claim 1, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

4. The method of claim 1, wherein the data file comprises one of a VSD file, a VSDX file, a TMT file, an XML file, a JSON file, a PNG file, and a JPEG file.

5. The method of claim 1, wherein determining, for the subset of third party diagram components, correlated threat model components, comprises first determining whether a user-defined name for one of the subset of third party diagram components has a match in the one or more mapping files and, if no match is found, then determining whether a default title for the third party diagram component, as defined by the third party software application, has a match in the one or more mapping files.

6. The method of claim 5, further comprising, if no match is found for the default title, matching the third party diagram component with a placeholder component and displaying the placeholder component on the relational diagram to indicate that no match was found.

7. The method of claim 1, further comprising determining, for the one or more relationships defined by the data file, correlated threat model components as defined in the one or more mapping files.

8. **A threat modeling system, comprising:**

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components");

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by one of the one or more computing devices, the data file generated by the third party software application, the data file comprising a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of third party diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, **the relational diagram defining a threat model; and**

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

9. The system of claim 8, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

10. The system of claim 8, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

11. The system of claim 8, wherein the data file comprises one of a VSD file, a VSDX file, a TMT file, an XML file, a JSON file, a PNG file, and a JPEG file.

12. The system of claim 8, wherein the one or more mapping files includes a file correlating user-defined names of the third party diagram components with the threat model components.

13. The system of claim 8, wherein the one or more mapping files includes a file correlating default titles of the third party diagram components with the threat model components.

14. The system of claim 8, wherein, for each of the subset of third party diagram components for which no correlated threat model component is found in the one or more mapping files, the diagram interface displays a placeholder component indicative of a match not being found.

15. The system of claim 8, further comprising one or more mapping interfaces displayed on the one or more computing devices and configured to, in response to receiving user input, modify the one or more mapping files.

16. **A threat modeling system, comprising:**

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with VISIO diagram components;

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by the one or more computing devices, the data file comprising a subset of the VISIO diagram components, the data file defining one or more relationships between the subset of VISIO diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of VISIO diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of a computing network, using visual representations of the threat model components correlated with the subset of VISIO diagram components, **the relational diagram defining a threat model; and**

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

17. The system of claim 16, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

18. The system of claim 16, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

19. The system of claim 16, wherein the one or more mapping files includes a file correlating user-defined names of the third party diagram components with the threat model components.

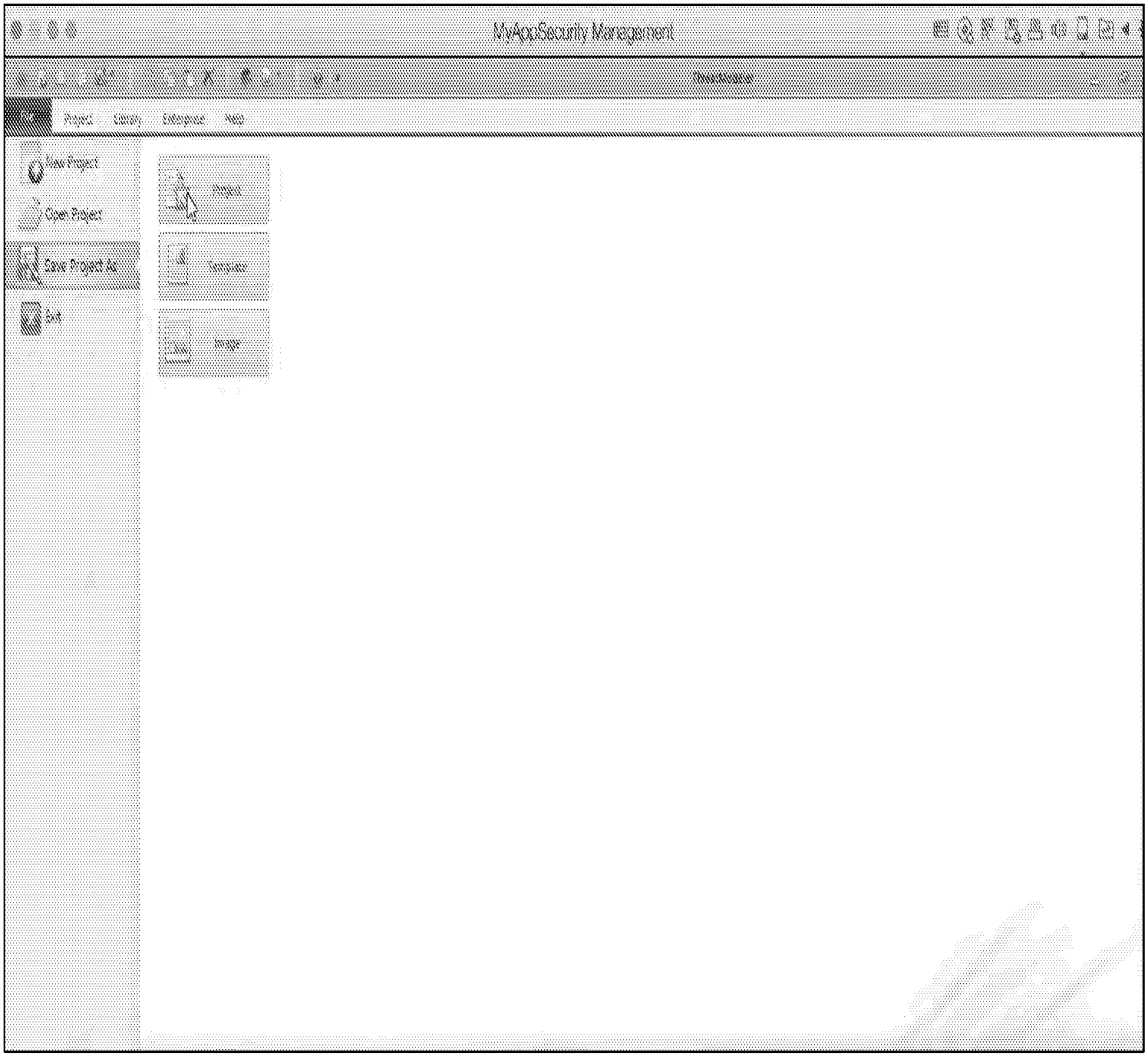
20. The system of claim 16, wherein the one or more mapping files includes a file correlating default titles of the third party diagram components with the threat model components.

Screenshots and explanations of some of the user interfaces of the 2011 version are shown below:

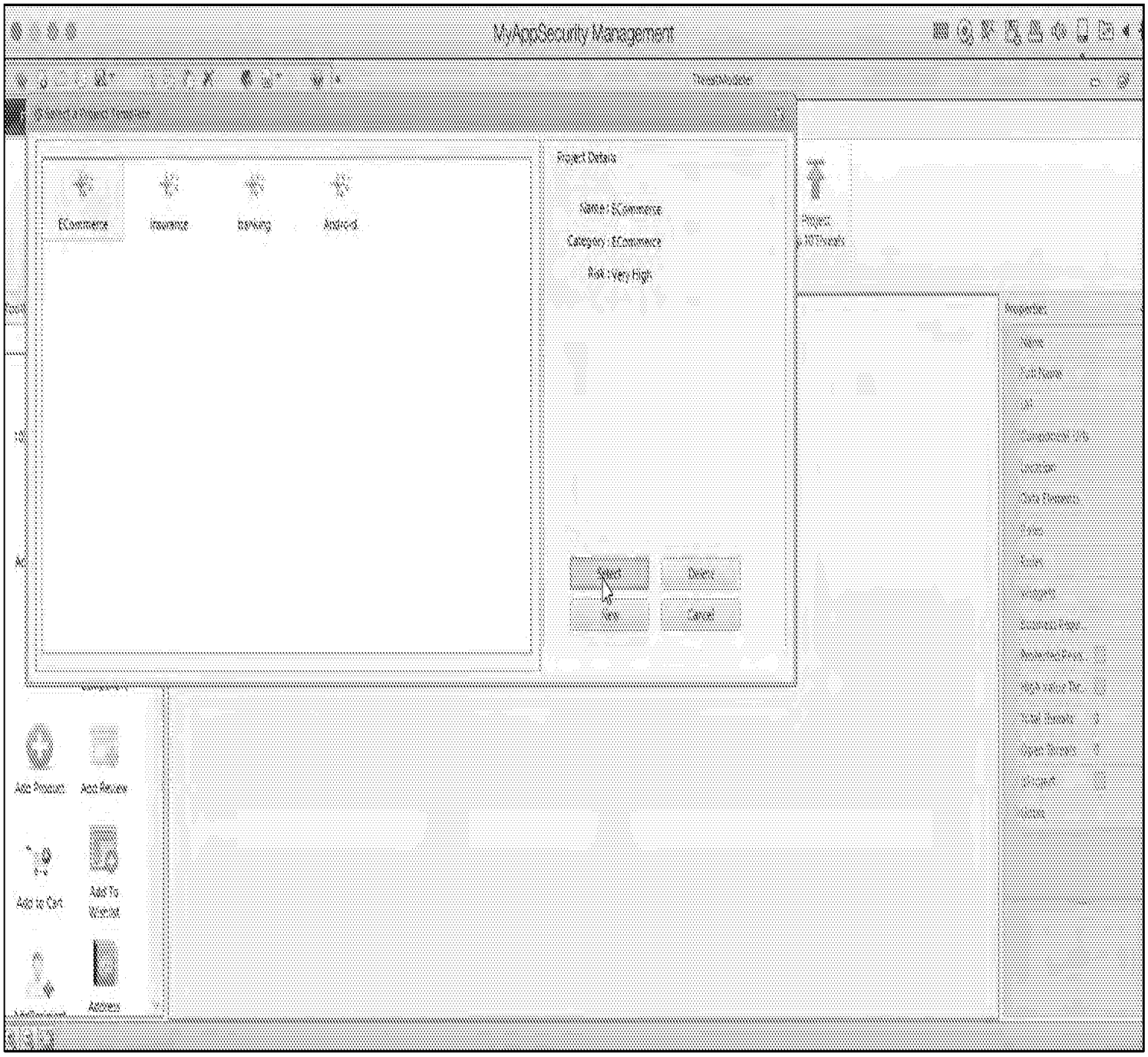
IriusRisk
Ex. 1002 - Page 352



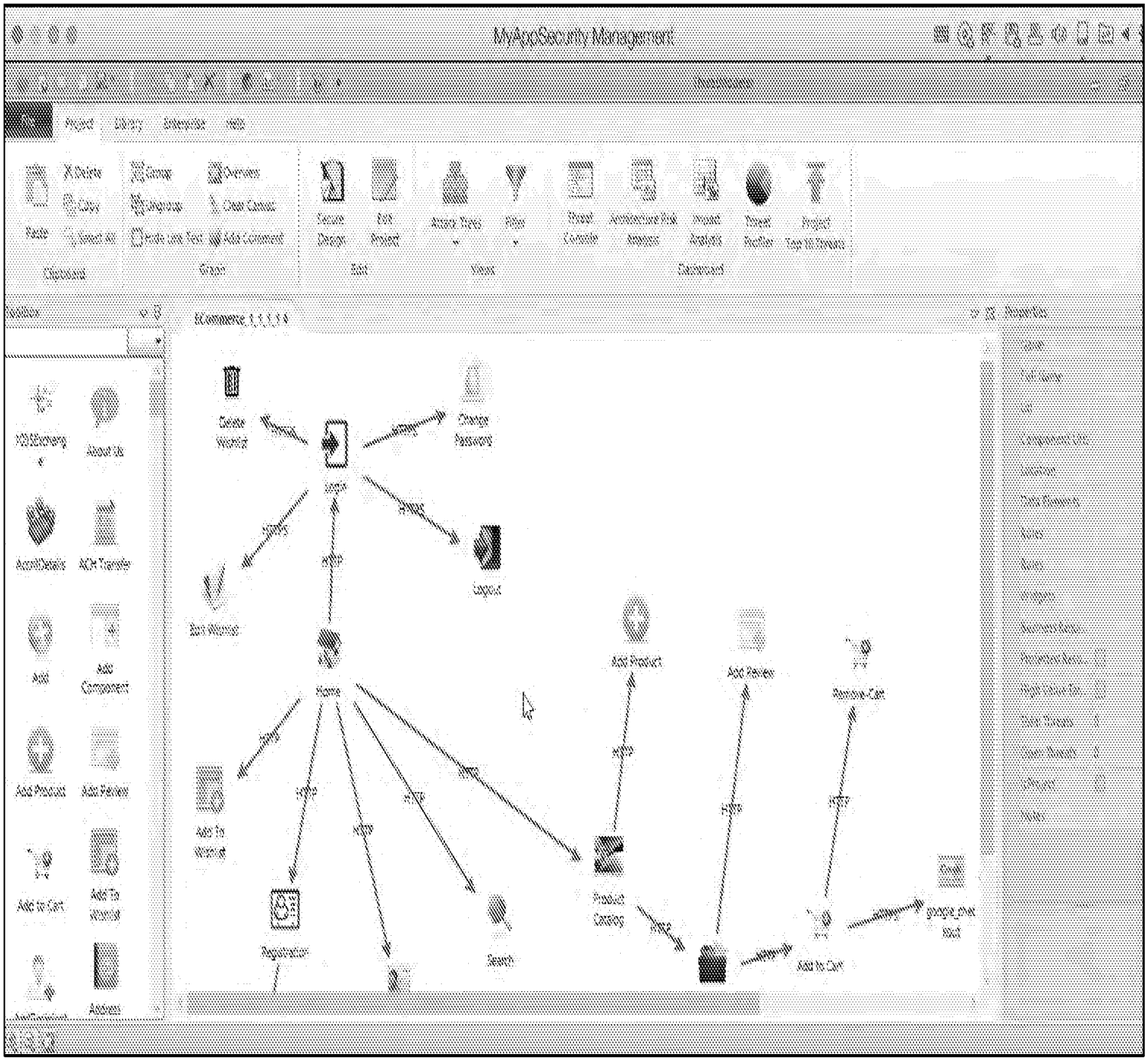
This is a home screen. The user is selecting the "File" menu item at the top left with the cursor.



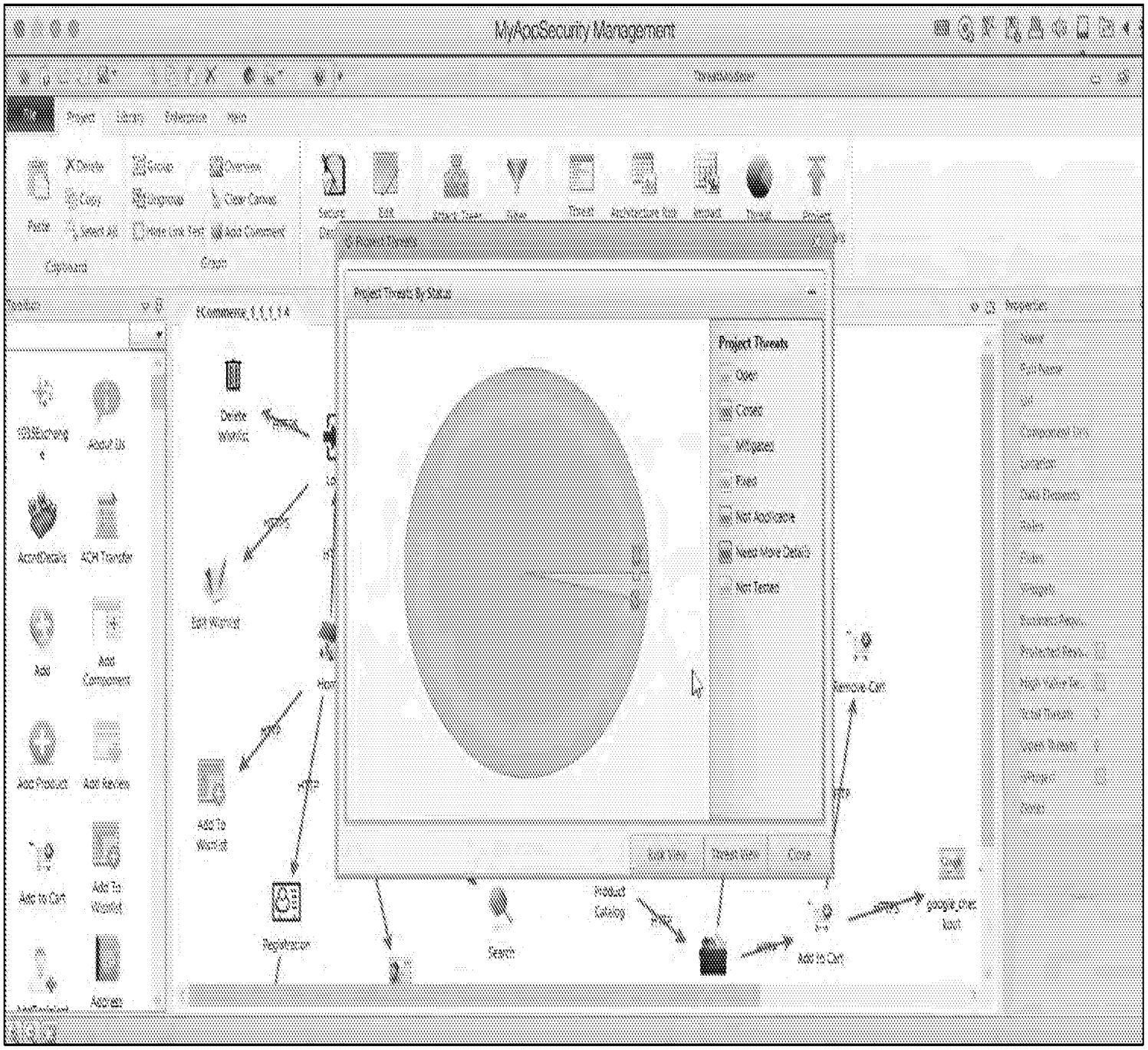
Once the file menu was selected the user was now able to open or create or save a new project, template, or image.



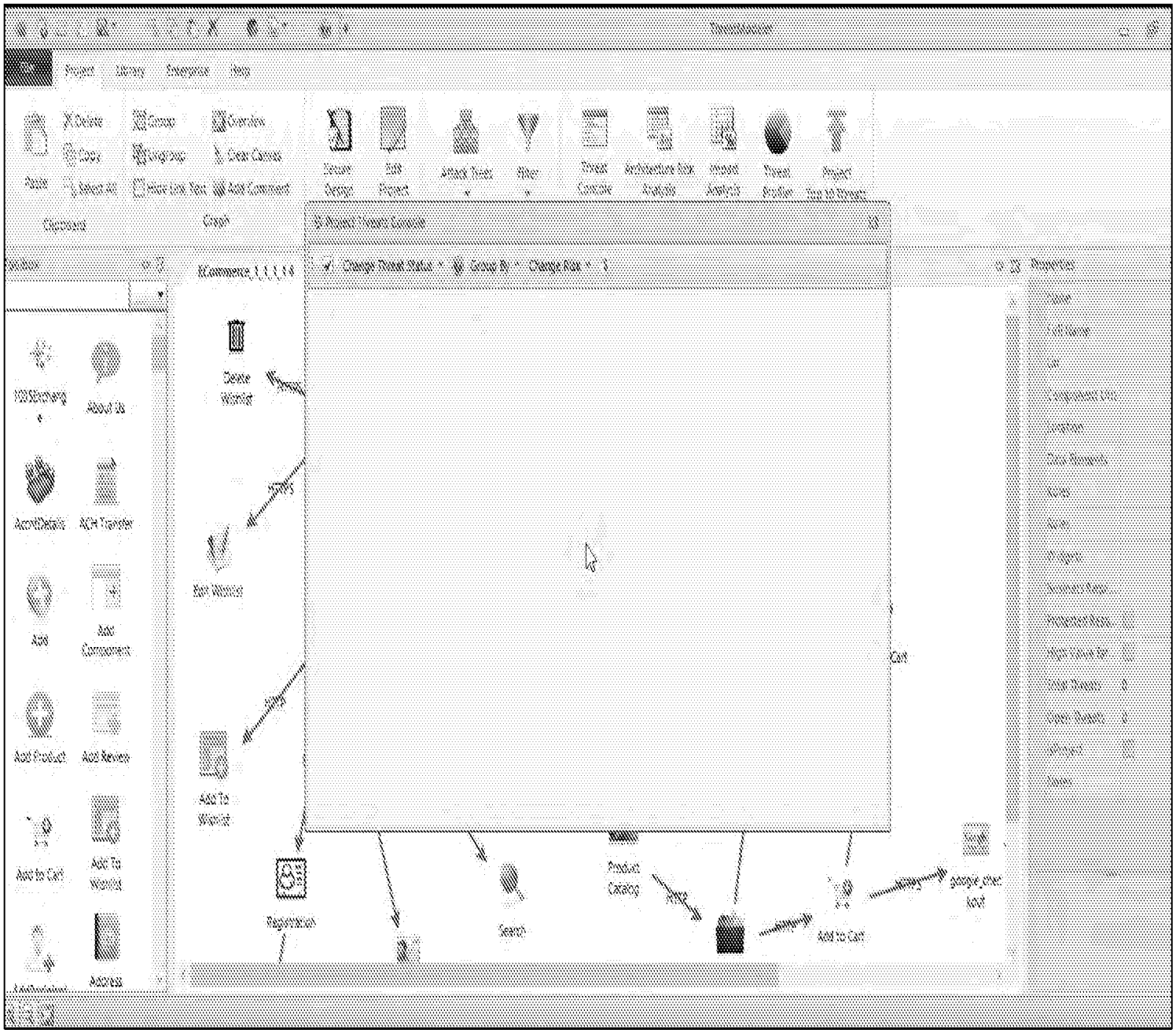
The user is here opening an existing threat model (project) titled Ecommerce, from this screen the user can also delete this threat model, or initiate the creation of a new threat model, or cancel to exit the screen.



Here the user has opened the existing ECommerce threat model, showing components linked to one another using communication protocols. Noticeably absent from this diagram are any compensating controls. This 2011 version did not have compensating controls. It showed the threats in a threat model (discussed later), but had no ability to remove or mitigate threats automatically through compensating controls.

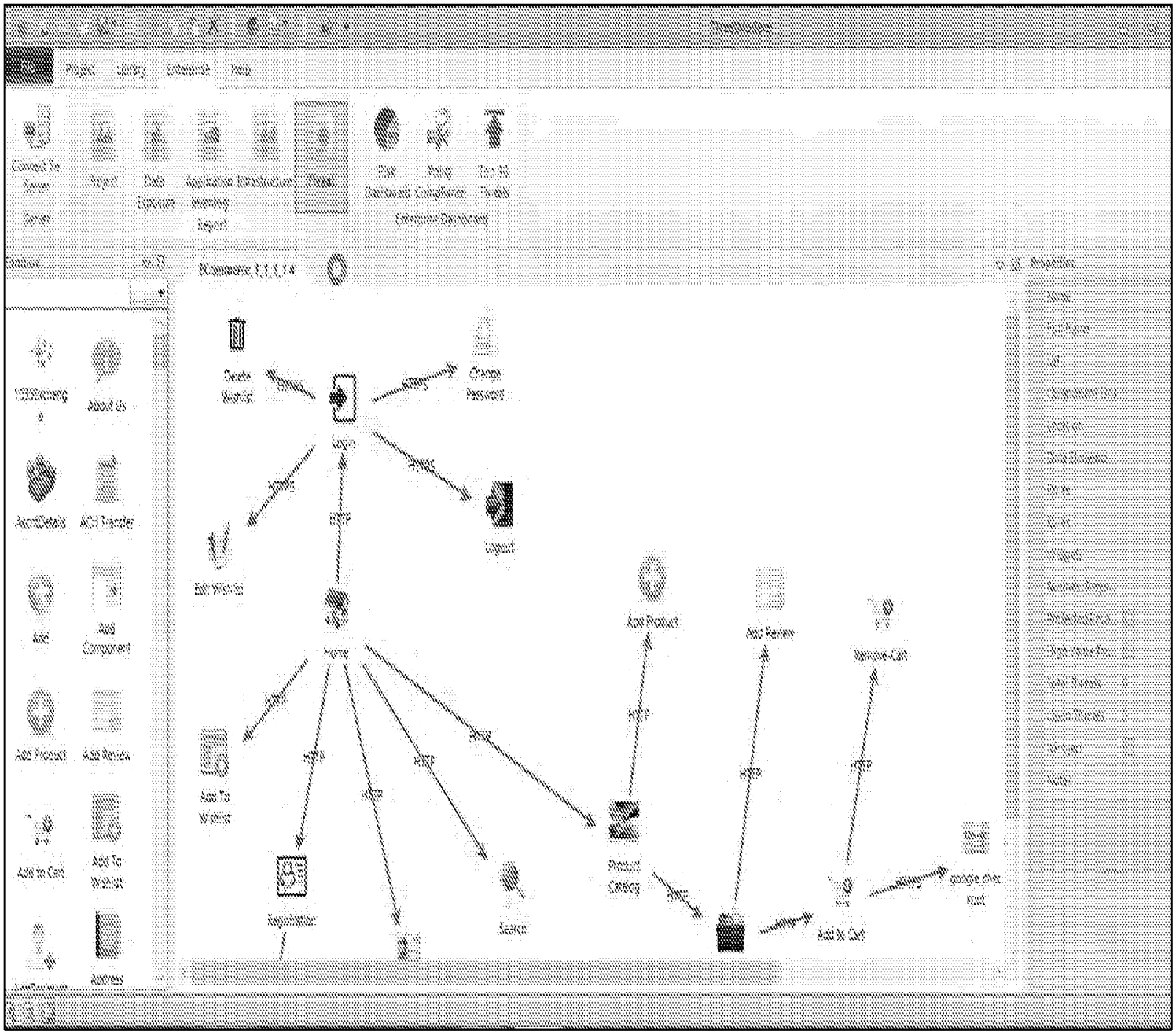


This is an interface showing a pie chart illustrating the percentages of threats by status including open, closed, mitigated, fixed, not applicable, need more details, and not tested.



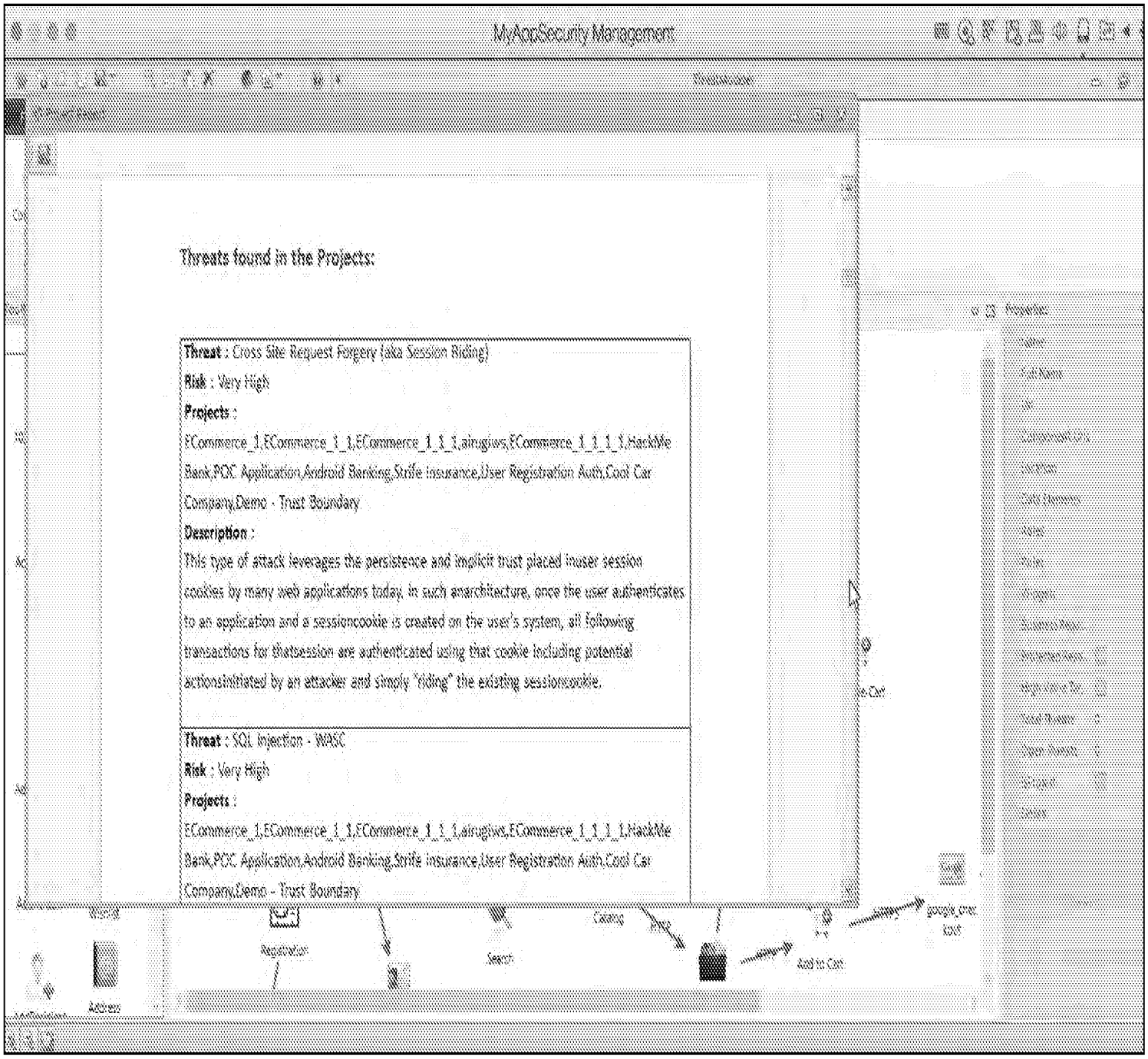
This interface shows the threat report for the diagrammed threat model, though the threats are now not displaying properly. But the user has the ability to change the threat status manually (between the aforementioned statuses shown in the previous screen), change the risk level of any selected threat, and group the threats in different ways to organize the view. When the threats displayed properly they were displayed inside the popup window similar to what is shown below:

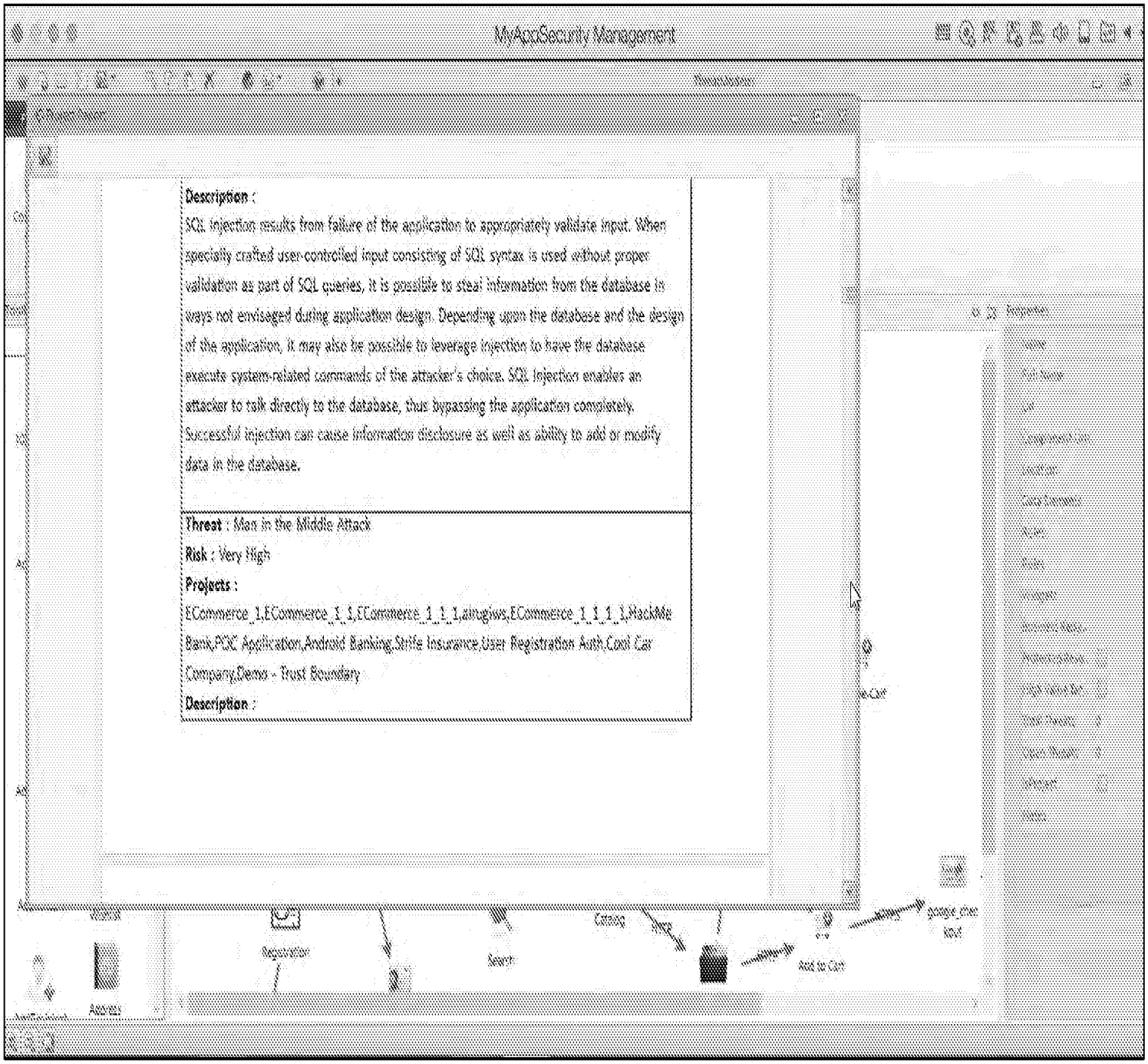
THREAT	RISK	STATUS	SOURCE
ACCT HIJACK	VERY HIGH	OPEN	AWS ELB
INSEC. COMM.	VERY HIGH	OPEN	AWS ELB
PWD BRUTE F...	HIGH	OPEN	APP
SENS DATA EXP	VERY HIGH	OPEN	APP
SQL INJECTION	HIGH	MITIGATED	APP
BLIND SQL INJ	HIGH	MITIGATED	APP
IDENT SPOOF	MEDIUM	OPEN	WEB BR
TARGETED MAL	VERY HIGH	OPEN	WEB BR
CLICKJACKING	VERY HIGH	OPEN	APP
X SITE TRACING	VERY HIGH	OPEN	WEB SVR
IIS SNIFFING	VERY HIGH	OPEN	WCF SVC
XML SVC DEN	VERY HIGH	OPEN	WEB SVC
BUFFER OVFLW	VERY HIGH	OPEN	MS WORD
FILE MANIP	MEDIUM	MITIGATED	FILE SYS
PHYS THEFT	VERY HIGH	MITIGATED	LAPTOP
BLUEJACKING	VERY HIGH	OPEN	BT PORT

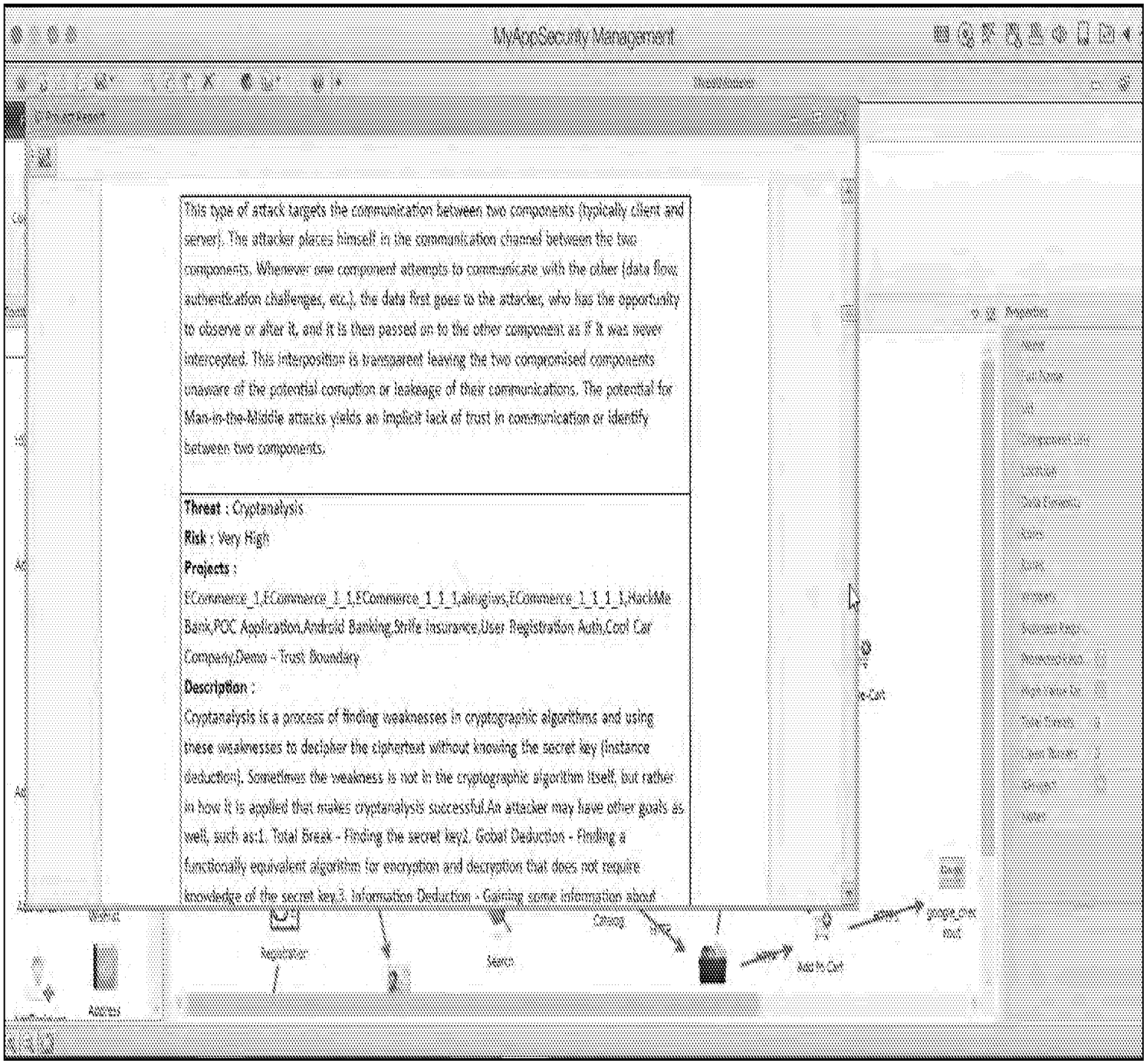


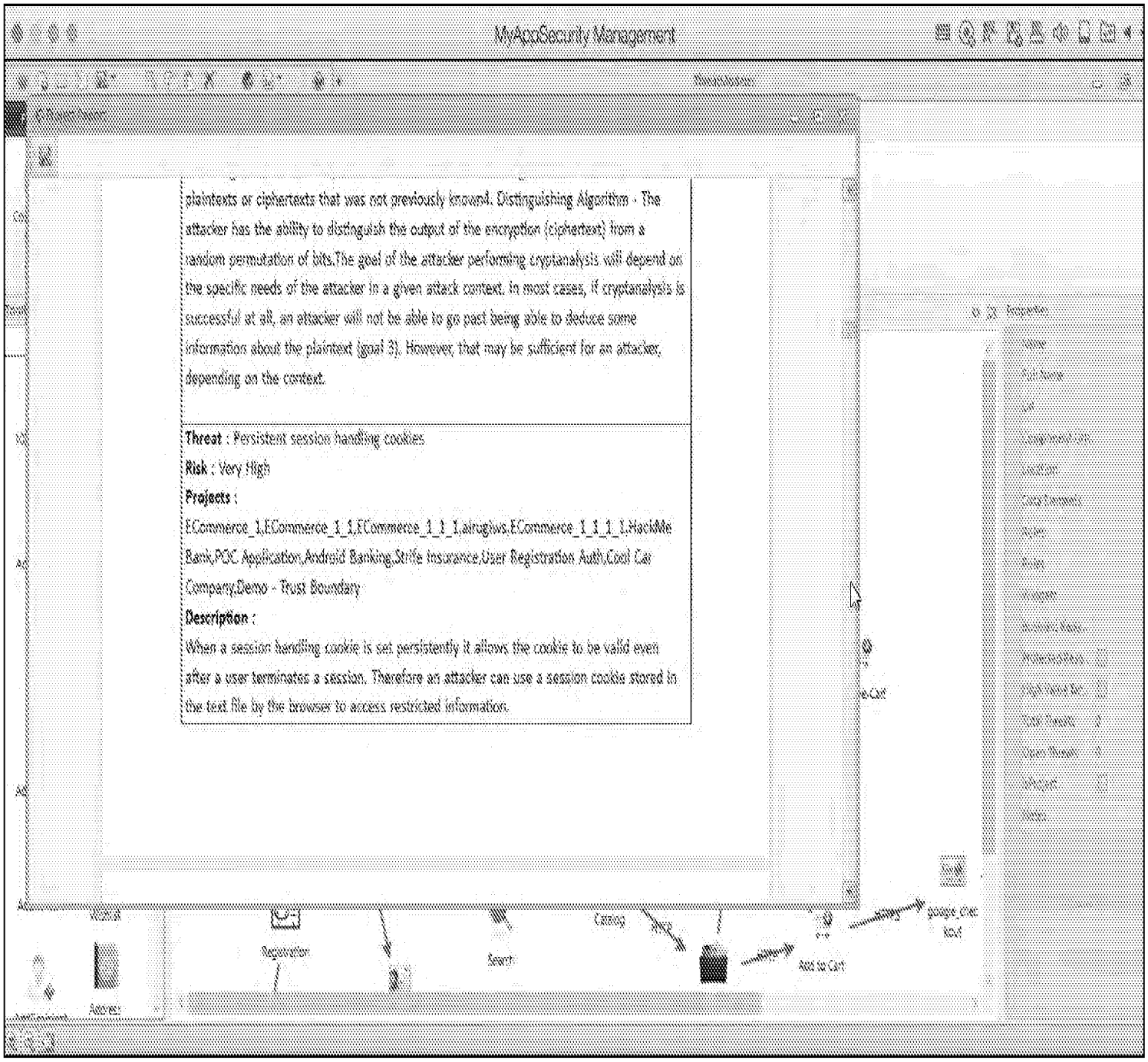
From this screen it can be seen that the user has selected the Enterprise tab at the top and, when selecting a Threat selector, can generate a MS Word version of the threat report. This first several pages of this report are shown in the screens below, and the bottommost image shows the user saving a copy of the report as an MS Word file.

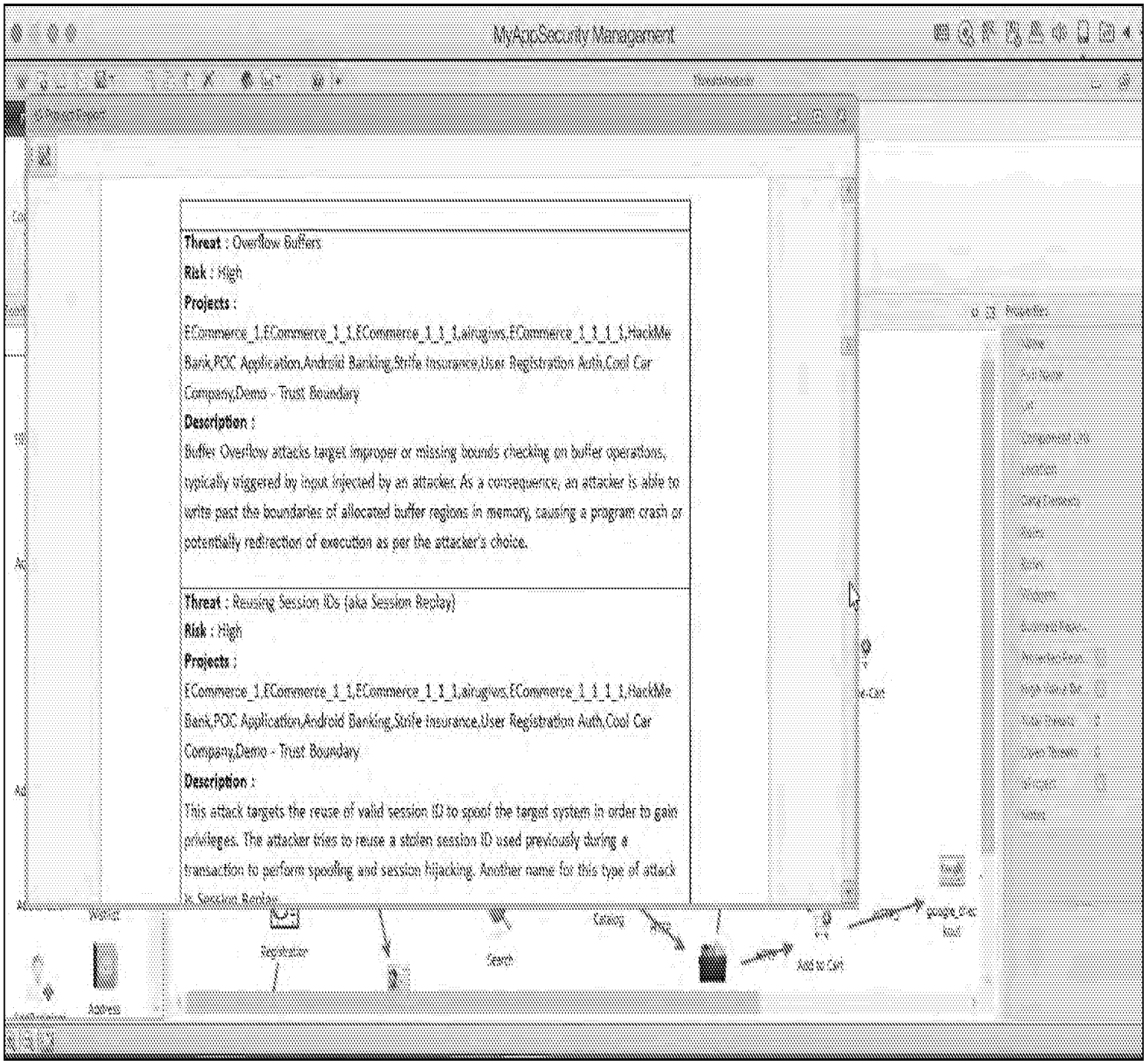

MyAppSecurity Management
MyAppSecurity
SECUREyourApplication
Threat Report
Dated: 02/08/2018

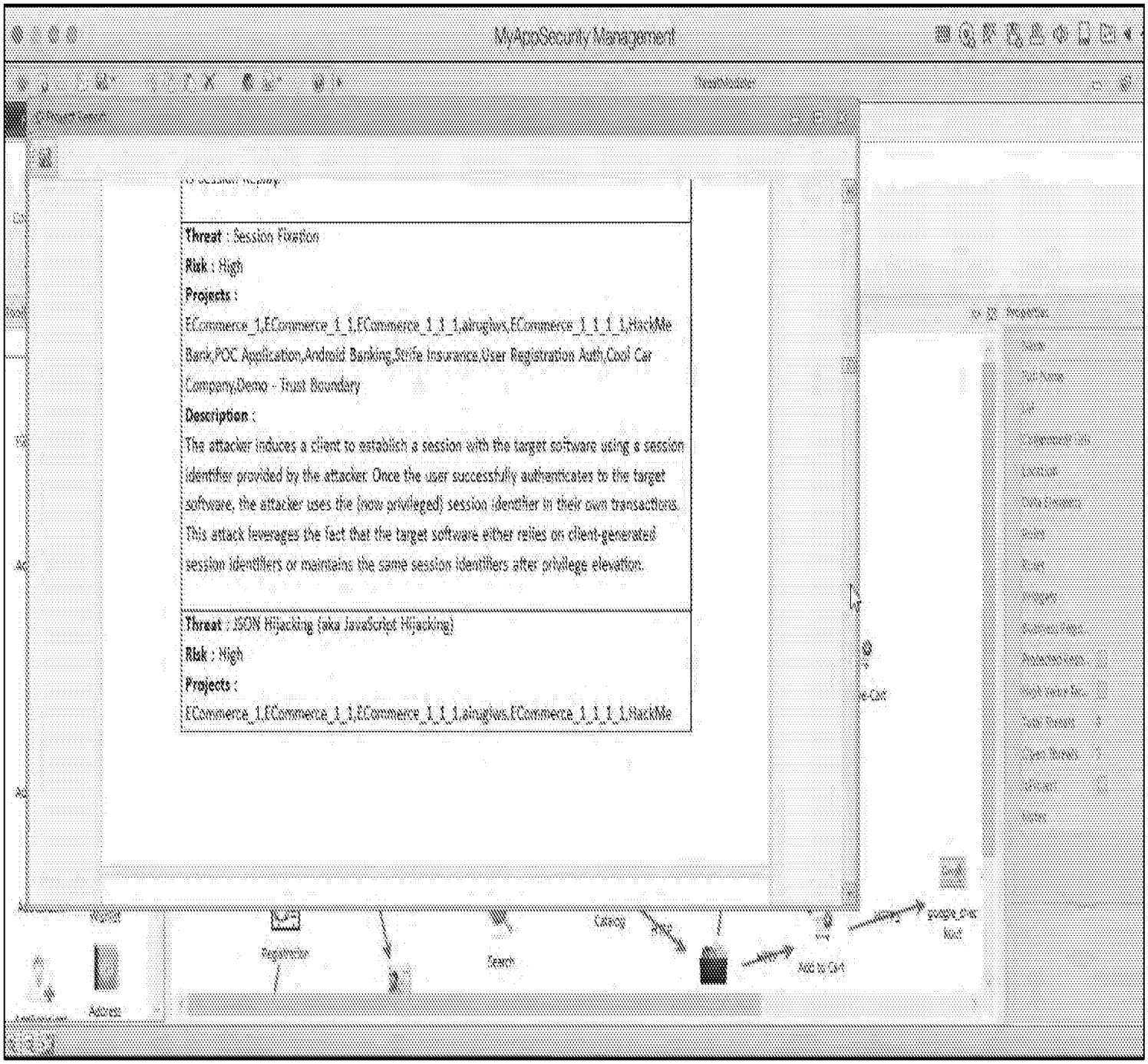

MyAppSecurity Management
Threats found in the Projects:
Threat : Cross Site Request Forgery (aka Session Riding)
Risk : Very High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,aiaugiaw,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
This type of attack leverages the persistence and implicit trust placed inuser session cookies by many web applications today. In such anarchitecture, once the user authenticates to an application and a sessioncookie is created on the user's system, all following transactions for thatsession are authenticated using that cookie including potential actionsinitiated by an attacker and simply "riding" the existing sessioncookie.
Threat : SQL Injection - WASC
Risk : Very High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,aiaugiaw,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Properties
Search
Add to Cart

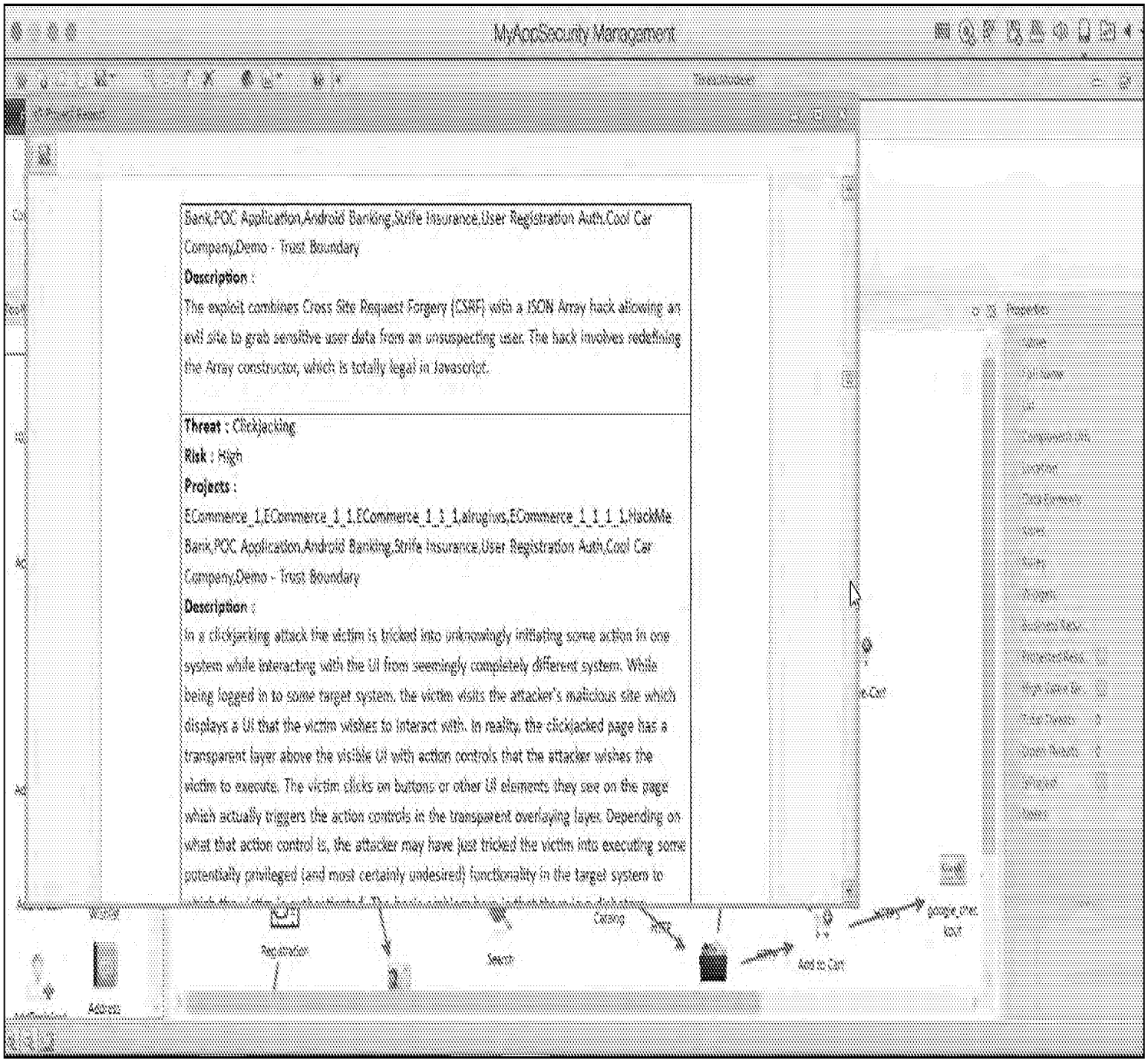

MyAppSecurity Management
Description :
SQL injection results from failure of the application to appropriately validate input. When specially crafted user-controlled input consisting of SQL syntax is used without proper validation as part of SQL queries, it is possible to steal information from the database in ways not envisaged during application design. Depending upon the database and the design of the application, it may also be possible to leverage injection to have the database execute system-related commands of the attacker's choice. SQL injection enables an attacker to talk directly to the database, thus bypassing the application completely. Successful injection can cause information disclosure as well as ability to add or modify data in the database.
Threat : Man in the Middle Attack
Risk : Very High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,aimgiws,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :

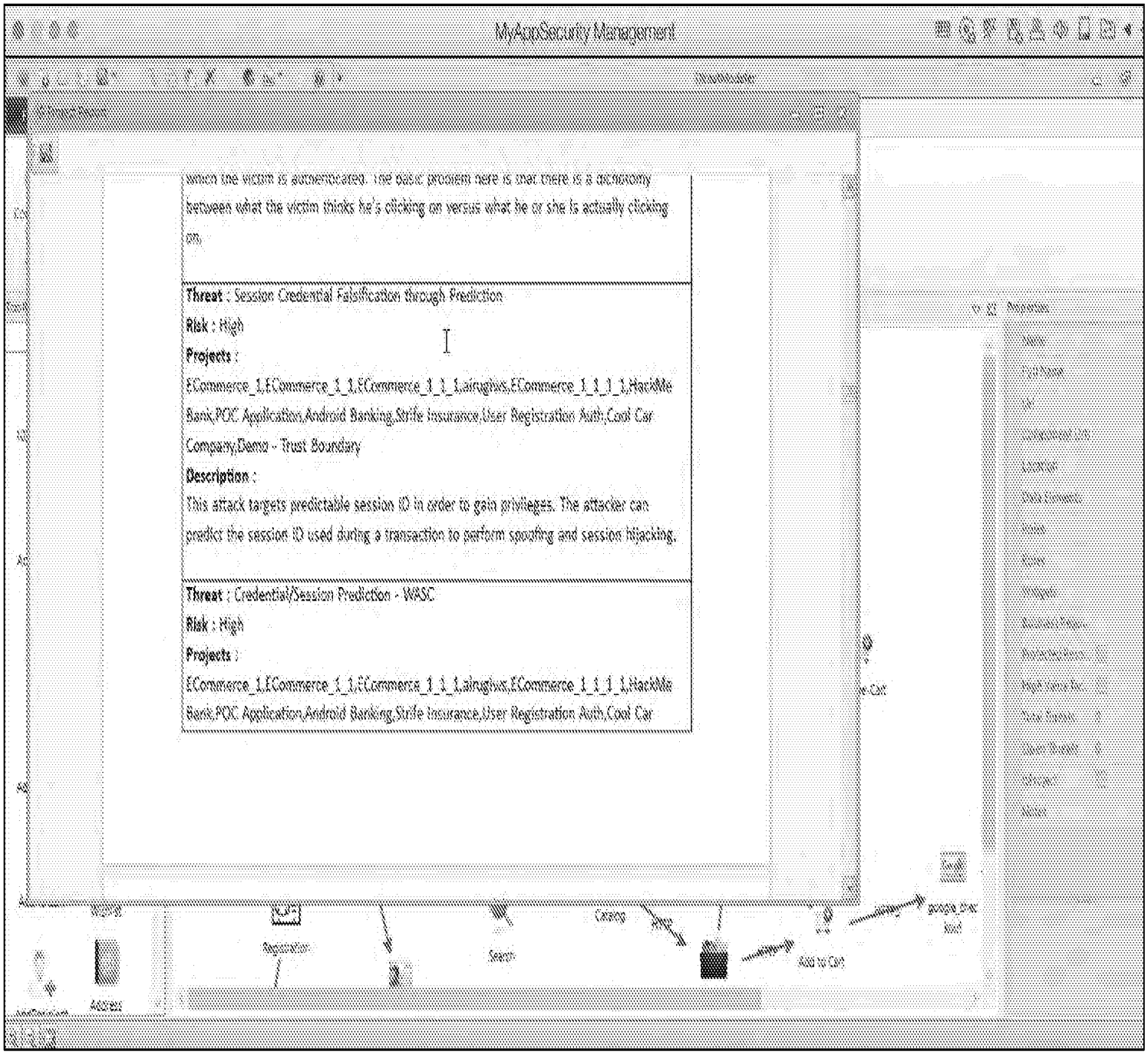

MyAppSecurity Management
This type of attack targets the communication between two components (typically client and server). The attacker places himself in the communication channel between the two components. Whenever one component attempts to communicate with the other (data flow, authentication challenges, etc.), the data first goes to the attacker, who has the opportunity to observe or alter it, and it is then passed on to the other component as if it was never intercepted. This interposition is transparent leaving the two compromised components unaware of the potential corruption or leakeage of their communications. The potential for Man-in-the-Middle attacks yields an implicit lack of trust in communication or identity between two components.
Threat : Cryptanalysis
Risk : Very High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,aiugiwa,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
Cryptanalysis is a process of finding weaknesses in cryptographic algorithms and using these weaknesses to decipher the ciphertext without knowing the secret key (instance deduction). Sometimes the weakness is not in the cryptographic algorithm itself, but rather in how it is applied that makes cryptanalysis successful.An attacker may have other goals as well, such as:1. Total Break - Finding the secret key2. Global Deduction - Finding a functionally equivalent algorithm for encryption and decryption that does not require knowledge of the secret key.3. Information Deduction - Gaining some information about
Properties
Wishlist
Address
Search
Add to Cart
Google_Checkout

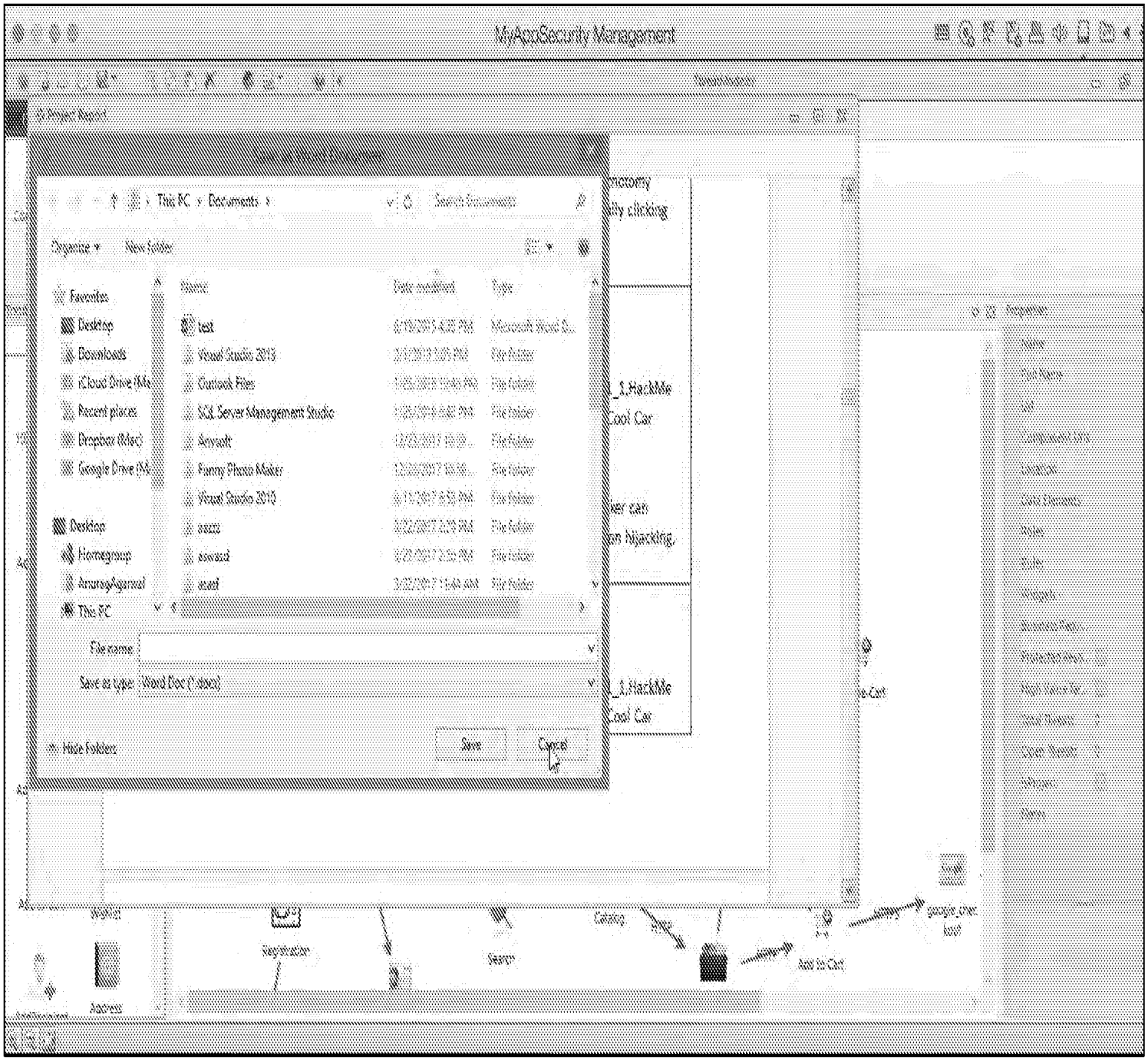

MyAppSecurity Management
plaintexts or ciphertexts that was not previously known. Distinguishing Algorithm - The attacker has the ability to distinguish the output of the encryption (ciphertext) from a random permutation of bits.The goal of the attacker performing cryptanalysis will depend on the specific needs of the attacker in a given attack context. In most cases, if cryptanalysis is successful at all, an attacker will not be able to go past being able to deduce some information about the plaintext (goal 3). However, that may be sufficient for an attacker, depending on the context.
Threat : Persistent session handling cookies
Risk : Very High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,airugiiva,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
When a session handling cookie is set persistently it allows the cookie to be valid even after a user terminates a session. Therefore an attacker can use a session cookie stored in the text file by the browser to access restricted information.
Properties
Name
Add to Cart
Search
Catalog
Add to Cart
Access


MyAppSecurity Management
Threat : Overflow Buffers
Risk : High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,airugiwa,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
Buffer Overflow attacks target improper or missing bounds checking on buffer operations, typically triggered by input injected by an attacker. As a consequence, an attacker is able to write past the boundaries of allocated buffer regions in memory, causing a program crash or potentially redirection of execution as per the attacker's choice.
Threat : Reusing Session IDs (aka Session Replay)
Risk : High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,airugiwa,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
This attack targets the reuse of valid session ID to spoof the target system in order to gain privileges. The attacker tries to reuse a stolen session ID used previously during a transaction to perform spoofing and session hijacking. Another name for this type of attack is Session Replay.
Properties
Customer
Address
Registration
Search
Catalog
Add to Cart


MyAppSecurity Management
Threat : Session Fixation
Risk : High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,alrugiiws,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
The attacker induces a client to establish a session with the target software using a session identifier provided by the attacker. Once the user successfully authenticates to the target software, the attacker uses the (now privileged) session identifier in their own transactions. This attack leverages the fact that the target software either relies on client-generated session identifiers or maintains the same session identifiers after privilege elevation.
Threat : JSON Hijacking (aka JavaScript Hijacking)
Risk : High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,alrugiiws,ECommerce_1_1_1_1,HackMe


MyAppSecurity Management
Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
The exploit combines Cross Site Request Forgery (CSRF) with a JSON Array hack allowing an evil site to grab sensitive user data from an unsuspecting user. The hack involves redefining the Array constructor, which is totally legal in Javascript.
Threat : Clickjacking
Risk : High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,alrugbies,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
In a clickjacking attack the victim is tricked into unknowingly initiating some action in one system while interacting with the UI from seemingly completely different system. While being logged in to some target system, the victim visits the attacker's malicious site which displays a UI that the victim wishes to interact with. In reality, the clickjacked page has a transparent layer above the visible UI with action controls that the attacker wishes the victim to execute. The victim clicks on buttons or other UI elements they see on the page which actually triggers the action controls in the transparent overlaying layer. Depending on what that action control is, the attacker may have just tricked the victim into executing some potentially privileged (and most certainly undesired) functionality in the target system to
Properties
Search
Add to Cart
Address


MyAppSecurity Management
which the victim is authenticated. The basic problem here is that there is a dichotomy between what the victim thinks he's clicking on versus what he or she is actually clicking on.
Threat : Session Credential Falsification through Prediction
Risk : High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,aiinsgiwa,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car Company,Demo - Trust Boundary
Description :
This attack targets predictable session ID in order to gain privileges. The attacker can predict the session ID used during a transaction to perform spoofing and session hijacking.
Threat : Credential/Session Prediction - WASC
Risk : High
Projects :
ECommerce_1,ECommerce_1_1,ECommerce_1_1_1,aiinsgiwa,ECommerce_1_1_1_1,HackMe Bank,POC Application,Android Banking,Strife Insurance,User Registration Auth,Cool Car
Properties
Name
Full Name
ID
Description
Comments
Notes
Status
Category
Business Value
Probability
Risk Score
Risk Level
Threat
Weakness
Registration
Search
Catalog
Add to Cart
Address


MyAppSecurity Management
This PC
Documents
Search Documents
Organize
New folder
Favorites
Desktop
Downloads
Recent places
Dropbox (Mac)
Google Drive (M...
Homegroup
This PC
Name
Date modified
Type
test
Visual Studio 2015
Outlook Files
SQL Server Management Studio
Xamarin
Funny Photo Maker
Visual Studio 2010
File name:
Save as type:
Word Doc (*.docx)
Hide Folders
Save
Cancel
Properties
Add to Cart

Electronic Acknowledgement Receipt

EFS ID:	36892447
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Correspondence Address:	Law Office of Paul B. Johnson - 8483 S. 1275 E. - Sandy UT 84094-1363 US 801-719-5229 paul@pauljohnsonlaw.com
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	15-AUG-2019
Filing Date:	
Time Stamp:	22:34:18
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Information Disclosure Statement (IDS) Form (SB08)	IDS1.pdf	1035766 b24aee0f0ebb84af938a73b7ef2d0e98805315c8	no	6
Warnings:					
Information:					
2	Non Patent Literature	NPL01.pdf	328259 571e01737a3f1ae312f9da4f512dbda505ece66f	no	12
Warnings:					
Information:					
3	Non Patent Literature	NPL02.pdf	334884 566bd40ca5886bf7002becae8b261a6d89e5a2e8	no	6
Warnings:					
Information:					
4	Non Patent Literature	NPL03.pdf	166989 4bf1903d6f4f870e231d3edac4d5e7fd21c5e13b	no	5
Warnings:					
Information:					
5	Non Patent Literature	NPL04.pdf	1592605 7973b868f296919547c7e9d1c6a950614225c253	no	190
Warnings:					
Information:					
6	Non Patent Literature	NPL06.pdf	967495 d91c689b892a8dd53787024c0b24d12494ce0c66	no	12
Warnings:					
Information:					
7	Non Patent Literature	NPL05.pdf	1012702 a03a3fa6a4bde973b8086dc4fb94b5dab14a8974	no	25
Warnings:					
Information:					

Total Files Size (in bytes):	5438700

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

CD Submission Review Results

Internal Worksheet Only - Do Not Scan

Application Number	16542263	Receipt Date	Aug 15, 2019

X	New Application		Response To Notice	Date Of Response

[ ] Jumbo – Large tables are included on the CD

[X] **There are no problems with the CDs in the submission.**

Items checked below are to be selected in the PALM Pre-Exam CD problem list.

Omitted Items

	Files on compact disc are unreadable:
	Disc(s) of either set not readable:
	Disc(s) do not contain file names listed:
	Amended CD missing unamended files
	Missing compact discs

Missing Parts

	Disc(s) do not contain file names listed
	Discs contains files that are non-ASCII files:
	Only one copy of the CD, no duplicate:
	Wrong media type, e.g., CD-RW:
	The transmittal letter and specification do not list information:
	Both disc(s) contain unreadable files:
	One disc(s) contain unreadable files:
	Compact disc(s) contain viruses, but are still readable:
	EFS/Paper Submission:
	Table does not maintain data in proper alignment:
	Not proper subject matter for CD's:

Informality

	No statement in transmittal letter that CD's are the same:
	No incorporation by reference statement for the CD's:
	CD's not labeled "Copy 1" and "Copy 2":
	Deficient CD packaging:
	Deficient Labeling on CD's:
	The transmittal letter does not list information:
	The specification does not list information:
	Table less than 51 pages submitted on CD:

File Directory Attachment

	Copy of file directory listing is attached and should be mailed to applicant
	Copy of file directory listing is not attached
	Copy of file directory listing is not attached because it is over one page long

Additional Comments

SCORE Placeholder Sheet for IFW Content

Application Number: 16542263 Document Date: 08/15/2019

The presence of this form in the IFW record indicates that the following document type was received in electronic format on the date identified above. This content is stored in the SCORE database.

Since this was an electronic submission, there is no physical artifact folder, no artifact folder is recorded in PALM, and no paper documents or physical media exist. The TIFF images in the IFW record were created from the original documents that are stored in SCORE.

- Drawing

At the time of document entry (noted above):

- USPTO employees may access SCORE content via DAV or via the SCORE web page.
- External customers may access SCORE content via PAIR using the Supplemental Content tab.

Form Revision Date: March 1, 2019

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	TOT CLAIMS	IND CLAIMS
16/542,263	08/15/2019		785	ThreatM010	20	3

CONFIRMATION NO. 7756

FILING RECEIPT

145042
Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

OC000000110677405

Date Mailed: 08/28/2019

Receipt is acknowledged of this non-provisional utility patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF FIRST INVENTOR, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection.

Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a corrected Filing Receipt, including a properly marked-up ADS showing the changes with strike-through for deletions and underlining for additions. If you received a "Notice to File Missing Parts" or other Notice requiring a response for this application, please submit any request for correction to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections provided that the request is grantable.

Inventor(s)

Anurag Agarwal, Jersey City, NJ;

Applicant(s)

ThreatModeler Software Inc., Jersey City, NJ;

Assignment For Published Patent Application

ThreatModeler Software Inc., Jersey City, NJ

Power of Attorney: The patent practitioners associated with Customer Number 145042

Domestic Priority data as claimed by applicant

This application is a CIP of 16/228,738 12/20/2018
which is a CIP of 15/922,856 03/15/2018 PAT 10200399
which is a CIP of 15/888,021 02/03/2018 PAT 10255439
which claims benefit of 62/507,691 05/17/2017
and claims benefit of 62/520,954 06/16/2017
and claims benefit of 62/527,671 06/30/2017
and claims benefit of 62/530,295 07/10/2017

Foreign Applications for which priority is claimed (You may be eligible to benefit from the **Patent Prosecution Highway** program at the USPTO. Please see http://www.uspto.gov for more information.) - None.
Foreign application information must be provided in an Application Data Sheet in order to constitute a claim to foreign priority. See 37 CFR 1.55 and 1.76.

IriusRisk
Ex. 1002 - Page 377

Permission to Access Application via Priority Document Exchange: Yes

Permission to Access Search Results: Yes

Applicant may provide or rescind an authorization for access using Form PTO/SB/39 or Form PTO/SB/69 as appropriate.

If Required, Foreign Filing License Granted: 08/26/2019

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US 16/542,263**

Projected Publication Date: 12/05/2019

Non-Publication Request: No

Early Publication Request: No
**** SMALL ENTITY ****
Title

Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

Preliminary Class

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications: No

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

IriusRisk
Ex. 1002 - Page 378

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4258).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

SelectUSA

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The U.S. offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to promote and facilitate business investment. SelectUSA provides information assistance to the international investor

IriusRisk
Ex. 1002 - Page 379

community; serves as an ombudsman for existing and potential investors; advocates on behalf of U.S. cities, states, and regions competing for global investment; and counsels U.S. economic development organizations on investment attraction best practices. To learn more about why the United States is the best country in the world to develop technology, manufacture products, deliver services, and grow your business, visit http://www.SelectUSA.gov or call +1-202-482-6800.

IriusRisk
Ex. 1002 - Page 380

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 16/542,263

APPLICATION AS FILED - PART I

	(Column 1)	(Column 2)	SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE($)	FEE($)		RATE($)	FEE($)
BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A	75		N/A	
SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A	330		N/A	
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A	380		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	20 minus 20 =	*	x 50 =	0.00	OR		
INDEPENDENT CLAIMS (37 CFR 1.16(h))	3 minus 3 =	*	x 230 =	0.00			
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).			0.00			
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))				0.00			
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL	785		TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)	SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE($)	ADDITIONAL FEE($)		RATE($)	ADDITIONAL FEE($)
AMENDMENT A	Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

		(Column 1)		(Column 2)	(Column 3)					
		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE($)	ADDITIONAL FEE($)		RATE($)	ADDITIONAL FEE($)
AMENDMENT B	Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".
The "Highest Number Previously Paid For" (Total or Independent) is the highest found in the appropriate box in column 1.

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Aug 28, 2019 03:32:12 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	APP.FILE.REC	08/28/2019	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Aug 28, 2019 03:32:12 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	APP.FILE.REC	08/28/2019	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2491
Examiner Name	Patel, Ashokkumar B.
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	9043924	B2	2015-05-26	Maor et al.	
	2	8413249	B1	2013-04-02	Chou	
	3	8407801	B2	2013-03-26	Ikegami et al.	
	4	6952779	B1	2005-10-04	Cohen et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16542263
	Filing Date	2019-08-15
	First Named Inventor	Anurag Agarwal
	Art Unit	2491
	Examiner Name	Patel, Ashokkumar B.
	Attorney Docket Number	ThreatM010

	1						

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1		

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2491
Examiner Name	Patel, Ashokkumar B.
Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2019-09-05
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Acknowledgement Receipt

EFS ID:	37082933
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	06-SEP-2019
Filing Date:	15-AUG-2019
Time Stamp:	00:08:46
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Information Disclosure Statement (IDS) Form (SB08)	IDS2.pdf	1034378 2a8034562135e14597e76c88b333d73434a0960c	no	4

Warnings:

IriusRisk

Information:	
Total Files Size (in bytes):	1034378

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Electronic Acknowledgement Receipt

EFS ID:	37082961
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	06-SEP-2019
Filing Date:	15-AUG-2019
Time Stamp:	00:53:22
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Preliminary Amendment	PreliminaryAmendment.pdf	116166 7dae2e0d1602992a2124522bcd53288f43708138	no	4

Warnings:

IriusRisk

Information:	
Total Files Size (in bytes):	116166

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

CONCLUSION

In conclusion, it is respectfully requested that the application be allowed. If it will in any way speed examination and/or allowance, the Examiner is invited to contact an attorney of record using the contact information listed below.

Respectfully submitted,

Date: September 5, 2019 By: /Paul B. Johnson/

Paul B. Johnson
Reg. No. 59375

Law Office of Paul B. Johnson
8483 S 1275 E
Sandy, UT 84094-1363
Ph: 801.719.5229
Fax: 801.317.8889
paul@pauljohnsonlaw.com

IriusRisk
Ex. 1002 - Page 392

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

AMENDMENT/REPLY

Applicant: ThreatModeler Software Inc.

Docket No: THREATM010

Examiner: Patel, Ashokkumar B.

Ser. No: 16542263

Filed: August 15, 2019

Group Art Unit: 2491

Title: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

PRELIMINARY AMENDMENT UNDER 37 C.F.R. 1.115

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Examiner:

This is a preliminary amendment.

Specification Amendments begin on page 2.

Remarks/Arguments begin on page 4.

IriusRisk
Ex. 1002 - Page 393

AMENDMENTS TO THE SPECIFICATION

Please make the following amendments to the specification, which amendments

Please replace paragraph [0179] with the following paragraph:

[0179] Interface 2400 is seen to list threat models by name and to include a version number (1.1, 1.0, etc.). These may be all threat models that have been created by a specific organization or end user of the system and methods. Also listed are a percentage (indicating a completion percentage of the threat model), an overall threat risk of the model (Medium, Very High, etc.) as assigned by a user using interface 2500 (described later), who created or last worked on the model, and a trash icon for deleting the model. A filter selector at the top left allows a user to apply various filters to search for a specific threat model. On the right are shown THREAT TRACEABILITY MATRIX and TASKS areas which show further information and which may operate as selectors to navigate to further information. The user is seen to have selected the END POINT SEC. threat model, so that it is highlighted. Additional selectors include a new selector (selector) 2402 which may be used to begin a new threat model, an edit selector which may be used to edit a threat model or the aforementioned descriptive information, a number indicator (48) indicating the total number of threat models in the one or more databases (or the total number responsive to the filter request), and other selectors. In implementations a user could double-click on any given threat model to navigate to another interface, such as the diagram interface of FIG. 21 or the threat report interface of FIG. 13.

Below please find the full text of the replacement paragraph(s) with markings to show all the changes relative to the previous versions:

[0179] Interface 2400 is seen to list threat models by name and to include a version number (1.1, 1.0, etc.). These may be all threat models that have been created by a specific organization or end user of the system and methods. Also listed are a percentage (indicating

IriusRisk
Ex. 1002 - Page 394

a completion percentage of the threat model), an overall threat risk of the model (Medium, Very High, etc.) as assigned by a user using interface 2500 (described later), who created or last worked on the model, and a trash icon for deleting the model. A filter selector at the top left allows a user to apply various filters to search for a specific threat model. On the right are shown THREAT TRACEABILITY MATRIX and TASKS areas which show further information and which may operate as selectors to navigate to further information. The user is seen to have selected the END POINT SEC. threat model, so that it is highlighted. Additional selectors include a new selector (selector) 2402 which may be used to begin a new threat model, an edit selector which may be used to edit a threat model or the aforementioned descriptive information, a number indicator (48) indicating the total number of threat models in the one or more databases (or the total number ~~response~~ responsive to the filter request), and other selectors. In implementations a user could double-click on any given threat model to navigate to another interface, such as the diagram interface of FIG. 21 or the threat report interface of FIG. 13.

IriusRisk
Ex. 1002 - Page 395

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/542,263	08/15/2019	Anurag Agarwal	ThreatM010	7756

145042 7590 09/19/2019

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

EXAMINER
PATEL, ASHOKKUMAR B

ART UNIT	PAPER NUMBER
2491	

NOTIFICATION DATE	DELIVERY MODE
09/19/2019	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

paul@pauljohnsonlaw.com

IriusRisk
Ex. 1002 - Page 396

Decision Granting Request for Prioritized Examination (Track I)	**Application No.** 16/542,263	**Applicant(s)** Agarwal, Anurag	
	Examiner CHERYL P GIBSON BAYLOR	**Art Unit** OPET	**AIA (FITF) Status** Yes

1. THE REQUEST FILED 15 August 2019 IS **GRANTED** .

The above-identified application has met the requirements for prioritized examination

A. ☑ for an original nonprovisional application (Track I).
B. ☐ for an application undergoing continued examination (RCE).

2. **The above-identified application will undergo prioritized examination**. The application will be accorded special status throughout its entire course of prosecution until one of the following occurs:

A. filing a **petition for extension of time** to extend the time period for filing a reply;

B. filing an **amendment to amend the application to contain more than four independent claims, more than thirty total claims** , or a multiple dependent claim;

C. filing a **request for continued examination** ;

D. filing a notice of appeal;

E. filing a request for suspension of action;

F. mailing of a notice of allowance;

G. mailing of a final Office action;

H. completion of examination as defined in 37 CFR 41.102; or

I. abandonment of the application.

Telephone inquiries with regard to this decision should be directed to CHERYL GIBSON BAYLOR at (571)272-3213. In his/her absence, calls may be directed to Petition Help Desk at (571) 272-3282.

/CHERYL GIBSON BAYLOR/ Paralegal Specialist, OPET	

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Sep 20, 2019 03:34:36 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	TRACK1.GRANT	09/19/2019	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

PATENT ASSIGNMENT COVER SHEET

Electronic Version v1.1
Stylesheet Version v1.2

EPAS ID: PAT5789632

SUBMISSION TYPE:	NEW ASSIGNMENT
NATURE OF CONVEYANCE:	ASSIGNMENT

CONVEYING PARTY DATA

Name	Execution Date
ANURAG AGARWAL	10/23/2019

RECEIVING PARTY DATA

Name:	THREATMODELER SOFTWARE INC.
Street Address:	101 HUDSON ST., STE. 2100
City:	JERSEY CITY
State/Country:	NEW JERSEY
Postal Code:	07302

PROPERTY NUMBERS Total: 2

Property Type	Number
Application Number:	16542263
Application Number:	16664679

CORRESPONDENCE DATA

Fax Number: (801)317-8889

Correspondence will be sent to the e-mail address first; if that is unsuccessful, it will be sent using a fax number, if provided; if that is unsuccessful, it will be sent via US Mail.

Phone: 8017195229
Email: paul@pauljohnsonlaw.com
Correspondent Name: LAW OFFICE OF PAUL B. JOHNSON
Address Line 1: 8483 S 1275 E
Address Line 4: SANDY, UTAH 84094

ATTORNEY DOCKET NUMBER:	THREATM010
NAME OF SUBMITTER:	PAUL B. JOHNSON
SIGNATURE:	/Paul B. Johnson/
DATE SIGNED:	10/25/2019
	This document serves as an Oath/Declaration (37 CFR 1.63).

Total Attachments: 4
source=PATAssign_Signed#page1.tif
source=PATAssign_Signed#page2.tif
source=PATAssign_Signed#page3.tif
source=PATAssign_Signed#page4.tif

PATENT ASSIGNMENT

This Assignment is made and executed by Anurag Agarwal having an address of 105 Greene St. Unit 1801, Jersey City, NJ 07302 (hereinafter "Assignor"), to and in favor of ThreatModeler Software Inc., a New Jersey Corporation having an address of 101 Hudson St., Ste. 2100, Jersey City, NJ 07302 (hereinafter "Assignee").

Whereas Assignor desires to assign his entire right, title and interest in and to the invention(s) and subject matter described in the United States nonprovisional utility patent application titled "Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files" which was filed on Aug. 15, 2019 and assigned Application Serial No. 16/542,263, and the United States nonprovisional utility patent application titled "Systems and Methods for Assisted Model Generation" which was filed on Oct. 25, 2019 and assigned Application Serial No. 16/664,679 (hereinafter collectively the "Patent Applications") and all patents, patent applications and the like on the invention(s) or the subject matter of the Patent Applications to Assignee for the territory of the United States of America (hereinafter "U.S.") and all foreign countries, for valuable consideration, and whereas Assignee has provided such good and valuable consideration to Assignor, the receipt and sufficiency of which Assignor hereby acknowledges; accordingly, Assignor hereby warrants, covenants, and agrees as follows:

1. Assignor hereby sells, assigns, conveys, transfers, and sets over to Assignee, its successors, legal representatives and assigns his/her/its entire right, title and interest in and to: (a) the Patent Applications themselves; (b) every U.S. patent that has issued or that issues and that is based in whole or in part on one or more of the Patent Applications or the subject matter thereof, including every non-provisional, divisional, continuation, continuation-in-part, reissue, reexamination, substitution, extension, counterpart and the like; (c) every U.S. patent application, already filed or to be filed, that is based in whole or in part on one or more of the Patent Applications or the subject matter thereof, including every counterpart, divisional, continuation, continuation-in-part, provisional, non-provisional, reissue, reexamination, extension, substitution and the like; (d) every non-U.S. patent, inventor's certificate, utility model, design and every like property or right that has issued or that issues and that is based in whole or in part on one or more of the Patent Applications or the subject matter thereof, including every provisional, non-provisional, divisional, continuation, continuation-in-part, substitution and the like; (e) every non-U.S. application for a patent, inventor's certificate, utility model, design or like property or right, filed or to be filed, that is based in whole or in part on one or more of the Patent Applications or the subject matter thereof, including every divisional, continuation, continuation-in-part, non-provisional, provisional, substitution and the like, and; (f) all priority rights, Convention rights and like benefits in the U.S. and every foreign country related to one or more of the Patent Applications, the subject matter thereof, or any application or issued or granted property or right described in this section 1, including but not limited to any and all rights under the Paris Convention, the Patent Cooperation Treaty and any other international agreements to which the U.S. adheres.

IriusRisk
Ex. 1002 - Page 400

2. Assignor agrees that Assignee may apply for patents, utility models, designs, inventor's certificates and the like, in the U.S. and in all foreign countries, based in whole or in part on the invention(s) or the subject matter described in one or more of the Patent Applications or any other application or issued or granted property or right described in section 1 above. Assignor hereby grants to Assignee the sole and exclusive right to prosecute the Patent Applications and every application described in section 1 above as well as any and all opposition proceedings, reissue applications, reexaminations, supplemental examinations, nullity proceedings, invalidity proceedings, pre-grant opposition proceedings, and the like. Assignor authorizes all U.S. and foreign governmental bodies, including the U.S. Patent and Trademark Office (hereinafter "USPTO"), having the power and/or authority to issue patents, inventor's certificates, designs, utility models, or the like based in whole or in part on one or more of the Patent Applications or the subject matter thereof, or any application or issued or granted property or right described in section 1 above, to issue such in the name of Assignee or, as the case may be, Assignee's successors, legal representatives or assigns.

3. Assignor grants to Assignee, its successors, legal representatives, and assigns, the exclusive right to sue third parties for present and past damages related to the invention(s) and the subject matter described in the Patent Applications and any application or issued or granted registration or like property or right described in section 1 above.

4. Assignor warrants and covenants that he/she/it has the full and unencumbered right to sell, assign, convey, transfer and set over the interests hereby sold, assigned, conveyed, transferred and set over, and that he/she/it has not executed and will not execute any document or instrument in conflict with this Assignment.

5. Assignor further covenants and agrees that, upon request and without further consideration, but at the expense of Assignee, he/she/it will: (a) assist Assignee in prosecuting the Patent Applications and each application described in section 1 above as well as opposition proceedings, reissue applications, reexaminations, supplemental examinations, nullity proceedings, invalidity proceedings, pre-grant opposition proceedings and the like; (b) provide Assignee with information concerning the invention(s) and the subject matter described in the Patent Applications and each application and issued or granted property or right described in section 1 above; (c) execute, sign and deliver papers, make all rightful oaths and testify in any legal proceedings related to one or more of the Patent Applications or any application or issued or granted property or right described in section 1 above; (d) execute additional assignment documents in favor of Assignee, similar to this Assignment, for each application and each issued or granted property or right described in section 1 above, and; (e) perform all other reasonably necessary acts; to enable Assignee to obtain, secure, maintain, perfect, protect and enforce all of the rights, title and interest sold, assigned, conveyed, transferred and set over under this Assignment and to file and issue the Patent Applications and each application and issued or granted property or right described in section 1 above in the name of Assignee as applicant and owner.

IriusRisk
Ex. 1002 - Page 401

6. Assignee may assign any right, title and interest sold, assigned, conveyed, transferred and set over under this Assignment, in whole or in part, to any party.

7. All references herein to Assignee include any successors, legal representatives or assigns of Assignee.

8. The above-identified Patent Applications were made or authorized to be made by Assignor. Assignor believes that he/she is the original inventor or an original joint inventor of a claimed invention in the Patent Applications. Assignor acknowledges that any willful false statement made in this paragraph is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five (5) years, or both.

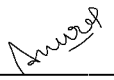
9. Assignor hereby authorizes and requests any attorney of the Law Office of Paul B. Johnson, Customer number 145042, to insert, after execution of this Assignment, the above filing date and application number of each of the Patent Applications and any further information necessary or desirable for purposes of recording this Assignment with the USPTO.

10. This Assignment shall be binding upon any and all successors, legal representatives, and assigns of Assignor.

IriusRisk
Ex. 1002 - Page 402

Executed this 23 day of October, 2019 at Jersey City, NJ by:
(City and State)

Legal Name: Anurag Agarwal
Print or Type

Signature: 

Mailing Address: 105 Greene St. Unit 1801

Jersey City, NJ 07302

Residence: ☑ Same as mailing address, or:

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/542,263	08/15/2019	Anurag Agarwal	ThreatM010	7756

145042 7590 11/18/2019

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

EXAMINER
PARSONS, THEODORE C

ART UNIT	PAPER NUMBER
2494	

NOTIFICATION DATE	DELIVERY MODE
11/18/2019	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

paul@pauljohnsonlaw.com

Office Action Summary	Application No. 16/542,263	Applicant(s) Agarwal, Anurag	
	Examiner THEODORE C PARSONS	Art Unit 2494	AIA (FITF) Status Yes

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTHS FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☑ Responsive to communication(s) filed on 2019-10-25.

☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2a)☐ This action is **FINAL.** 2b) ☑ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims*

5) ☑ Claim(s) 1-20 is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6) ☐ Claim(s) _____ is/are allowed.

7) ☑ Claim(s) 1-20 is/are rejected.

8) ☐ Claim(s) _____ is/are objected to.

9) ☐ Claim(s) _____ are subject to restriction and/or election requirement

* If any claims have been determined allowable, you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to **PPHfeedback@uspto.gov.**

Application Papers

10)☐ The specification is objected to by the Examiner.

11)☑ The drawing(s) filed on 2019-08-15 is/are: a)☐ accepted or b)☑ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

a)☐ All b)☐ Some** c)☐ None of the:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

** See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☑ Notice of References Cited (PTO-892)

2) ☑ Information Disclosure Statement(s) (PTO/SB/08a and/or PTO/SB/08b) Paper No(s)/Mail Date 2019-08-15, 2019-09-06.

3) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

4) ☐ Other: _____.

DETAILED ACTION

1. The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

2. This is in reply to papers filed on 2019-10-25. Claims 1-20 are pending. Claims 1, 8, 16 is/are independent.

Information Disclosure Statement PTO-1449

3. The Information Disclosure Statement(s) submitted by applicant on 2019-08-15, 2019-09-06 has/have been considered. The submission is in compliance with the provisions of 37 CFR § 1.97. Form PTO-1449 signed and attached hereto.

Specification

4. The use of the term "VISIO", which is a trade name or a mark used in commerce, has been noted in this application. It should be capitalized wherever it appears and be accompanied by the generic terminology.

5. Although the use of trade names and marks used in commerce (*i.e.*, trademarks, service marks, certification marks, and collective marks) are permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner which might adversely affect their validity as commercial marks.

Drawings

6. The drawings are objected to because the unlabeled box(es) and shapes shown in Figure(s) 1 should be provided with descriptive text labels sufficient to orient the viewer to what is being depicted. Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Rejections - 35 U.S.C. § 112

7. The following is a quotation of 35 U.S.C. § 112(b):

> (b) CONCLUSION.—The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the inventor or a joint inventor regards as the invention.

8. **Claim(s) 1-20** is/are **rejected** under 35 U.S.C. § 112(b) or 35 U.S.C. § 112 ¶ 2 (pre-AIA) as being indefinite for failing to particularly point out and distinctly claim the subject matter which the inventor or a joint inventor, or for pre-AIA the applicant regards as the invention.

Claim(s) 1, 8, 16 are incomplete for omitting essential structural cooperative relationships of elements, such omission amounting to a gap between the necessary structural connections. See MPEP § 2172.01. In particular, claim 1 recites "using visual representations of the threat model components correlated with the subset of third party diagram components". As best understood, the "third party diagram components" are visual representations. Accordingly, a person of ordinary skill in the art would not be able to determine how the claimed "visual representations" relate to the "third party diagram components". Does Applicant intend to claim using members of the subset of third party diagram components to visually represent respective correlated threat model components? Or, are the "visual representations" and the "third party diagram components" intended to be separate entities, as the current claim language states? Claims 8 and 16 suffer the same defects, *mutatis mutandis.*

Claim 16 contains the trademark/trade name "VISIO". Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph. *See* Ex parte Simpson, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods

associated with the trademark or trade name. In the present case, the trademark/trade name is used to identify/describe particular diagram components and, accordingly, the identification/description is indefinite.

Claim 16 contains the trademark/trade name "VISIO". Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112(b) or pre-AIA 35 U.S.C. 112, second paragraph. *See* Ex parte Simpson, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. In the present case, the trademark/trade name is used to identify/describe particular diagram components and, accordingly, the identification/description is indefinite.

Claims 1, 5, 6, 8, 12-14, 19-20 contain the term "third party diagram components". However, "third party" refers to an issue of ownership, which can change at arbitrary times. That is, a diagram component owned by a third party can be bought, ceasing to infringe; similarly, a diagram component owned by a first party can be sold, creating infringement where none existed before the sale. Thus, the scope of the claims is uncertain. Further, whether a diagram component is owned by a third party has no connection to the structure or characteristics of the diagram component, making it impossible for a person of ordinary skill in the art to determine, from its features and characteristics, whether a diagram component would fall within the scope of the claim or not. To the extent that Applicant desires to claim

diagram components of a separate and independent pre-existing diagram format, Applicant should consult with Applicant's counsel to determine how best describe the scope Applicant desires to claim.

In claim 19-20, the phrase "the third party diagram components" makes the claims indefinite and unclear in that it lacks antecedent basis.

Dependent claims 2-7, 9-15, 17-20 are rejected for the reasons presented above with respect to rejected claims 1, 8, 16 and in view of their dependence thereon.

Conclusion

9. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

- U.S. Publication 20150033346 to Hebert et al. (hereinafter "Herbert '346") discloses using an architecture model to generate a threat model by mapping threats to components of the model [Herbert '346 ¶ 0031]; storing mappings in a database [Herbert '346 ¶ 0031]; manual user inputs to refine mappings [Herbert '346 ¶ 0032]; and generating a visual diagram of the threat model [Herbert '346 ¶ 0087-0088, 0097-0100, Fig. 5, Fig. 8]. However, Herbert '346 does not disclose third-party ownership/control of the diagram components of the architecture model.

- U.S. Patent 8413249 to Chou (hereinafter "Chou '249") discloses using an architecture model to generate a threat model [Chou '249 col. 6, l. 3-29; Fig. 2C] in which architecture components and relationships are correlated to known vulnerabilities [Chou '249 col. 7, l. 1-44]. However, Chou '249 does not disclose third-party ownership/control of the diagram components of the architecture model.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to THEODORE C PARSONS whose telephone number is (571)270-1475. The examiner can normally be reached on MTWRF 7:30-4:30.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is

encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jung Kim can be reached on (571) 272-3804. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/THEODORE C PARSONS/
Primary Examiner, Art Unit 2494

Notice of References Cited	Application/Control No. 16/542,263	Applicant(s)/Patent Under Reexamination Agarwal, Anurag	
	Examiner THEODORE C PARSONS	Art Unit 2494	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-8413249-B1	04-2013	Chou; Andy C	G06F11/00	726/25
*	B	US-20140157417-A1	06-2014	Grubel; Brian Christopher	G06F21/577	726/25
*	C	US-20110277034-A1	11-2011	Hanson; Jim	G06F21/554	726/25
*	D	US-20160248798-A1	08-2016	Cabrera; Luis Felipe	G06F21/577	1/1
*	E	US-20150033346-A1	01-2015	Hebert; Cedric	G06F21/577	726/25
	F					
	G					
	H					
	I					
	J					
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	Bruce Potter, Microsoft SDL Threat Modelling Tool, Network Security, Volume 2009, Issue 1, Pages 15-18 (Elsevier 2009) (Year: 2009)
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	Art Unit
	THEODORE C PARSONS	2494

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

CLAIMS

☑ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIM		DATE								
Final	Original	10/28/2019								
	1	✓								
	2	✓								
	3	✓								
	4	✓								
	5	✓								
	6	✓								
	7	✓								
	8	✓								
	9	✓								
	10	✓								
	11	✓								
	12	✓								
	13	✓								
	14	✓								
	15	✓								
	16	✓								
	17	✓								
	18	✓								
	19	✓								
	20	✓								

IriusRisk
Ex. 1002 - Page 413

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	Art Unit
	THEODORE C PARSONS	2494

CPC - Searched*		
Symbol	Date	Examiner
G06F21/577; G06F16/2282; G06F16/248; G06F9/451; G06F16/212; G06F2221/033	2019-11-12	TCP

CPC Combination Sets - Searched*		
Symbol	Date	Examiner

US Classification - Searched*			
Class	Subclass	Date	Examiner

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

Search Notes		
Search Notes	Date	Examiner
inventor, assignee search	2019-10-28	TCP
EAST (see notes attached)	2019-10-28	TCP
Google Patents Sigma (see notes attached)	2019-10-28	TCP
IP.com InnovationQ (see notes attached)	2019-10-28	TCP
Google Scholar (see notes attached)	2019-11-12	TCP
inventor, assignee search	2019-11-12	TCP
EAST (see notes attached)	2019-11-12	TCP
foreign counterpart search	2019-11-12	TCP

/THEODORE C PARSONS/ Primary Examiner, Art Unit 2494	

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	Art Unit
	THEODORE C PARSONS	2494

Interference Search			
US Class/CPC Symbol	US Subclass/CPC Group	Date	Examiner
	PGPUB text search via EAST (see interference notes attached)	2019-11-12	TCP

/THEODORE C PARSONS/ Primary Examiner, Art Unit 2494	

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	
First Named Inventor	Anurag Agarwal
Art Unit	
Examiner Name	
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	7891003	B2	2011-02-15	Mir	
	2	7900259	B2	2011-03-01	Jeschke et al.	
	3	9774613	B2	2017-09-26	Thomas et al.	
	4	4959015		1990-09-25	Rasinski et al.	
	5	7433829	B2	2008-10-07	Borgia et al.	
	6	7096502	B1	2006-08-22	Fox et al.	
	7	8191139	B2	2012-05-29	Heimerdinger et al.	
	8	8255995	B2	2012-08-28	Kraemer et al.	

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Anurag Agarwal
	Art Unit	
	Examiner Name	
	Attorney Docket Number	ThreatM010

	9	8413237	B2	2013-04-02	O'Rourke et al.	
	10	9497203	B2	2016-11-15	Honig et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20180324207	A1	2018-11-08	Reybok, Jr. et al.	
	2	20060015941	A1	2006-01-19	McKenna	
	3	20060241991	A1	2006-10-26	Pudhukottai et al.	
	4	20110178942	A1	2011-06-21	Watters et al.	
	5	20140137257	A1	2014-05-15	Martinez et al.	
	6	20110126111	A1	2011-05-26	Gill et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	
First Named Inventor	Anurag Agarwal
Art Unit	
Examiner Name	
Attorney Docket Number	ThreatM010

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	Threat Risk Modeling, published online at least as early as Feb. 3, 2018 by OWASP, available at https://www.owasp.org/index.php/Threat_Risk_Modeling.	
	2	Threat Model, Wikipedia, published online at least as early as Feb. 3, 2018, available at https://en.wikipedia.org/wiki/Threat_model (note that some of the material in this Wikipedia article appears to be posted by one or more inventors of the present application or its parent provisional applications - some of the material appears to be posted by Brian Beyst, for example, who is an inventor on two of the parent provisional applications).	
	3	Comparisons of Threat Model Modeling Methodologies, published online by ThreatModeler at least as early as Apr. 15, 2016, available at http://threatmodeler.com/comparison-threat-modeling-methodologies/ (note that this article appears to be affiliated with a business organization of, and may have been published under the direction of, one or more inventors of the present application or its parent provisional applications).	
	4	Microsoft Threat Modeling Tool, published online by Microsoft, different sections published at different times from Aug. 22, 2017 to Jan. 24, 2018, available online at https://docs.microsoft.com/en-us/azure/opbuildpdf/security/TOC.pdf?branch=live	
	5	A description of an on-sale version of systems and methods, on sale in the U.S. by applicants at least as early as 12/31/2011, which on-sale version disclosed some of the elements disclosed in the present application. The description includes a detailed discussion of which claim elements of the claims as originally filed in the present application were disclosed in the 2011 version and which were not, and includes screenshots of user interfaces used in the system/methods of the 2011 version.	
	6	Lockheed Martin Corporation, "Seven Ways to Apply the Cyber Kill Chain with a Threat Intelligence Platform," published at least as early as 2015.	

If you wish to add additional non-patent literature document citation information please click the Add button Add

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	
	First Named Inventor	Anurag Agarwal
	Art Unit	
	Examiner Name	
	Attorney Docket Number	ThreatM010

EXAMINER SIGNATURE			
Examiner Signature	/THEODORE C PARSONS/	Date Considered	11/11/2019

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	
First Named Inventor	Anurag Agarwal
Art Unit	
Examiner Name	
Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2019-05-31
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

IriusRisk
Ex. 1002 - Page 420

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

Google Scholar

threat|vulnerability visual|depicting visio

Articles

About 4,250 results (0.06 sec)

Any time
Since 2019
Since 2018
Since 2015
Custom range...

Sort by relevance
Sort by date

include patents
include citations

Create alert

Microsoft SDL **threat** modelling tool

B Potter - Network Security, 2009 - Elsevier

... 2 In addition, Microsoft has released tools such as PREFast, a static code analysis capability integrated into **Visual** Studio ... To assist with **threat** modelling and analysis, Microsoft has released the SDL **Threat** Modelling Tool as a plug-in for **Visio** 2007 that creates a special ...

☆ 99 Cited by 27 Related articles All 3 versions

Preface for the Proceedings of the 13th Biannual Meeting of the Child Vision Research Society, June 2011 at Royal **Visio**, Huizen, the Netherlands

M Dik, M de Vries, H Verbunt, H Lijnders... - ..., 2012 - Taylor & Francis

... Society, June 2011 at Royal **Visio**, Huizen, the Netherlands ... Click to increase image sizeClick to decrease image size Free first page. During this year's meeting the objective was to share new insights into the consequences of the human **visual** system's **vulnerability** to early ...

☆ 99 Related articles All 3 versions

An experiment on comparing textual vs. **visual** industrial methods for security risk assessment

K Labunets, F Paci, F Massacci... - 2014 IEEE 4th ..., 2014 - ieeexplore.ieee.org

... The challenge revealed that **threat**-based methods performs better for security analysis. More recently, Labunets et al. [11] conducted a controlled experiment with 28 MSc students to compare two types of security risk assessment methods, **visual** (CORAS) and textual (SREP ...

☆ 99 Cited by 13 Related articles All 5 versions

Topological **vulnerability** analysis: A powerful new approach for network attack prevention, detection, and response

S Jajodia, S Noel - Algorithms, architectures and information ..., 2009 - World Scientific

... To manage **visual** complexity of attack graphs, our TVA system employs so- phisticated methods of recursive ... Topological **Vulnerability** Analysis 305 ... 6. Symantec Corporation, Symantec DeepSight **Threat** Management System, http:// tms.symantec.com/Default.aspx, last retrieved ...

☆ 99 Cited by 72 Related articles All 7 versions »

[PDF] SQL Slammer Worm

S Frei - Citeseer

... Single packet, UDP based attack that exploited the SQL Server **vulnerability** and executed its payload, turning the victim into a new distribution point ... SQL Server 2000 • **Visio** Enterprise Network Tools • **Visual** Basic .NET Standard 2002 • **Visual** C++ .NET ...

☆ 99 »

TAM 2: automated **threat** analysis

A Schaad, M Borozdin - Proceedings of the 27th Annual ACM ..., 2012 - dl.acm.org

... Such a **visual** representation by using different shapes is a simple case of introducing ... can load, parse and automatically analyse architecture diagrams created in Microsoft **Visio** with FMC ... database • Ability to create and assign bug tracker tickets for **threat/vulnerability** issues out ...

☆ 99 Cited by 17 Related articles »

Visual NRM User's Manual: Tools for Applying the Network Rating Methodology

AP Moore, B Strohmayer - 2000 - apps.dtic.mil

... This assumes that the mission can tolerate an interruption of the downgrade service, a **vulnerability** for the argument described, and that only the DO has keys to the cabinet ... The installation copies a template file for **Visual** NRM Drawings to the **Visio** Solutions directory ...

☆ 99 Cited by 3 Related articles All 3 versions »

Visio: Enabling interactive visualization of ultra-scale, time series data via high-bandwidth distributed i/o systems

C Mitchell, J Ahrens, J Wang - 2011 IEEE International Parallel ..., 2011 - ieeexplore.ieee.org

... resolutions rang- ing into billions of cells and write terabytes of data for **visual**- ization and ... Our **VisIO** enabled reader for ParaView is currently de- signed to operate on VTK MultiBlock files ... loss of a DataNode process or its underlying storage infrastructure poses no **threat** to the ...

☆ 99 Cited by 26 Related articles All 11 versions

Google Scholar | threat|vulnerability visual|depicting "cots" mapping

Articles | About 8,350 results (0.07 sec)

Any time
Since 2019
Since 2018
Since 2015
Custom range...

Sort by relevance
Sort by date

include patents
include citations

Create alert

Threat stream data generator: creating the known unknowns for test and evaluation of **visual** analytics tools
MA Whiting, W Cowley, J Haack, D Love... - Proceedings of the ..., 2006 - dl.acm.org
... We use existing commercial off-the- shelf (**COTS**) tools to support our work for this, including narrative description, Analysts ... of synthetic data sets with embedded threats that may be used for test and evaluation of **visual** analytic tools, a viable **threat** stream generator ...
☆ 99 Cited by 9 Related articles All 3 versions »

Uni-and bimodal **threat** cueing with vibrotactile and 3D audio technologies in a combat vehicle
O Carlander, L Eriksson - Proceedings of the Human Factors ..., 2006 - journals.sagepub.com
... belt with one row of tactors was developed from standard **COTS** (there are ... by multisensory neurons maximally activated by temporally and spatially synchronized **visual**, auditory, and ... Thus, **threat** warnings utilizing redundant cues or information from multisensory displays could ...
☆ 99 Cited by 32 Related articles All 2 versions »

OptaSense distributed acoustic and seismic sensing using **COTS** fiber optic cables for infrastructure protection and counter terrorism
GL Duckworth, EM Ku - Sensors, and command, control ..., 2013 - spiedigitallibrary.org
... OptaSense® distributed acoustic and seismic sensing using **COTS** fiber optic cables for Infrastructure Protection and Counter Terrorism ... detection and reliable classification of these activities allows an early response to assess the **threat**—through **visual** inspection and ...
☆ 99 Cited by 13 Related articles All 2 versions »

[PDF] Managing vulnerabilities in your commercial-off-the-shelf (**cots**) systems using an industry standards effort (cve)
RA Martin - Lecture notes in computer science, 2004 - researchgate.net
... 3rd International Conference on **COTS**-Based Software Systems Page 2. 2 Outline ... Development Tools ActivePerl ClearCase ColdFusion CVS Flash Frontpage GNU Emacs JRun WebLogic Server **Visual** Basic ... Provides MITRE with "**vulnerability**" item **mappings** to validate the ...
☆ 99 Cited by 1 Related articles All 10 versions »

Handling uni-and multimodal **threat** cueing with simultaneous radio calls in a combat vehicle setting
O Carlander, L Eriksson, PA Oskarsson - International Conference on ..., 2007 - Springer
... on the skin can thus be considered analogous to the perception of auditory or **visual** stimulation ... Furthermore, some commercials off the shelf (COTS) alternatives are now available that even have the ... Handling Uni- and Multimodal **Threat** Cueing with Simultaneous Radio Calls ...
☆ 99 Cited by 14 Related articles All 6 versions »

[HTML] ... -water coral habitats at different scales within the Northern Ionian Sea (Central Mediterranean): an assessment of coral coverage and associated **vulnerability**
A Savini, A Vertino, F Marchese, L Beuck, A Freiwald - PLoS One, 2014 - journals.plos.org
... Numerous types of predictive models have been employed for **mapping** habitat distributions [44], [45] ... coral coverage at a variety of spatial scales, our work sought to **map** the distribution of ... anthropogenic threats allowed us to provide a first insight into the **vulnerability** of CWC ...
☆ 99 Cited by 60 Related articles All 16 versions »

Enterprise **vulnerability** management
JM LaBumbard - US Patent 8,789,192, 2014 - Google Patents
... a solution to be quickly tailored to a specific customer's IT environment and **threat** exposure ... 3U-3V depict several **visual** outputs relating to the reporting portion of the computer ... Collect results (210) from **vulnerability** assessment from vendor tool(s); For each result: Analyze result ...
☆ 99 Cited by 90 Related articles All 4 versions »

[HTML] An empirical study of variations in **COTS**-based software development processes in the Norwegian IT industry
J Li, FO Bjørnson, R Conradi, VB Kampenes - Empirical Software ..., 2006 - Springer
... Then, the **COTS** modules that claim to be applicable are identified by the **mapping** from the domain models to the **COTS** modules. After that, the non-functional properties of the identified **COTS** modules are assessed ... 3,000. **Visual** Basic. El ... C++. Scientific. 4.4 **COTS** Components
☆ 99 Cited by 73 Related articles All 16 versions »

recipients at that address, it will be 'tarpitted'. The drawback is that this only works against spammers who use random-generated names for a particular target.

Sender Policy Framework (SPF)

A Sender Policy Framework record is held within DNS records and lists the addresses recognised as valid senders for that domain name, thus avoiding the abuse by spammers of that domain (to send out spam). The recipient's mail server checks the validity of the sender's address against the DNS record.

Take-up has been slow, meaning that this mechanism has not been very effective at combating spam. It is more often used to whitelist a sender than to block them.

Relationship management

Relationship management, shown in figure 3, is a relatively new development, offering significant improvement on any existing spam detection technology. It analyses not just content and IP address, but also applies learning from email user behaviour and relationships, to understand what the user does and does not want to receive. So in figure 3, if Sender B suddenly claimed to be sender A, a lot more than the email address would have to be spoofed.

Spammers do not have and cannot fake this sort of relationship with their targets. They get email addresses from a list and cannot know the other details. Like someone meeting up with a friend who has new clothes and a haircut, some details can change but the system will still recognise the relationship. This is proving to be a very effective anti-spam technique. It requires a learning period but as the system learns, the detection improves, providing a significant step towards improving anti-spam protection.

Conclusion

None of the techniques above can provide total protection in isolation and most have a way round them. The only real solution to the spam problem is to combine techniques, to create a probability score to assess whether an email is spam. Currently, this can achieve detection rates of 95–98%, depending on traffic, with a very low false positive rate. But emerging new techniques, like relationship management, are likely to significantly increase the success of spam detection.

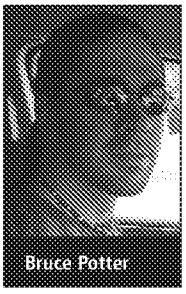
Spam is a continually moving target. Spammers have a financial motivation to find new ways round detection techniques. The defence systems that will succeed are those that are able to learn and adapt; and that are continuously developing new solutions.

Reference

1. "CANSPAM Act of 2003." *Spamlaws*, 9 November 2008 <www.spamlaws.com/f/pdf/108s877enrolled.pdf>

Microsoft SDL Threat Modelling Tool

Bruce Potter, CTO, Ponte Technologies



Gone are the days when software systems can be built without regard for security. Regardless of whether the system is commercial off the shelf (COTS), developed by a software provider, or custom code for use in an enterprise, it is clear that systems and networks are under attack and that the software being developed today is the primary line of defence.

Software engineers and architects as well as IT staff need to understand the risks to their systems, how the software mitigates the risk, and the risks that remain in the system. Unfortunately, though, it is not usually software engineers but security engineers who determine the risks to software and find ways to mitigate that risk. Ideally, security is integrated throughout the software development lifecycle (SDLC) from the inception of a project, through its design and development, right to its deployment and operation. Security should be 'baked in' from the beginning.

Integrating security into the SDLC has proven to be a huge challenge, however. Most software developers have a large gap in security-specific knowledge which prevents them from understanding the real risks to the system. There is also a distinct lack of tools to assist in tracking the mitigation of risks through the development process. Finally, integrating security into the SDLC is often viewed as expensive and disruptive to existing development methodologies such as Rational, Agile, and Extreme.

SDL Threat Modelling Tool

Enter the Microsoft Security Development Lifecycle (SDL) Threat Modelling Tool.[1] Over the last several years, Microsoft has made a significant investment in understanding how to build secure software and integrating security into the development of its own

Ex. 1002 - Page 424

products. Much of this information has been made public in books and articles including the *Security Development Lifecycle* by Howard and Lipner.[2] In addition, Microsoft has released tools such as PREFast, a static code analysis capability integrated into Visual Studio.

One of the core aspects of the SDL is to develop an understanding of risks to a system and how to mitigate them. Microsoft uses the acronym STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Escalation of Privilege) to summarise the breadth of threats that a system may face. If the user understands where these threats may be realised in the software and how the software addresses them, Microsoft believes that software can be created which defends itself from attackers.

To assist with threat modelling and analysis, Microsoft has released the SDL Threat Modelling Tool as a plug-in for Visio 2007 that creates a special environment for developers and architects to diagram systems and learn about threats to systems. The tool integrates seamlessly into Visio to create an intuitive, stand-alone experience.

Usage

The first thing a user of the SDL Threat Modelling Tool must do is enter in the design of the software system. A handful of different entities can be placed in the architecture, including processes, data stores, and communication channels. The diagram created is not a highly detailed design specification such as may be created via UML. Rather, it is higher level and would typically be referred to as 'architecture' in software parlance.

The Threat Modelling Tool also includes an idea of boundaries between entities. The boundaries are simple, such as process, machine, or trust boundaries. These boundaries are used by the tool to determine what threats exist across different communication channels.

Figure 1 shows a high-level breakdown of pfcapd, the capture element of the Psyche netflow analysis tool.[3] Pfcapd starts up using a configuration file. It receives UDP data from remote netflow sensors, processes them, and inserts the data into a Postgres database. There is a machine boundary between the pfcapd process and the netflow sensors and a process boundary between pfcapd and the database.

Once the entire architecture is entered into the tool, the user can move on to the analysis phase. The tool uses built-in rules from Microsoft's STRIDE threat model and applies them to the user's architecture. Each data store, communication channel, process, and interactor is taken in context with its boundary and all the different threat types are assigned to each one.

The threats listed in the Threat Type column in figure 2 are all part of the STRIDE threat breakdown. This simple breakdown of the pfcapd architecture resulted in approximately 40 different threats identified across a total of eight different entities.

Each threat can be expanded to reveal more specific information about it. Microsoft has provided information regarding each threat to assist software developers in understanding the impact of the threat and recommending potential mitigation techniques. For instance, the tool has identified tampering as a threat to the configuration directives obtained through the configuration file. Within the specific threat breakdown, the tool provides a definition of tampering, recommends using canonical names for access to data, and implementing auditing on access to the configuration file.

"The tool can even hook into bug tracking systems to determine if the security controls are causing defects"

Within the tool, there is space to document how this threat specifically impacts the software and precisely which actions are going to be taken to address the threat. This is where the software developer adds the context of the exact security controls to be implemented, and it is also where the real value of the tool becomes apparent. These controls can be tracked through the design phase and right through

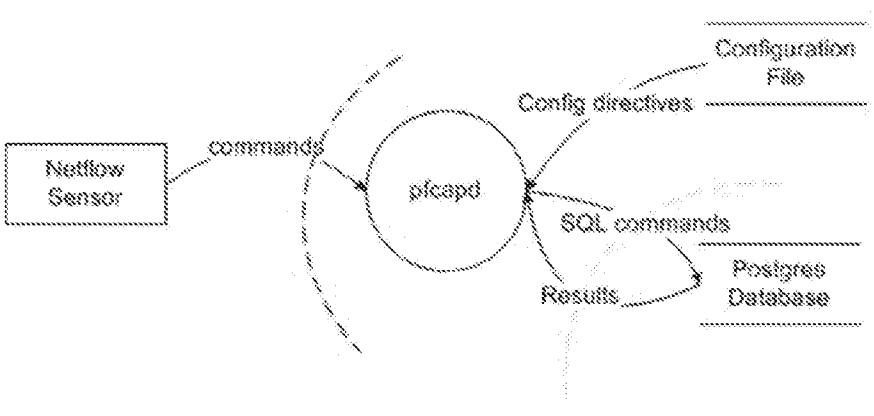


Figure 1: Simple example of an architecture created in the Threat Modeling Tool.

	ID	Element Name	Element Type	Element Diagram References	Threat Type
▶	3	commands (Netflo...	DataFlow	Context	Tampering
	4	commands (Netflo...	DataFlow	Context	InformationDisclos...
	5	commands (Netflo...	DataFlow	Context	DenialOfService
	12	Config directives (...	DataFlow	Context	Tampering
	13	Config directives (...	DataFlow	Context	InformationDisclos...
	14	Config directives (...	DataFlow	Context	DenialOfService
	38	Results (Postgres D...	DataFlow	Context	Tampering
	39	Results (Postgres D...	DataFlow	Context	InformationDisclos...
	40	Results (Postgres D...	DataFlow	Context	DenialOfService
	35	SQL commands (pf...	DataFlow	Context	Tampering
	36	SQL commands (pf...	DataFlow	Context	InformationDisclos...
	37	SQL commands (pf...	DataFlow	Context	DenialOfService
	21	Configuration File	DataStore	Context	Tampering
	22	Configuration File	DataStore	Context	Repudiation

Figure 2: Initial output of the analysis phase.

development and testing. At any point, someone could conduct a review of the software and validate that the controls created and approved during the architecture process are properly implemented and tested. The tool can even hook into bug tracking systems to determine if the security controls are causing defects that are found through the testing process.

Some threats may not be applicable. For example, either the tool has not been able to fully understand the boundaries and elements in the systems or there are threats that the user is willing to accept without mitigation.

Opinions

The SDL Threat Modelling Tool is oriented around software development. Many of the threat modelling/risk management processes for IT systems and software are geared around other factors such as attacker motivation, asset valuation within an enterprise, or historical activity by adversaries. The tool is relatively agnostic to the development process the user is using and the manner in which the user graphically describes the software. It is somewhat stand-alone in that regard and architectures will have to be created in the unique fashion required by the tool rather than directly ingesting lower-level design documents such as UML.

"The SDL Threat Modelling Tool does a great job of making threat modelling into a task that a huge number of software developers can execute"

The number of threats the tool generates is a bit more difficult to contend with. The more detailed the architectural description, the greater the number of threats that will be created. Furthermore, in order to be useful each of those threats needs to be analysed by a developer, the impact of the threat needs to be described, and the mitigation of it needs to be documented. In a reasonably complex system this could mean the analysis of hundreds of threats. It may be best to break apart systems into bite-sized chunks in order to not become overwhelmed by the volume of threats.

Microsoft has done a great job of documenting its Security Development Lifecycle and even providing tools to the public. Unfortunately, its tools and processes are what Microsoft uses for development. While the tools and processes have been very useful for Microsoft, they may not be as useful for other organisations, especially within custom development shops inside non-IT companies. No one can really write software like Microsoft can. It is unique in its size and mission. What is important to take away from its publications and tools is the intent of what is being accomplished, not the specific steps.

Parting shots

The SDL Threat Modelling Tool does a great job of making threat modelling (a process usually executed by dedicated security professionals) into a task that a huge number of software developers can execute. The STRIDE methodology is an important one, and any means of raising awareness of threats with software developers is a good thing. While the tool may not be perfect for every enterprise, especially as a tool to be used throughout the SDLC, it is worthy of at least one run through for each project. Ideally, software developers will become aware of potential weaknesses in their code and implement changes to address the threats.

Microsoft should be commended for the release of this tool. 'Baking security in' has been a mantra of the security

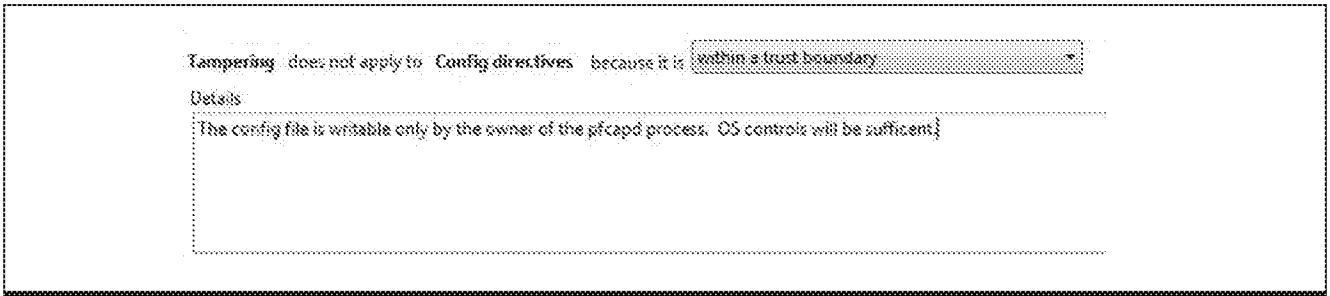


Figure 3: Threat modelling tool used to accept the tampering threat identified against the configuration directives.

Ex. 1002 - Page 426

industry for years. If developers are engaged and security processes such as threat modelling are demystified, software can be made more secure and software developers can become part of the solution rather than part of the problem.

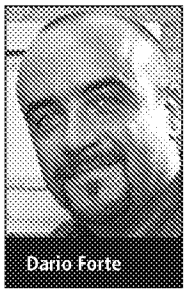
About the author

Bruce Potter is the founder of The Shmoo Group of security, crypto, and privacy professionals. He helps organise the yearly ShmooCon security conference held each winter in Washington DC. Potter is also the founder of Ponte Technologies, a company specialising in wireless security, IT security operations, and advanced network defence techniques.

References

1. "The Microsoft SDL Threat Modeling Tool v3." MSDN. Security Developer Center, 9 December 2008 <http://msdn.microsoft.com/en-us/security/dd206731.aspx>
2. Howard, Michael and Steve Lipner. *The Security Development Lifecycle.* Redmond, Washington: Microsoft Press, 2006.
3. "Psyche - Network Flow Analysis for the masses." Ponte Technologies, 9 December 2008 <http://psyche.ponteteс.com/>

So are we going to govern this security business?



Dario Forte, CFE, CISM, founder and CEO of DFLabs

Budget cuts, financial crises, compliance, and information security are currently at odds with one another in spite of the fact that they are all of equal strategic importance. They are managed in various ways by company functions which do not share the same priorities and budgets. IT security functions, for example, generally demonstrate a tactical commitment to maintaining existing platforms, with some emphasis on 'emergency' issues such as data loss prevention. At higher levels, some attention is paid to internal monitoring for prevention of fraud, general company security incidents, and illicit acts, the cost of which is high but justified by the requisites of good governance.

A recent initiative by Carnegie Mellon University concentrated on security governance. Carnegie Mellon is home to CyLab, a university-based cybersecurity research and education institute. CyLab's specialised researchers hail from all over the world and include Richard Power, the brain behind the CSI/FBI Computer Crime and Security Survey. CyLab has undertaken a survey called the CyLab Governance of Enterprise Security Survey (Governance Survey), which measures the degree of governance supported by boards of directors (BoDs) and senior management.[1] The study was carried out on a sample of 703 managers at listed American companies. Two-thirds of the respondents were external board members while the rest were internal personnel and non-voting board attendees who include senior management and general counsels.

The results

The survey revealed that boards take risk management extremely seriously but that there is still a notable gap in understanding the connection between IT and enterprise risk management. Many answers confirmed that boards and numerous senior executives are not adequately involved in the 'key areas' of enterprise security governance. Only 36% of the respondents indicated that the board had a direct role in determining company information security policies. A significant number of existing best practices relate to the involvement of the board in IT governance roles, but the survey results indicate that BoDs are only sporadically interested in the activities specified in these best practices.

One of the survey recommendations is greater involvement of top management in governance. Some respondents confirmed that many BoDs involved in revising security and privacy policy focus exclusively on the formal aspects. They do not pay attention to substantial controls or more important lines of conduct, such as addressing risks to company reputation or losses arising from faulty or improper processing of personal information (which is illegal under Italian law).

"Only 17% of respondents spoke of a well defined and effectively functioning security management organisation"

The Governance Survey revealed that BoDs place great trust in the work of audit committees, overestimating their role and responsibility at times such as in the event of an incident. The survey also indicated that many boards do not

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	33220	(G06F21/577 or G06F16/2282 or G06F16/248 or G06F9/451 or G06F16/212 or G06F2221/033).cpc.	US-PGPUB; USPAT	OR	ON	2019/10/28 06:11
S2	1	("20190147161").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2019/10/28 06:15
S3	1	"66431256".FMID.	US-PGPUB; USPAT; FPRS; JPO	OR	ON	2019/10/28 06:15
S4	0	("16542263" 16/542263).app,apn,rlan,ap.	US-PGPUB; USPAT; USOCR	OR	ON	2019/10/28 06:16
S5	6	(15/888021 "15888021" 15/922856 "15922856" 16/228738 "16228738" 16/270276 "16270276" 16/542263 "16542263" "16664679" 16/664679 62/507691 "62507691" 62/520954 "62520954" 62/527671 "62527671" 62/530295 "62530295").app,apn,rlan,ap.	US-PGPUB; USPAT; USOCR	OR	ON	2019/10/28 06:18
S6	72	US-9537884-$.did. US-9584536-$.did. US-9591010-$.did. US-9596254-$.did. US-9609009-$.did. US-9609011-$.did. US-9667641-$.did. US-9680855-$.did. US-9699205-$.did. US-9710644-$.did. US-9787714-$.did. US-9813435-$.did. US-9838410-$.did. US-10003605-$.did. US-10015177-$.did. US-10019677-$.did. US-10021138-$.did. US-10027711-$.did. US-10032020-$.did. US-10038707-$.did. US-10063570-$.did. US-10069849-$.did. US-10110617-$.did. US-10116670-$.did. US-10135848-$.did. US-10146944-$.did. US-10148677-$.did. US-10154047-$.did. US-10158652-$.did. US-10158677-$.did. US-10193901-$.did. US-10200399-$.did. US-10205735-$.did. US-10212174-$.did. US-10216938-$.did. US-10243970-$.did. US-10255439-$.did. US-10291635-$.did. US-10296748-$.did. US-10298602-$.did. US-10313203-$.did. US-10332003-$.did. US-10375093-$.did. US-10389738-$.did. US-10419450-$.did. US-10419462-$.did. US-10419463-$.did. US-10419465-$.did. US-10432660-$.did. US-20010027388-$.did. US-20010027389-$.did. US-20030033516-$.did. US-20050071643-$.did. US-20050193269-$.did. US-20070005654-$.did.	US-PGPUB; USPAT	OR	ON	2019/10/28 07:08

		US-20070192344-$.did. US-20070204346-$.did. US-20070222589-$.did. US-20070294766-$.did. US-20080034424-$.did. US-20080034425-$.did. US-20080052054-$.did. US-20090083695-$.did. US-20090319247-$.did. US-20090319248-$.did. US-20090319249-$.did. US-20090319647-$.did. US-20090319906-$.did. US-20090320137-$.did. US-20100235918-$.did. US-20100275263-$.did. US-20110126111-$.did. US-20120216243-$.did.				
S7	64	US-9888023-$.did. US-9900332-$.did. US-9906539-$.did. US-9906554-$.did. US-20120224057-$.did. US-20130325545-$.did. US-20130339930-$.did. US-20140046644-$.did. US-20140046645-$.did. US-20140157417-$.did. US-20140218389-$.did. US-20140337971-$.did. US-20150051893-$.did. US-20150207813-$.did. US-20150281287-$.did. US-20150381649-$.did. US-20160057167-$.did. US-20160162690-$.did. US-20160173446-$.did. US-20170026401-$.did. US-20170048273-$.did. US-20170063886-$.did. US-20170063887-$.did. US-20170063888-$.did. US-20170063889-$.did. US-20170063890-$.did. US-20170063891-$.did. US-20170063894-$.did. US-20170063896-$.did. US-20170063897-$.did. US-20170063898-$.did. US-20170063899-$.did. US-20170063900-$.did. US-20170063901-$.did. US-20170063902-$.did. US-20170063903-$.did. US-20170063904-$.did. US-20170063905-$.did. US-20170063906-$.did. US-20170063907-$.did. US-20170063908-$.did. US-20170063909-$.did. US-20170063910-$.did. US-20170063911-$.did. US-20170063912-$.did. US-20170134410-$.did. US-20170134415-$.did. US-20170134418-$.did. US-20170142140-$.did. US-20170163673-$.did. US-20170223036-$.did. US-20170237776-$.did. US-20170251013-$.did. US-20170272458-$.did. US-20170279843-$.did. US-20170316203-$.did. US-20170324768-$.did. US-20170329975-$.did. US-20180013771-$.did. US-20180041537-$.did. US-20180054452-$.did. US-20180069888-$.did. US-20180091528-$.did. US-20180146000-$.did.	US-PGPUB; USPAT	OR	ON	2019/10/28 07:08
S8	72	US-20180183766-$.did. US-20180191754-$.did. US-20180219888-$.did. US-20180219894-$.did. US-20180219897-$.did. US-20180219919-$.did. US-20180288079-$.did. US-20180295154-$.did. US-20180302423-$.did. US-20180309790-$.did. US-20180316705-$.did. US-20180316706-$.did. US-20180316713-$.did. US-20180316727-$.did. US-20180336355-$.did. US-20180337939-$.did. US-20180351981-$.did. US-20180357422-$.did. US-20180367551-$.did. US-20180367561-$.did. US-20190028498-$.did. US-20190034626-$.did. US-20190075126-$.did. US-20190089742-$.did. US-20190095599-$.did.	US-PGPUB; USPAT	OR	ON	2019/10/28 07:08

		US-20190098026-$.did. US-20190098068-$.did. US-20190104155-$.did. US-20190109868-$.did. US-20190124104-$.did. US-20190158517-$.did. US-20190171934-$.did. US-20190173893-$.did. US-20190173919-$.did. US-20190173921-$.did. US-20190205544-$.did. US-20190222597-$.did. US-20190238574-$.did. US-20190245894-$.did. US-20190260782-$.did. US-20190260804-$.did. US-20190289030-$.did. US-20190327251-$.did. US-6502102-$.did. US-6519571-$.did. US-6523027-$.did. US-6601233-$.did. US-6609128-$.did. US-6633878-$.did. US-6701514-$.did. US-6704873-$.did. US-6718535-$.did. US-6907546-$.did. US-7100195-$.did. US-7130779-$.did. US-7231327-$.did. US-7243374-$.did. US-7308388-$.did. US-7437614-$.did. US-7712137-$.did. US-7891003-$.did. US-7934253-$.did. US-8091065-$.did. US-8429751-$.did. US-8532970-$.did. US-8751629-$.did. US-8769412-$.did. US-9032533-$.did. US-9246768-$.did. US-9292874-$.did. US-9509715-$.did. US-9516053-$.did.				
S9	208	S8 S7 S6	US-PGPUB; USPAT	OR	ON	2019/10/28 07:08
S10	50	US-10110617-$.did. US-10116670-$.did. US-10135848-$.did. US-10148677-$.did. US-10154047-$.did. US-10193901-$.did. US-10200399-$.did. US-10212174-$.did. US-10255439-$.did. US-10291635-$.did. US-10296748-$.did. US-10419450-$.did. US-10419462-$.did. US-20070294766-$.did. US-20090083695-$.did. US-20160057167-$.did. US-20170063886-$.did. US-20170063894-$.did. US-20170063897-$.did. US-20170063898-$.did. US-20170063899-$.did. US-20170063900-$.did. US-20170063901-$.did. US-20170063902-$.did. US-20170063903-$.did. US-20170063905-$.did. US-20170063906-$.did. US-20170063907-$.did. US-20170134415-$.did. US-20170142140-$.did. US-20170223036-$.did. US-20180054452-$.did. US-20180069888-$.did. US-20180146000-$.did. US-20180219897-$.did. US-20180316706-$.did. US-20180316713-$.did. US-20180336355-$.did. US-20180337939-$.did. US-20180351981-$.did. US-20180357422-$.did. US-20190109868-$.did. US-20190158517-$.did. US-20190173893-$.did. US-20190205544-$.did. US-7891003-$.did. US-8091065-$.did. US-9516053-$.did. US-9609009-$.did. US-9609011-$.did.	US-PGPUB; USPAT	OR	ON	2019/10/28 07:10
S11	20	("7433829" | "7900259" | "8407801" | "8413237" | "20060015941" | "20110178942" | "20060241991" | "20140137257" | "8413249" | "9043924" | "20110126111" | "9774613" | "4959015" | "9497203" | "7096502" | "7891003" | "6952779" | "20180324207" | "8191139" |	US-PGPUB; USPAT	OR	ON	2019/10/28 07:27

		"8255995").PN.				
S25	334929	((visual depict$5 graphical viso shape) same (software system process network application) same (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:28
S26	113073	((visual depict$5 graphical viso shape) near20 (software system process network application) near20 (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:32
S27	33530	(G06F21/577 or G06F16/2282 or G06F16/248 or G06F9/451 or G06F16/212 or G06F2221/033).cpc.	US-PGPUB; USPAT	OR	ON	2019/11/12 07:32
S28	143948	S27 S26	US-PGPUB; USPAT	OR	ON	2019/11/12 07:32
S29	143948	S28 and (@pd<"20190901" @ad<"20190901" @rlad<"20190901")	US-PGPUB; USPAT	OR	ON	2019/11/12 07:33
S30	116404	S28 and ((visual depict$5 graphical viso shape) same (software system process network application) same (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:34
S31	133272	S28 and ((model$4 visual depict$5 graphical viso shape) near12 (software system process network application))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:35
S32	108663	S28 and ((visual depict$5 graphical viso shape) near12 (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:35
S33	108663	S28 and ((visual depict$5 graphical viso shape) near12 (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:35
S34	5512	S28 and ((threat vulnerability risk) near12 (model$4 visual depict$5 graphical viso shape))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:36
S35	16159	S28 and ((threat vulnerability risk) near12 (software system process network application))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:36
S36	2496	S28 and ((threat vulnerability risk) near12 (visual depict$5 graphical viso shape))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:36
S37	2030	S28 and ((threat vulnerability risk) same (visual depict$5 graphical viso shape) same (software system process network application) same (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:37
S38	677	S29 and S31 and S32 and S33 and S34 and S35 and S36 and S37	US-PGPUB; USPAT	OR	ON	2019/11/12 07:37
S39	3473	S28 and ((correspond$4 map mapping mapped correlat$4 associat$5) same (attack$4 threat vulnerability risk) same (visual depict$5 graphical viso shape))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:39
S40	516	S39 and S29 and S31 and S32 and S33 and S34 and S35 and S36 and S37	US-PGPUB; USPAT	OR	ON	2019/11/12 07:39
S41	2058	S28 and (((correspond$4 map mapping mapped correlat$4 associat$5) near12 (attack$4 threat vulnerability risk)) same	US-PGPUB; USPAT	OR	ON	2019/11/12 07:40

		(visual depict$5 graphical viso shape))				
S42	3015	S28 and (((map mapping mapped correlat$4) near20 (attack$4 threat vulnerability risk)))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:41
S43	241	S41 and S42 and S29 and S31 and S32 and S33 and S34 and S35 and S36 and S37	US-PGPUB; USPAT	OR	ON	2019/11/12 07:41
S44	397	S28 and ((map mapping mapped correlat$4).ab,ti,clm. and (attack$4 threat vulnerability risk).ab,ti,clm. and (visual$4 visualiz$5 visualis$5 depict$5 graphical viso shape).ab,ti,clm.)	US-PGPUB; USPAT	OR	ON	2019/11/12 07:43
S45	23479	S28 and ((map mapping mapped correlat$4).ab,ti,clm.)	US-PGPUB; USPAT	OR	ON	2019/11/12 07:43
S46	7066	S28 and ((attack$4 threat vulnerability risk).ab,ti,clm.)	US-PGPUB; USPAT	OR	ON	2019/11/12 07:43
S47	41619	S28 and ((visual$4 visualiz$5 visualis$5 depict$5 graphical viso shape).ab,ti,clm.)	US-PGPUB; USPAT	OR	ON	2019/11/12 07:43
S48	86	(S46 and S47) and S41 and S42 and S29 and S31 and S32 and S33 and S34 and S35 and S36 and S37	US-PGPUB; USPAT	OR	ON	2019/11/12 07:44
S49	89	(09/500101 | 09/500108 | 09/500269 | 09/704838 | 10/802613 | 10/888965 | 11/053575 | 11/262425 | 11/442479 | 11/710874 | 11/869388 | 11/912660 | 11/954801 | 12/354481 | 12/485775 | 12/641128 | 12/775120 | 12/889904 | 12/889963 | 12/890018 | 12/895847 | 13/413586 | 13/472857 | 13/525298 | 13/559692 | 13/559707 | 13/559732 | 13/670420 | 13/670435 | 13/690517 | 13/722731 | 13/800730 | 13/830759 | 13/897639 | 13/947508 | 14/078514 | 14/213986 | 14/305671 | 14/623333 | 14/683362 | 14/873075 | 14/928563 | 14/934966 | 15/086330 | 15/118999 | 15/217129 | 15/234970 | 15/554553 | 15/560411 | 15/560882 | 15/583623 | 15/872285 | 15/888021 | 15/922856 | 15/940604 | 15/956266 | 16/058726 | 16/130352 | 16/164288 | 16/228738 | 16/270276 | 16/278957).APP.	US-PGPUB; USPAT; USOCR	OR	ON	2019/11/12 07:45
S50	89	S49 and ((diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 viso shape) same (attack$4 threat vulnerability risk) same ((architecture model modelling) (map mapping mapped correlat$4)) same ((attack$4 threat vulnerability risk) (correspond$4 map mapping mapped correlat$4 associat$5)))	US-PGPUB; USPAT	OR	ON	2019/11/12 07:49
S51	69	S49 and ((map mapping mapped) same (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 viso shape) same (attack$4 threat vulnerability risk) same ((attack$4 threat vulnerability risk) (architecture model modelling) (correspond$4 map mapping mapped	US-PGPUB; USPAT	OR	ON	2019/11/12 07:51

		correlat$4 associat$5)))				
S52	8584	S28 and ((g06f21/5$).cpc.)	US-PGPUB; USPAT	OR	ON	2019/11/12 09:21
S53	28	S52 and (S46 and S47) and S41 and S42 and S29 and S31 and S32 and S33 and S34 and S35 and S36 and S37	US-PGPUB; USPAT	OR	ON	2019/11/12 09:21
S54	28	(09/500269 | 12/775120 | 12/895847 | 13/413586 | 13/559692 | 13/559707 | 13/559732 | 13/690517 | 14/078514 | 14/623333 | 15/086330 | 15/554553 | 15/888021 | 15/922856 | 15/940604 | 16/130352 | 16/228738 | 16/270276 | 16/278957).APP.	US-PGPUB; USPAT; USOCR	OR	ON	2019/11/12 09:22
S55	28	S54 and ((diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 viso shape) same (attack$4 threat vulnerability risk) same ((architecture model modelling) (map mapping mapped correlat$4)) same ((attack$4 threat vulnerability risk) (correspond$4 map mapping mapped correlat$4 associat$5)))	US-PGPUB; USPAT	OR	ON	2019/11/12 09:22
S56	6	(US-20160248798-$ or US-20140157417-$ or US-20130298243-$ or US-20110277034-$).did. or (US-8413249-$ or US-6883101-$).did.	US-PGPUB; USPAT	OR	ON	2019/11/12 09:30
S57	6	S56 and ((correspond$4 map mapping mapped correlat$4 associat$5) same (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 viso shape) same (attack$4 threat vulnerability risk) same ((attack$4 threat vulnerability risk) (architecture model modelling)))	US-PGPUB; USPAT	OR	ON	2019/11/12 09:32
S58	6	S56 and (((correspond$4 map mapping mapped correlat$4 associat$5) near12 ((diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape) (attack$4 threat vulnerability risk))) same (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape) same (attack$4 threat vulnerability risk) same ((attack$4 threat vulnerability risk) (architecture model modelling)))	US-PGPUB; USPAT	OR	ON	2019/11/12 09:34
S59	20	("7433829" | "7900259" | "8407801" | "8413237" | "20060015941" | "20110178942" | "20060241991" | "20140137257" | "8413249" | "9043924" | "20110126111" | "9774613" | "4959015" | "9497203" | "7096502" | "7891003" | "6952779" | "20180324207" | "8191139" | "8255995").PN.	US-PGPUB; USPAT	OR	ON	2019/11/12 09:42
S60	7	S59 and ((diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape) same (correspond$4 map mapping mapped correlat$4 associat$5) same ((architecture model modelling) (map mapping mapped correlat$4)) same ((architecture model modelling) (attack$4 threat vulnerability risk)))	US-PGPUB; USPAT	OR	ON	2019/11/12 09:42
S61	50	US-10110617-$.did. US-10116670-$.did. US-	US-	OR	ON	2019/11/12

		10135848-$.did. US-10148677-$.did. US-10154047-$.did. US-10193901-$.did. US-10200399-$.did. US-10212174-$.did. US-10255439-$.did. US-10291635-$.did. US-10296748-$.did. US-10419450-$.did. US-10419462-$.did. US-20070294766-$.did. US-20090083695-$.did. US-20160057167-$.did. US-20170063886-$.did. US-20170063894-$.did. US-20170063897-$.did. US-20170063898-$.did. US-20170063899-$.did. US-20170063900-$.did. US-20170063901-$.did. US-20170063902-$.did. US-20170063903-$.did. US-20170063905-$.did. US-20170063906-$.did. US-20170063907-$.did. US-20170134415-$.did. US-20170142140-$.did. US-20170223036-$.did. US-20180054452-$.did. US-20180069888-$.did. US-20180146000-$.did. US-20180219897-$.did. US-20180316706-$.did. US-20180316713-$.did. US-20180336355-$.did. US-20180337939-$.did. US-20180351981-$.did. US-20180357422-$.did. US-20190109868-$.did. US-20190158517-$.did. US-20190173893-$.did. US-20190205544-$.did. US-7891003-$.did. US-8091065-$.did. US-9516053-$.did. US-9609009-$.did. US-9609011-$.did.	PGPUB; USPAT			09:42
S62	47	S61 and ((diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape) same (correspond$4 map mapping mapped correlat$4 associat$5) same ((architecture model modelling) (map mapping mapped correlat$4)) same ((architecture model modelling) (attack$4 threat vulnerability risk)))	US-PGPUB; USPAT	OR	ON	2019/11/12 09:42
S63	6	(US-20160248798-$ or US-20140157417-$ or US-20130298243-$ or US-20110277034-$).did. or (US-8413249-$ or US-6883101-$).did.	US-PGPUB; USPAT	OR	ON	2019/11/12 10:05
S64	3	S63 and ((((attack$4 threat vulnerability risk) near4 (model$4 diagram report)) near20 (component asset visio vsd vsdx shape)) same ((attack$4 threat vulnerability risk) (model$4 diagram report) (component asset visio vsd vsdx shape) (architecture model modelling) (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape)))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:05
S65	3	S63 and ((map mapping mapped correlat$4) same (((attack$4 threat vulnerability risk) near4 (model$4 diagram report)) near20 (component asset visio vsd vsdx shape)) same ((attack$4 threat vulnerability risk) (model$4 diagram report) (component asset visio vsd vsdx shape) (architecture model modelling) (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape)))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:19
S66	1	(US-20150033346-$).did.	US-PGPUB; USPAT	OR	ON	2019/11/12 10:41
S67	1	S66 and ((map mapping mapped correlat$4	US-	OR	ON	2019/11/12

		) same (((attack$4 threat vulnerability risk) near4 (model$4 diagram report)) near20 (component asset visio vsd vsdx shape)) same ((attack$4 threat vulnerability risk) (model$4 diagram report) (component asset visio vsd vsdx shape) (architecture model modelling) (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape)))	PGPUB; USPAT			10:41
S68	1	S66 and ((((attack$4 threat vulnerability risk) near4 (model$4 diagram report)) near20 (component asset visio vsd vsdx shape)) same ((attack$4 threat vulnerability risk) (model$4 diagram report) (component asset visio vsd vsdx shape) (architecture model modelling) (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape)))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:41
S69	1	S66 and (((third 3rd independent separate visio existing known cots! commercial$4 available conventional) near20 (model$4 diagram architecture component format)))	US-PGPUB; USPAT	OR	ON	2019/11/12 11:02
S70	5	(US-20160248798-$ or US-20140157417-$ or US-20110277034-$).did. or (US-8413249-$).did. (US-20150033346-$).did.	US-PGPUB; USPAT	OR	ON	2019/11/12 11:06
S71	7	(12/775120 | 12/895847 | 13/690517 | 15/086330).APP.	US-PGPUB; USPAT; USOCR	OR	ON	2019/11/12 11:07
S72	7	S71 and (((shelf third 3rd independent separate visio existing cots! commercial$4 conventional) near20 (model$4 diagram architecture component format)))	US-PGPUB; USPAT	OR	ON	2019/11/12 11:08

EAST Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S12	56	((("AGARWAL") near3 ("Anurag"))).INV.	US-PGPUB; USPAT	OR	ON	2019/10/28 06:14
S13	4	(THREATMODELER).AS,AANM.	US-PGPUB; USPAT	OR	ON	2019/10/28 06:14
S14	56	((("AGARWAL") near3 ("Anurag"))).INV.	US-PGPUB; USPAT	OR	ON	2019/11/11 21:18
S15	4	(THREATMODELER).AS,AANM.	US-PGPUB; USPAT	OR	ON	2019/11/11 21:18
S16	56	S14 S15	US-PGPUB; USPAT	OR	ON	2019/11/11 21:18
S17	3	S16 and ((map mapping mapped).clm.)	US-PGPUB; USPAT	OR	ON	2019/11/11 21:22
S18	11	S16 and ((model$4 visual depict$5 graphic visio template component shape connector).clm. near12 ((correspond$4 map	US-PGPUB; USPAT	OR	ON	2019/11/11 21:27

		mapping mapped correlat$4 associat$5).clm. (threat vulnerability risk attack$4).clm.))				
S19	9	S16 and ((threat vulnerability risk attack$4).clm. near12 (correspond$4 map mapping mapped correlat$4 associat$5).clm.)	US-PGPUB; USPAT	OR	ON	2019/11/11 21:34
S20	8	S18 and (S17 S19)	US-PGPUB; USPAT	OR	ON	2019/11/11 21:34
S21	8	"Term Removed"	US-PGPUB; USPAT	OR	ON	2019/11/11 21:35
S22	8	S21 and ((correspond$4 map mapping mapped correlat$4 associat$5).clm. near12 ((model$4 visual depict$5 graphic visio template component shape connector).clm. (threat vulnerability risk attack$4).clm.))	US-PGPUB; USPAT	OR	ON	2019/11/11 21:35
S23	0	(S18 S19) and S17	US-PGPUB; USPAT	OR	ON	2019/11/11 21:36
S24	0	S23 and ((correspond$4 map mapping mapped correlat$4 associat$5).clm. near12 ((model$4 visual depict$5 graphic visio template component shape connector).clm. (threat vulnerability risk attack$4).clm.))	US-PGPUB; USPAT	OR	ON	2019/11/11 21:36
S73	76	(architecture model modelling).clm. same (component asset).clm. same (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm. same (threat vulnerability risk attack$4).clm.	US-PGPUB; USPAT	OR	ON	2019/11/12 10:31
S74	23	S73 and ((map mapping mapped correlat$4 correspond$3).clm. near12 ((threat vulnerability risk attack$4).clm.))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:32
S75	14	S73 and ((map mapping mapped correlat$4 correspond$3).clm. near12 ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm.))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:32
S76	56	((("AGARWAL") near3 ("Anurag"))).INV.	US-PGPUB; USPAT	OR	ON	2019/11/12 10:34
S77	4	(THREATMODELER).AS,AANM.	US-PGPUB; USPAT	OR	ON	2019/11/12 10:34
S78	56	S76 S77	US-PGPUB; USPAT	OR	ON	2019/11/12 10:34
S79	6	S78 and S73	US-PGPUB; USPAT	OR	ON	2019/11/12 10:34
S80	0	S78 and ((map mapping mapped correlat$4 correspond$3).clm. near12 ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm.))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:34
S81	2	S78 and ((map mapping mapped correlat$4 correspond$3).clm. near12 ((threat vulnerability risk attack$4).clm.))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:34
S82	0	S78 and ((map mapping mapped correlat$4 correspond$3).clm. near30 ((component asset).clm. (diagram visio vsd vsdx shape visual	US-PGPUB; USPAT	OR	ON	2019/11/12 10:35

		visualiz$5 depict$5 graphic$3).clm.))				
S83	12	S75 and S74	US-PGPUB; USPAT	OR	ON	2019/11/12 10:36
S84	16	"Term Removed"	US-PGPUB; USPAT	OR	ON	2019/11/12 10:37
S85	16	S84 and (((architecture model modelling).clm. same (threat vulnerability risk attack$4).clm. same ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm.)) or ((map mapping mapped correlat$4 correspond$3).clm. near12 ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm. (threat vulnerability risk attack$4).clm.)))	US-PGPUB; USPAT	OR	ON	2019/11/12 10:38

11/ 12/ 2019 11:16:21 AM
C:\ Users\ tparsons1\ Documents\ EAST\ Workspaces\ 16542263-track1.wsp

Google Patents

(~patent/US20190147161A1)

SEARCH TERMS

~patent/US20190147161A1 ×

+ Synonym

+ Synonym

SEARCH FIELDS

Date · Priority

YYYY-MM-DD — YYYY-MM-DD

+ Inventor

+ Assignee

Patent Office · Language

Status · Type

Litigation

About 4,915 results

Download

Sort by · Relevance · Group by · None · Deduplicate by · Family · Results / page · 10

System and method for cyber attacks analysis and decision support



US · US9426169B2 · Shay ZANDANI · Cytegic Ltd.

Priority 2012-02-29 · Filed 2013-02-14 · Granted 2016-08-23 · Published 2016-08-23

Similar A method for cyber attack risk assessment, the method including operating at least one hardware processor for: collecting global cyber attack data from a networked resource; collecting organizational profile data from a user, wherein the organizational profile data includes: types of computerized ...

Methods and systems for monitoring user, application or device activity



US · US20050183143A1 · Eric Anderholm · Anderholm Eric J.

Priority 2004-02-13 · Filed 2004-02-13 · Published 2005-08-18

Similar 1 . A method of managing security in an enterprise, comprising: detecting at periodic intervals events that correspond to user interactions with computers connected to a network of the enterprise; storing such events in a data facility; organizing the events by user, by computer and by event type;

Systems for network risk assessment including processing of user access rights ...



US · US9648036B2 · Miles Seiver · Palantir Technologies Inc.

Priority 2014-12-29 · Filed 2016-07-13 · Granted 2017-05-09 · Published 2017-05-09

Similar Methods, systems, and apparatus, including computer programs encoded on computer storage media, for network risk assessment. One of the methods includes obtaining information describing network traffic between a plurality of network devices within a network. A network topology of the network is ...

System for identifying data relationships



~~WO~~ ~~US~~ ~~AU~~ · US20050085378A1 · Robert Groat · Robert Groat

Priority 2001-12-14 · Filed 2002-12-13 · Published 2005-04-20

Similar 1 . A method for identifying data relationships comprising the steps of: (a) providing source data; and (b) reducing said data into canonical data, said canonical data being capable of being used in an inferential web that shows a user connections between at least two entities. 2 . The method of

Malware data item analysis



US · US9021260B1 · Matthew Falk · Palantir Technologies Inc.

Priority 2014-07-03 · Filed 2014-08-29 · Granted 2015-04-28 · Published 2015-04-28

Similar Embodiments of the present disclosure relate to a data analysis system that may automatically analyze a suspected malware file, or group of files. Automatic analysis of the suspected malware file(s) may include one or more automatic analysis techniques. Automatic analysis of may include production ...

Threat model chaining and attack simulation systems and methods



US · US10200399B2 · Anurag Agarwal · Threatmodeler Software Inc.

Priority 2017-05-17 · Filed 2018-03-15 · Granted 2019-02-05 · Published 2019-02-05

Similar Attack simulation systems include a computing device coupled with a database, the device displaying input interfaces configured to store a plurality of threat model components, threats, and compensating controls in the database, and associate each stored threat with at least one stored component ...

Threat Model Chaining and Attack Simulation Systems and Related Methods



US · US20190147161A1 · Anurag Agarwal · Threatmodeler Software Inc.

Priority 2017-05-17 · Filed 2018-12-20 · Published 2019-05-16

Similar Threat model chaining methods include providing one or more databases including a threat model components, threats, each threat associated with at least one of the threat model components, and compensating controls, each compensating control associate with one of the threats, providing a diagram ...

Threat analysis and modeling during a software development lifecycle of a ...



US · US8099165B2 · Talhah Munawar Mir · Microsoft Corporation

Priority 2007-09-25 · Filed 2007-09-25 · Granted 2012-01-03 · Published 2012-01-03

Similar Systems and methods relating to a method for generating a threat analysis and modeling tool are described. In an implementation, aggregate analysis is performed upon applications of an enterprise for complete risk management of the enterprise. The threat analysis model is generated by defining the ...

Malware data item analysis



EP US · EP2963578A1 · Matthew Falk · Palantir Technologies, Inc.

Priority 2014-07-03 · Filed 2015-07-02 · Published 2016-01-06

Similar Embodiments of the present disclosure relate to a data analysis system that may automatically analyze a suspected malware file, or group of files. Automatic analysis of the suspected malware file(s) may include one or more automatic analysis techniques. Automatic analysis of may include production ...

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2491
Examiner Name	Patel, Ashokkumar B.
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	9043924	B2	2015-05-26	Maor et al.	
	2	8413249	B1	2013-04-02	Chou	
	3	8407801	B2	2013-03-26	Ikegami et al.	
	4	6952779	B1	2005-10-04	Cohen et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

IriusRisk
Ex. 1002 - Page 439

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2491
Examiner Name	Patel, Ashokkumar B.
Attorney Docket Number	ThreatM010

	1						

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1		

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature	/THEODORE C PARSONS/	Date Considered	10/28/2019

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2491
Examiner Name	Patel, Ashokkumar B.
Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2019-09-05
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P/

1 – 50 2,690,830 results

1. **Threat Model Chaining and Attack Simulation Systems and Methods**

Attack simulation systems include a computing device coupled with a database, the device displaying input interfaces configured to store a plurality of threat model components, threats, and compensating controls in the database, and associate each stored threat with at least one stored component and...

CURRENT ASSIGNEES: THREATMODELER SOFTWARE INC
US20180337939 | US APPLICATIONS | 22-NOV-2018

2. **Threat model chaining and attack simulation systems and methods**

Attack simulation systems include a computing device coupled with a database, the device displaying input interfaces configured to store a plurality of threat model components, threats, and compensating controls in the database, and associate each stored threat with at least one stored component and...

CURRENT ASSIGNEES: THREATMODELER SOFTWARE INC
US10200389 | US PATENTS | 05-FEB-2019

3. **Threat modeling systems and related methods including compensating controls**

Threat modeling methods include, in response to receiving user input using computing device interfaces: storing threat model components, threats, and security requirements in a database; associating each threat with a component; storing an indication of whether each security requirement is a...

CURRENT ASSIGNEES: THREATMODELER SOFTWARE INC
US20180336355 | US APPLICATIONS | 22-NOV-2018

4. **Threat Modeling Systems and Related Methods Including Compensating Controls**

Threat modeling methods include, in response to receiving user input using computing device interfaces: storing threat model components, threats, and security requirements in a one or more database(s); associating each threat with a component; storing an indication of whether each security...

CURRENT ASSIGNEES: THREATMODELER SOFTWARE INC
US20190205544 | US APPLICATIONS | 04-JUL-2019

5. **Threat modeling systems and related methods including compensating controls**

Threat modeling methods include, in response to receiving user input using computing device interfaces: storing threat model components, threats, and security requirements in a database; associating each threat with a component; storing an indication of whether each security requirement is a...

CURRENT ASSIGNEES: THREATMODELER SOFTWARE INC
US10255439 | US PATENTS | 09-APR-2019

6. **Enterprise threat modeling**

A threat modeling application generates threats to a business application. The threat modeling application receives user input associated with business application data. The input may be received through an interface provided by the threat modeling application. Threats are automatically generated...

CURRENT ASSIGNEES: MICROSOFT TECH LICENSING LLC
US7891003 | US PATENTS | 15-FEB-2011

7. **Enterprise threat modeling**

A threat modeling application generates threats to a business application. The threat modeling application receives user input associated with business application data. The input may be received through an interface provided by the threat modeling application. Threats are automatically generated...

CURRENT ASSIGNEES: MICROSOFT TECH LICENSING LLC
US20070294766 | US APPLICATIONS | 20-DEC-2007

8. **Simulated attack generator for testing a cybersecurity system**

A computing device generates a simulated attack for testing a cybersecurity system. A user of a networked system of computers is selected. A user definition defines a normal network usage behavior of the selected user. A current simulation time is initialized. Netflow data is generated and written...

CURRENT ASSIGNEES: SAS INST INC
US10296748 | US PATENTS | 21-MAY-2019

Bibliographic Data

Application No: 16/542,263

Foreign Priority claimed: ○ Yes ◉ No

35 USC 119 (a-d) conditions met: ☐ Yes ☑ No ☐ Met After Allowance

Verified and Acknowledged: /THEODORE C PARSONS/ (Examiner's Signature) | (Initials)

Title: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

FILING or 371(c) DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
08/15/2019	726	2494	ThreatM010
RULE			

APPLICANTS

ThreatModeler Software Inc., Jersey City, NJ, UNITED STATES

INVENTORS

Anurag Agarwal Jersey City, NJ, UNITED STATES

CONTINUING DATA

This application is a CIP of 16228738 12/20/2018

16228738 is a CIP of 15922856 03/15/2018 PAT 10200399

15922856 is a CIP of 15888021 02/03/2018 PAT 10255439

15888021 has PRO of 62530295 07/10/2017

15888021 has PRO of 62527671 06/30/2017

15888021 has PRO of 62520954 06/16/2017

15888021 has PRO of 62507691 05/17/2017

FOREIGN APPLICATIONS

IF REQUIRED, FOREIGN LICENSE GRANTED**

08/26/2019

**** SMALL ENTITY ****

STATE OR COUNTRY

UNITED STATES

ADDRESS

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

FILING FEE RECEIVED

$2,855

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Nov 18, 2019 03:31:39 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	CTNF	11/18/2019	ThreatM010
	892	11/18/2019	ThreatM010
	1449	11/18/2019	ThreatM010
	1449	11/18/2019	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	THREATM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20170213037	A1	2017-07-27	Toledano et al.	
	2	20180255084	A1	2018-09-06	Kotinas et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16542263
	Filing Date	2019-08-15
	First Named Inventor	Anurag Agarwal
	Art Unit	2494
	Examiner Name	Theodore C. Parsons
	Attorney Docket Number	THREATM010

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1		

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	THREATM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☒ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2019-11-25
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Acknowledgement Receipt

EFS ID:	37857656
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	26-NOV-2019
Filing Date:	15-AUG-2019
Time Stamp:	00:11:51
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Information Disclosure Statement (IDS) Form (SB08)	IDS3.pdf	1034320 b8ea9c20e9614f5ca9dc6459370f89970c4d08c4	no	4

Warnings:

IriusRisk

Information:	
Total Files Size (in bytes):	1034320

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371(C) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
16/542,263	08/15/2019	Anurag Agarwal	ThreatM010

CONFIRMATION NO. 7756

145042
Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

PUBLICATION NOTICE

OC000000113122818

Title: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

Publication No.US-2019-0370477-A1
Publication Date:12/05/2019

NOTICE OF PUBLICATION OF APPLICATION

The above-identified application will be electronically published as a patent application publication pursuant to 37 CFR 1.211, et seq. The patent application publication number and publication date are set forth above.

The publication may be accessed through the USPTO's publically available Searchable Databases via the Internet at www.uspto.gov. The direct link to access the publication is currently http://www.uspto.gov/patft/.

The publication process established by the Office does not provide for mailing a copy of the publication to applicant. A copy of the publication may be obtained from the Office upon payment of the appropriate fee set forth in 37 CFR 1.19(a)(1). Orders for copies of patent application publications are handled by the USPTO's Public Records Division. The Public Records Division can be reached by telephone at (571) 272-3150 or (800) 972-6382, by facsimile at (571) 273-3250, by mail addressed to the United States Patent and Trademark Office, Public Records Division, Alexandria, VA 22313-1450 or via the Internet.

In addition, information on the status of the application, including the mailing date of Office actions and the dates of receipt of correspondence filed in the Office, may also be accessed via the Internet through the Patent Electronic Business Center at www.uspto.gov using the public side of the Patent Application Information and Retrieval (PAIR) system. The direct link to access this status information is currently https://portal.uspto.gov/pair/PublicPair. Prior to publication, such status information is confidential and may only be obtained by applicant using the private side of PAIR.

Further assistance in electronically accessing the publication, or about PAIR, is available by calling the Patent Electronic Business Center at 1-866-217-9197.

Office of Data Managment, Application Assistance Unit (571) 272-4000, or (571) 272-4200, or 1-888-786-0101

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Dec 06, 2019 03:29:28 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	NTC.PUB	12/05/2019	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	10262132	B2	2019-04-16	Reinecke et al.	
	2	7624448	B2	2009-11-24	Coffman	
	3	7260844	B1	2007-08-21	Tidwell et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20160162690	A1	2016-06-09	Reith et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

If you wish to add additional Foreign Patent Document citation information please click the Add button [Add]

NON-PATENT LITERATURE DOCUMENTS [Remove]

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1		

If you wish to add additional non-patent literature document citation information please click the Add button [Add]

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☒ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2020-01-01
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Acknowledgement Receipt

EFS ID:	38179054
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	01-JAN-2020
Filing Date:	15-AUG-2019
Time Stamp:	19:34:35
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Information Disclosure Statement (IDS) Form (SB08)	IDS4.pdf	1034421 a956c23273d070f6f5623a887c0070bc9fc037d3	no	4

Warnings:

Information:	
Total Files Size (in bytes):	1034421

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Doc Code: M865 or FAI.REQ.INTV

PTOL-413A (07-16)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Applicant Initiated Interview Request Form

Application No.: 16542263 First Named Applicant: Anurag Agarwal
Examiner: Theodore C. Parsons Art Unit: 2494 Status of Application: Pending

Tentative Participants:
(1) Theodore C. Parsons (2) Paul B. Johnson
(3) _____ (4) _____

Proposed Date of Interview: Tues. Jan. 7, 2020 **Proposed Time:** 2 _____ (◉AM ◉PM)

Type of Interview Requested:
(1) ☑ Telephonic (2) ☐ Personal (3) ☐ Video Conference

Exhibit To Be Shown or Demonstrated: ☐ YES ☑ NO
If yes, provide brief description: _____

Issues To Be Discussed

Issues (Rej., Obj., etc)	Claims/ Fig. #s	Prior Art	Discussed	Agreed	Not Agreed
(1) Drawing Objection	Fig. 1		☐	☐	☐
(2) 112	1-20		☐	☐	☐
(3)			☐	☐	☐
(4)			☐	☐	☐

☐ Continuation Sheet Attached ☑ Proposed Amendment or Arguments Attached
Brief Description of Arguments to be Presented: Proposed claim amendments and clarification of claim elements.

An interview was conducted on the above-identified application on _____

NOTE: This form should be completed and filed by applicant in advance of the interview (see MPEP § 713.01). If this form is signed by a registered practitioner not of record, the Office will accept this as an indication that he or she is authorized to conduct an interview on behalf of the principal (37 CFR 1.32(a)(3)) pursuant to 37 CFR 1.34. This is not a power of attorney to any above named practitioner. See the Instruction Sheet for this form, which is incorporated by reference. By signing this form, applicant or practitioner is certifying that he or she has read the Instruction Sheet. After the interview is conducted, applicant is advised to file a statement of the substance of this interview (37 CFR 1.133(b)) as soon as possible. This application will not be delayed from issue because of applicant's failure to submit a written record of this interview.

/Paul B. Johnson/
Applicant/Applicant's Representative Signature

_____
Examiner/SPE Signature

Paul B. Johnson
Typed/Printed Name of Applicant or Representative

801-719-5229
Applicant's/Applicant's Representative's Telephone Number

59375
Registration Number, if applicable

This collection of information is required by 37 CFR 1.133. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 24 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Proposed Discussion for Tues. 1/7/20 Interview

- **Requesting clarification on drawing objection. Counsel for applicant is unclear what element(s) in FIG. 1 need(s) correction. Counsel's best guesses as to the potential issues are:**
    - The globe-looking portion of element 120. This simply identifies server 120 as a web server but, if required, a box can be placed around the server and globe-like element and the element number 120 can point to the box.
    - The three portions of element 122. Each of the three servers depicted are intended to representatively illustrate a part of a server stack/rack but, if required, a box can be placed around these three depictions and the element number 122 can point to the box.

- **112 Rejections**
    - **Claim 1, 8, 16 use of "using visual representations of the threat model components correlated with the subset of third party diagram components"**
        - The "visual representations of the threat model components" are indeed separate entities from the "third party diagram components." It is believed this is made clear by the claim language as a whole, and that the manner of their correlation is clear. For example, Claim 1 states that the "threat model components" are stored in the one or more databases, that a "mapping file" coupled with the one or more databases correlates the threat model components with visual diagram components of a third party software application, and that visual representations of the threat model components are displayed on the relational diagram. The claim does not state that the visual diagram components of the third party software application are displayed on the relational diagram, and nor does the claim equate the "visual representations of the threat model components" with the "third party diagram components." They are accordingly separate entities, but the mapping file correlates them. The claim does not indicate whether the "visual representations of the threat model components" and the "third party diagram components" look the same, or identical, or different, and accordingly any of these scenarios could be the case (which counsel for applicant believes does not result in indefiniteness of the claim language, but relates to the breadth of the claim language). The same applies to claims 8 and 16.
        - It is believed no amendment is required but, if one is deemed necessary, counsel for applicant would appreciate ideas of how to best reword the language to clarify.
    - **All other 112 rejections are proposed to be overcome by the following proposed claim amendments which generally simply remove "VISIO" and "third party" from all claims, insert a bit of wording into claim 16, and correct the antecedent basis issues of claims 19-20.**

1. (Proposed Amendment) A threat modeling method, comprising:

providing one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

providing one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a ~~third-party~~ software application (hereinafter "~~third-party~~ diagram components"); and

in response to receiving one or more user inputs, using one or more user interfaces displayed on one or more computing devices communicatively coupled with the one or more databases:

using the one or more computing devices, reading a data file generated by the ~~third-party~~ software application, the data file comprising a subset of the ~~third-party~~ diagram components, the data file defining one or more relationships between the subset of ~~third-party~~ diagram components;

determining using the one or more computing devices, for the subset of ~~third-party~~ diagram components, correlated threat model components as defined in the one or more mapping files;

displaying on the one or more user interfaces a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of ~~third-party~~ diagram components, the relational diagram defining a threat model; and

generating, using the one or more computing devices, and displaying, on the one or more user interfaces, a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

2. (Original) The method of claim 1, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

3. (Original) The method of claim 1, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

4. (Original) The method of claim 1, wherein the data file comprises one of a VSD file, a VSDX file, a TMT file, an XML file, a JSON file, a PNG file, and a JPEG file.

5. (Proposed Amendment) The method of claim 1, wherein determining, for the subset of ~~third-party~~ diagram components, correlated threat model components, comprises first determining whether a user-defined name for one of the subset of ~~third-party~~ diagram components has a match in the one or more mapping files and, if no match is found, then determining whether a default title for the ~~third-party~~ diagram component, as defined by the ~~third-party~~ software application, has a match in the one or more mapping files.

6. (Proposed Amendment) The method of claim 5, further comprising, if no match is found for the default title, matching the ~~third-party~~ diagram component with a placeholder component and displaying the placeholder component on the relational diagram to indicate that no match was found.

7. (Original) The method of claim 1, further comprising determining, for the one or more relationships defined by the data file, correlated threat model components as defined in the one or more mapping files.

8. (Proposed Amendment) A threat modeling system, comprising:

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a ~~third-party~~ software application (hereinafter "~~third-party~~ diagram components");

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by one of the one or more computing devices, the data file generated by the ~~third-party~~ software application, the data file comprising a subset of the ~~third-party~~ diagram components, the data file defining one or more relationships between the subset of ~~third-party~~ diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of ~~third-party~~ diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of ~~third-party~~ diagram components, the relational diagram defining a threat model; and

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

9. (Original) The system of claim 8, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

10. (Original) The system of claim 8, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

11. (Original) The system of claim 8, wherein the data file comprises one of a VSD file, a VSDX file, a TMT File, an XML file, a JSON file, a PNG file, and a JPEG file.

12. (Proposed Amendment) The system of claim 8, wherein the one or more mapping files includes a file correlating user-defined names of the ~~third-party~~ diagram components with the threat model components.

13. (Proposed Amendment) The system of claim 8, wherein the one or more mapping files includes a file correlating default titles of the ~~third-party~~ diagram components with the threat model components.

14. (Proposed Amendment) The system of claim 8, wherein, for each of the subset of ~~third party~~ diagram components for which no correlated threat model component is found in the one or more mapping files, the diagram interface displays a placeholder component indicative of a match not being found.

15. (Original) The system of claim 8, further comprising one or more mapping interfaces displayed on the one or more computing devices and configured to, in response to receiving user input, modify the one or more mapping files.

16. (Proposed Amendment) A threat modeling system, comprising:

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with ~~VISIO~~ visual diagram components of a software application (hereinafter "diagram components");

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by the one or more computing devices, the data file comprising a subset of the ~~VISIO~~ diagram components, the data file defining one or more relationships between the subset of ~~VISIO~~ diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of ~~VISIO~~ diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of a computing network, using visual representations of the threat model components correlated with the subset of ~~VISIO~~ diagram components, the relational diagram defining a threat model; and

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

17. (Original) The system of claim 16, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

18. (Original) The system of claim 16, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

19. (Proposed Amendment) The system of claim 16, wherein the one or more mapping files includes a file correlating user-defined names of the ~~third party~~ diagram components with the threat model components.

20. (Proposed Amendment) The system of claim 16, wherein the one or more mapping files includes a file correlating default titles of the ~~third party~~ diagram components with the threat model components.

Electronic Acknowledgement Receipt

EFS ID:	38190007
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	02-JAN-2020
Filing Date:	15-AUG-2019
Time Stamp:	18:11:23
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Letter Requesting Interview with Examiner	PTOL413A.pdf	404800 b8a94bd78cc341a7149acd8ee0f92b818a3d798e	no	8

Warnings:

IriusRisk

Information:	
Total Files Size (in bytes):	404800

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20010027388	A1	2001-10-04	Beverina et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16542263
	Filing Date	2019-08-15
	First Named Inventor	Anurag Agarwal
	Art Unit	2494
	Examiner Name	Theodore C. Parsons
	Attorney Docket Number	ThreatM010

	1	Roy, Maurer. "Top Database Security Threats and How to Mitigate Them." July 30, 2015. Retrieved from "https://www.shrm.org/resouresandtools/hr-topics/risk-management/pages/top-database-security-threats.aspx" (Year: 2015).
	2	A. Amini, N. Jamil, A.R. Ahmad and M.R. Z'aba, 2015. Threat Modeling Approaches for Securing Cloud Computing. Journal of Applied Sciences, 15: 953-967. Retrieved from "https://scialert.net/fulltextmobile/?doi=jas.2015.953.967" (Year: 2015).

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☒ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2020-01-27
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Acknowledgement Receipt

EFS ID:	38410299
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	27-JAN-2020
Filing Date:	15-AUG-2019
Time Stamp:	16:57:01
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Non Patent Literature	NPL07.pdf	134210 f5e2ca8cb9f2c6a596108dd199d05dd89779947b	no	3

Warnings:

IriusRisk

Information:					
2	Non Patent Literature	NPL08.pdf	1542821 66b4c0c674846a00f8c3fe6646d24d1eee9b19d6	no	16
Warnings:					
Information:					
3	Information Disclosure Statement (IDS) Form (SB08)	IDS5.pdf	1034648 4584dd652913bdfb078448e738dac326ecdf36a2	no	4
Warnings:					
Information:					
			Total Files Size (in bytes):	2711679	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

<u>New Applications Under 35 U.S.C. 111</u>
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.
<u>National Stage of an International Application under 35 U.S.C. 371</u>
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.
<u>New International Application Filed with the USPTO as a Receiving Office</u>
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Top Database Security Threats and How to Mitigate Them

By Roy Maurer
July 30, 2015

Every day, hackers unleash attacks designed to steal confidential data, and an organization's database servers are often the primary targets of these attacks. Databases are one of the most compromised assets according to the 2015 Verizon Data Breach Investigations Report (http://www.verizonenterprise.com/DBIR/).

"The reason databases are targeted so often is quite simple—they are at the heart of any organization, storing customer records and other confidential business data," said Morgan Gerhart, vice president of product marketing at cybersecurity firm Imperva (http://www.imperva.com/). Organizations are not protecting these crucial assets well enough, he added.

"When hackers and malicious insiders gain access to sensitive data, they can quickly extract value, inflict damage or impact business operations. In addition to financial loss or reputation damage, breaches can result in regulatory violations, fines and legal fees," he said.

Top Database Threats

The threats identified over the last couple of years are the same that continue to plague businesses today, according to Gerhart. The most common database threats include:

***Excessive privileges.** When workers are granted default database privileges that exceed the requirements of their job functions, these privileges can be abused, Gerhart said. "For example, a bank employee whose job requires the ability to change only account holder contact information may take advantage of excessive database privileges and increase the account balance of a colleague's savings account." Further, some companies fail to update access privileges for employees who change roles within an organization or leave altogether.

***Legitimate privilege abuse.** Users may abuse legitimate database privileges for unauthorized purposes, Gerhart said.

***Database injection attacks.** The two major types of database injection attacks are SQL injections that target traditional database systems and NoSQL injections that target "big data" platforms. "A crucial point to realize here is that, although it is technically true that big data solutions are impervious to SQL injection attacks because they don't actually use any SQL-based technology, they are, in fact, still susceptible to the same fundamental class of attack," Gerhart said. "In both types, a successful input injection attack can give an attacker unrestricted access to an entire database."

***Malware.** A perennial threat, malware is used to steal sensitive data via legitimate users using infected devices.

***Storage media exposure.** Backup storage media is often completely unprotected from attack, Gerhart said. "As a result, numerous security breaches have involved the theft of database backup disks and tapes. Furthermore, failure to audit and monitor the activities of administrators who have low-level access to sensitive information can put your data at risk. Taking the appropriate measures to protect backup copies of sensitive data and monitor your most highly privileged users is not only a data security best practice, but also mandated by many regulations," he said.

Feedback

We use cookies to improve your browsing experience on our website, and we use our own, as well as third-party cookies, to display advertising for products and services we believe may be of interest to you. By closing this message or continuing to browse this site, you agree to the use of cookies in accordance with our **Cookie Policy**.

×

***Exploitation of vulnerable databases.** It generally takes organizations months to patch databases, during which time they remain vulnerable. Attackers know how to exploit unpatched databases or databases that still have default accounts and configuration parameters. "Unfortunately, organizations often struggle to stay on top of maintaining database configurations even when patches are available. Typical issues include high workloads and mounting backlogs for the associated database administrators, complex and time-consuming requirements for testing patches, and the challenge of finding a maintenance window to take down and work on what is often classified as a business-critical system," Gerhart said.

***Unmanaged sensitive data.** Many companies struggle to maintain an accurate inventory of their databases and the critical data objects contained within them. "Forgotten databases may contain sensitive information, and new databases can emerge without visibility to the security team. Sensitive data in these databases will be exposed to threats if the required controls and permissions are not implemented," he said.

***The human factor.** The root cause for 30 percent of data breach incidents is human negligence, according to the Ponemon Institute Cost of Data Breach Study (http://www-03.ibm.com/security/data-breach/). "Often this is due to the lack of expertise required to implement security controls, enforce policies or conduct incident response processes," Gerhart said.

Multilayered Security Solutions

A defensive matrix of best practices and internal controls is needed to properly protect databases, according to Imperva. This matrix includes:

- Assessing for any database vulnerabilities, identifying compromised endpoints and classifying sensitive data.
- Managing user access rights and removing excessive privileges and dormant users.
- Monitoring all database access activity and usage patterns in real time to detect data leakage, unauthorized SQL and big data transactions, and protocol and system attacks.
- Blocking malicious web requests.
- Automating auditing with a database auditing and protection platform.
- Archiving external data and encrypting databases.
- Training employees on risk-mitigation techniques including how to recognize common cyberthreats such as a spear-phishing attack, best practices around Internet and e-mail usage, and password management. "Failure to enforce training and create a security-conscious work culture increases the chances of a security breach," Gerhart said.

Roy Maurer is an online editor/manager for SHRM.

Follow him @SHRMRoy (https://twitter.com/SHRMRoy)

Feedback

HR DAILY NEWSLETTER

News, trends and analysis, as well as breaking news alerts, to help HR professionals do their jobs better each business day.

Email Address

We use cookies to improve your browsing experience on our website, and we use our own, as well as third-party cookies, to display advertising for products and services we believe may be of interest to you. By closing the message or continuing to browse this site, you agree to the use of cookies in accordance with our **Cookie Policy**.

×

CONTACT US (WWW.SHRM.ORG/ABOUT-SHRM/PAGES/CONTACT-US.ASPX) | 800.283.SHRM (7476)

© 2020 SHRM. All Rights Reserved

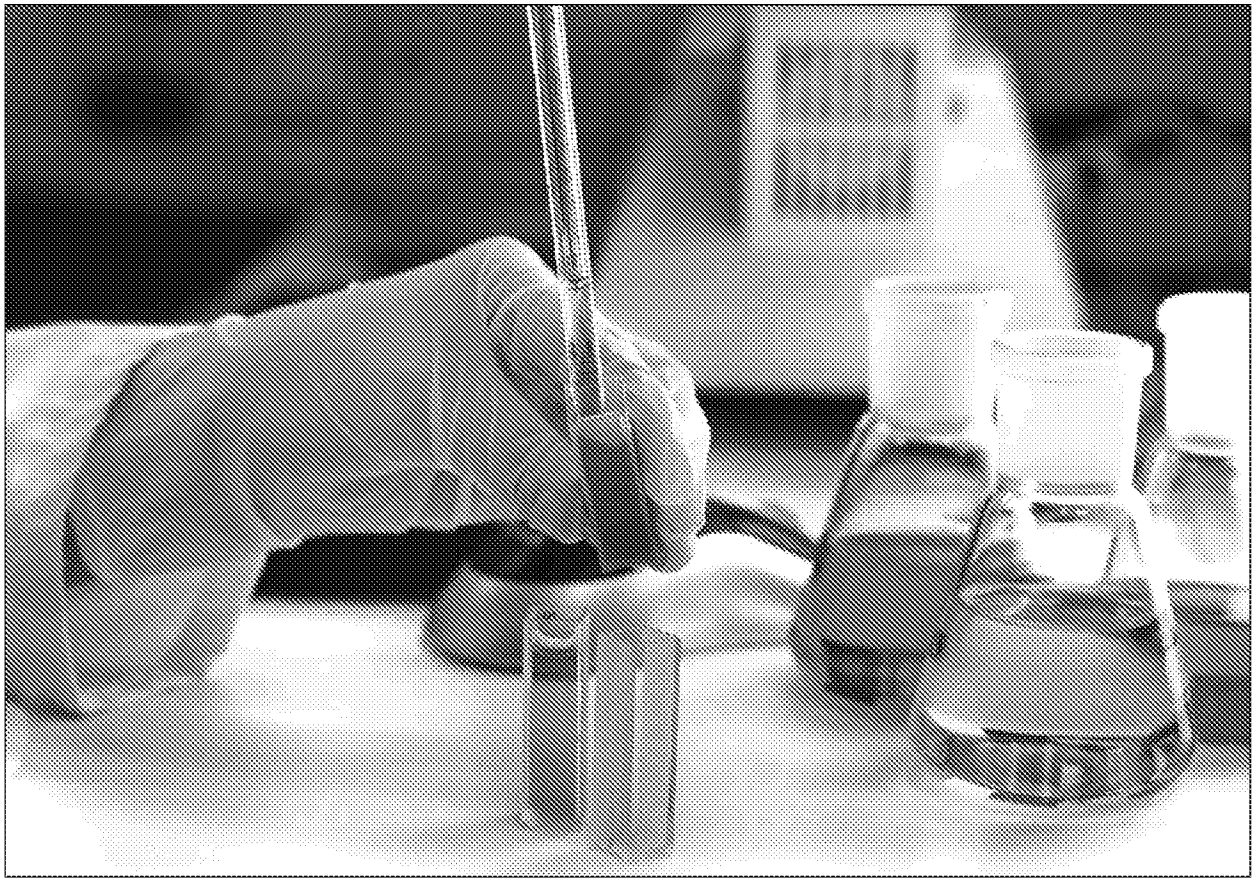
SHRM provides content as a service to its readers and members. It does not offer legal advice, and cannot guarantee the accuracy or suitability of its content for a particular purpose.

Disclaimer (www.shrm.org/about-shrm/Pages/Terms-of-Use.aspx#Disclaimer)

Feedback

We use cookies to improve your browsing experience on our website, and we use our own, as well as third-party cookies, to display advertising for products and services we believe may be of interest to you. By closing the message or continuing to browse this site, you agree to the use of cookies in accordance with our **Cookie Policy**.

×



Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

IriusRisk
Ex. 1002 - Page 480

Journal of Applied Sciences 15 (7): 953-967, 2015
ISSN 1812-5654
© 2015 Asian Network for Scientific Information

RESEARCH ARTICLE OPEN ACCESS
DOI: 10.3923/jas.2015.953.967

Threat Modeling Approaches for Securing Cloud Computing

[1]A. Amini, [1]N. Jamil, [1]A.R. Ahmad and [2]M.R. Z'aba
[1]College of Information Technology, Universiti Tenaga Nasional, Selangor, Malaysia
[2]MIMOS Berhad, Technology Park Malaysia, Kuala Lumpur, Malaysia

ARTICLE INFO
Article History:
Received: December 29, 2014
Accepted: April 25, 2015

Corresponding Author:
A. Amini
College of Information Technology,
Universiti Tenaga Nasional,
Selangor, Malaysia

ABSTRACT
The cloud computing delivers services of processing data by using parallel and high computational processors virtually and remotely. This is very useful and beneficial to enterprises that have limitations in terms of processing resources, data storages and man power. However, one of the main challenge in cloud computing is data security. Supposed each user has an access to the domain based on the privilege given by the server. However, an attacker always has the opportunity to bypass the privilege by exploring and exploiting every vulnerability and threat found in the cloud computing environment. A systematic approach to identify vulnerabilities and threats is thus very important to ensure only authorized party is allowed to access the data. In this study, several approaches of vulnerability and threat modeling are reviewed and found that none of them is suitable for the cloud computing environment. Therefore, authors presented a dynamic model of identifying vulnerabilities and threats in the cloud computing environment. The proposed model enables the organization to systematically identify vulnerability and threats and analyze the security risk when they use cloud computing services.

Key words: Cloud computing, data security, threat modeling, vulnerability

INTRODUCTION

Cloud computing as a new distributed computing paradigm delivers on-demand services by decreasing capital investment on infrastructure and maximizing the usage of available resources. Cloud technology brings application and infrastructure services mobility and physical/hardware platform independency to the existing distributed computing and networking application (Demchenko *et al*., 2012). By growing cloud computing and making compute resources available for consumers, IT industry becomes more flexible, scalable, capable and cost effective for developing and hosting applications. However, by adopting this new powerful emerging system, security has been cited as the most essential and critical issue. For understanding security issues, threats, vulnerabilities and risks should be analysed as different factors that influenced cloud computing security.

Threat modeling as a systematic and comprehensive analysis of threats and vulnerabilities is needed to provide confidentiality, integrity and availability for deploying cloud computing security. Threat modeling collects background information required in the form of usage scenario, external dependencies, implementation assumption, internal and external security implementation detail (Malik *et al*., 2008). A number of threat modeling techniques have been developed for evaluating and analyzing the threats and vulnerabilities such as:

- Microsoft threat modeling is a model to offer an efficient process that includes five logical steps from classifying assets to addressing and classifying threats and vulnerabilities
- Microsoft's Threat Analysis and Modeling (TAM) (Malik *et al*., 2008) is a model based on business objectives that have to be met by application. The TAM tools have been used to generate and classify threats by measuring their harm effects on system's components
- Practical Threat Analysis (PTA) (McRee, 2008) defines an effective risk mitigation plan for specific system architecture to obtain the value of system's assets
- Threat model framework and methodology for Personal Networks (PNs) (Jehangir and de Groot, 2012). This threat model by analyzing personal network gives a good view of system's assets and their values. Determining all assets by UML diagrams gives opportunity to protect data and network functions against any threats

IriusRisk
Ex. 1002 - Page 481

- Threat modeling in pervasive computing paradigm by developing cloud computing, each user is faced by different security domains with multi identify. The above model presents a new threat modeling to incorporate the problem of pervasive computing environment

Unfortunately by growing cloud as a large scalable distributed computing, all above methodologies do not incorporate problems, so new approaches for threat modeling have to be evolved to solve problems of pervasive computing security.

This study presents a methodology of the threat model to deploy a secure computing environment by showing threats and vulnerabilities in the cloud computing and determining security solutions. The proposed methodology to build threat model consists of several steps, those after considering all steps the result represents the final threat model for cloud computing.

MATERIALS AND METHODS

Security issues of cloud computing: Before, a closer look is taken at cloud-specific threats, vulnerabilities and solutions, we define vulnerability and threat as follows:

- Threat is harm or unauthorized access that might occur due to vulnerability and destroy organization assets, organization operations or system information
- Vulnerability is any weakness in information system, system security procedures, internal controls or implementation that could be exploited or triggered by a threat resources (Bertino *et al*., 2010)

Cloud computing threats: Cloud Security Alliance (CSA) (Soares *et al*., 2013) as a security assurance provider released the security guidance for critical areas in cloud computing for managing risks and understanding security threats. The most of significant threats that are related to on-demand nature of cloud computing are categorized as below:

- **Data lose or leakage (T1):** This threat is measured as the most critical and terrifying threat for businesses and consumers. Any data deletion by service provider or baleful accident such as fire can lead to lose the consumer's data
- **Account or service hijacking (T2):** This weakness allows attackers to steal credentials and access to critical areas of cloud computing services. The organization should prohibit the sharing of credentials between different services and users and use strong authentication techniques
- **Insecure interface (T3):** Cloud computing's customers use Application Programming Interface (API) or software interfaces to interact and manage cloud services. Authentication, access control and monitoring technologies for APIs protect computing resources from malicious attacks
- **Denial of service (T4):** Distributed Denial of Service (DDoS) is the major security threat to availability when it comes to increase reliability of organizations on public cloud services (Lohman, 2011). On the other hand, this attack prevents users from accessing their data or applications and there is no way to reach their destination. Cloud service providers need to be sure about availability protection and customers need to be sure about the level of availability protection within a provider
- **Malicious insider (T5):** The system has been damaged by authorized employee, business partner or administrator that has access to a network or resources. This dangerous threat affects the confidentiality, integrity or availability of business information
- **Data breaches (T6):** One of the worse situations for each organization is unauthorized access or illegal viewing data by competitors. Data encryption can reduce the risk of this threat, but should be careful about encryption key because if you lose it, you will lose your data as well
- **Abuse of cloud services (T7):** Cloud computing providers do not enforce any strong registration process and any user with a valid credit card can register to receive cloud services (Hamza *et al*., 2013). Integrating of registration faint and weak fraud detection allow attacker to leverage data by offensive cloud models same as PaaS and IaaS
- **Insufficient due diligence (T8):** The cost reduction, access to pool of resources and improving security are the most important interesting factors for organization to rush cloud computing. Without understanding of Cloud Service Provider (CSP) environment, mismatched expectation has been created as a critical issue on security control of cloud computing. However, for sufficient qualification of resources, organizations have to understand the service provider offerings and risks
- **Insecure VM migration (T9):** By migrating different VMs during hybrid and federated clouds, attackers can access data illegally and transfer VM to untrusted host (Hashizume *et al*., 2013). Virtualization as the main component of IaaS is the main goal to be targeted by attackers. Trusted cloud computing and encryption technology protect data resources from insecure VM migration

Threat of shared vulnerabilities exists in all models (IaaS, PaaS, SaaS) because underlying components that deploy infrastructure, platforms and applications does not offer strong isolation between cloud computing models.

Cloud computing vulnerability: Vulnerability is a probability of an asset that will be unable to resist the action of a threat agent (The Open Group, 2009). By moving organization's critical data and applications to cloud computing, different significant vulnerabilities should be considered based on CC's

IriusRisk
Ex. 1002 - Page 482

technologies, essential cloud characteristics, known security controls and state-of-the-art cloud offerings (Grobauer *et al.*, 2011). In this study, indicators were discussed as cloud specific vulnerabilities:

- **Session riding (V1):** Session riding refers to send command to web application by hackers to gain unauthorized access for the information or use web service weaknesses for giving the chance to hackers to do malicious activities same as deleting of user data or sending spam to a network via internet
- **Virtual machine escape (V2):** This vulnerability allows attacker runs code on a VM that let operating system to break out and interact directly with the hypervisor to access host operating system and other virtual machines (Rouse, 2009). For detecting and preventing system should detect malicious activity at VM level
- **Obsolete cryptography (V3):** Developing not enough strong encryption or no encryption at all allows attacker to decode encrypted data. To protect system from this vulnerability, user should be sure the true data is encrypted, use proper key storage and develop a good algorithm
- **Unauthorized access to management interface (V4):** The cloud management interface has access to cloud service users to manage on-demand services. An unauthorized access enables attackers to gain total control of users and applications
- **Internet protocol (V5):** The lack of authentication methods that is not a part of the base protocol design, allows attackers to inject their malicious traffic to network. On the other hand, the IP protocol or related protocols same as UDP and TCP are vulnerable to different type of Denial of Service (DoS) attacks, including session hijacking and cash poisoning
- **Data recovery (V6):** Cloud computing allows resources to be allocated or reallocated by different users. This elastic characteristic could lead to data stolen, data breaches and other security threats. The most of organizations use third party vendors to recover data so they should consider security risk of handling data with outside company and ensure proper security vetting of the service provider (Kenyon, 2010)
- **Metering and billing (V7):** Cloud computing meters and measures services such as storage, user account and processing are used to optimize service delivery. Applicable vulnerabilities contain metering and billing data treatment and billing elusion (Rai and Bunkar, 2014)
- **Vendor lock-in (V8):** Vendor lock-in is the situation that cloud's user is dependent to a single vendor and is unable to deal with another provider without substantial and inconvenience. The lack of the standards is the main reason that users cannot transfer easily from one provider to another.

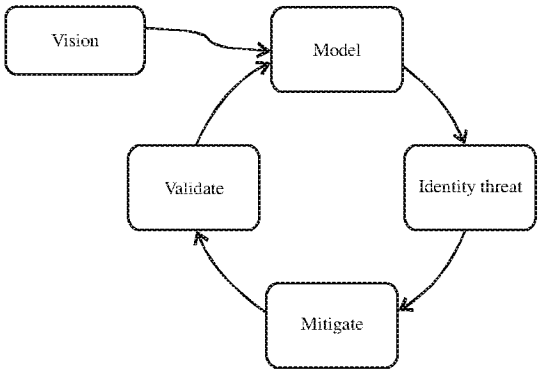


Fig. 1: SDLC process diagram

Existing threat modeling frameworks: In this section, existing threat models are discussed and new approach has been evolved to satisfy and fulfill cloud computing as a pervasive computing paradigm.

Microsoft threat modeling: Threat modeling as a core of the Microsoft Security Development Lifecycle (SDL) (Fig. 1), proposes a model to identify and mitigate threats and vulnerabilities by reducing the total cost of development process. This methodology offers a good process cycle by following five logical steps, start from classifying assets, system overview by creating Data Follow Diagram (DFD), modeling and identifying threats by using STRIDE[1], addressing issues and finally ranking the threats by DREAD[2].

Vulnerabilities as an important and hazardous security risk in cloud computing have to be addressed by threat model but Microsoft threat modeling cannot model vulnerabilities. Another important challenge to adopt this popular model with cloud computing is predefining a set of threats as STRIDE. In addition, STRIDE defines a static method to identify threats that cannot be used in dynamic computing environment. For instance, denial of service as one of the top ranked threat in 2013 was not listed as cloud computing threats in 2010. However, defining new threats which are more critical than expected STRIDE, should be intended in our new threat model.

Microsoft's Threat Analysis and Modeling (TAM): Threat Analysis and Modeling (TAM) proposes a model to identify threats and facilitate system to define a security strategy (Fig. 2) (Malik *et al.*, 2008). In the first step of this model, threat modeling has been addressed on the base of business objectives. Business objectives involve business needs and problems that have to be fulfilled by the application. Components as the main part of application's architecture, technology of creating components, user roles, trust level, authentication mechanisms and external dependencies are define as a second step. In the third step, TAM's tools generate

[1]STRIDE: Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of Privilege
[2]DREAD: Damage potential, Reproducibility, Exploitability, Affected users, Discoverability

IriusRisk
Ex. 1002 - Page 483

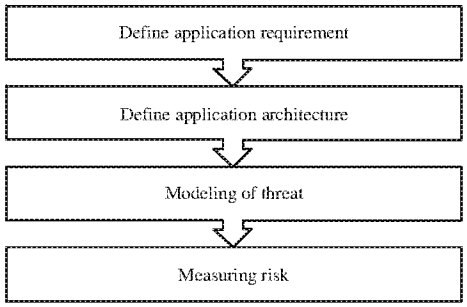


Fig. 2: TAN process diagram

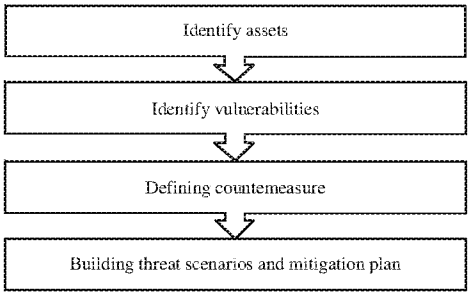


Fig. 3: PTA process diagram

and classify threats into confidentiality, integrity and availability. In the last step each threat is measured on the base of business needs.

The proposed threat modeling for computing environment unlike TAM has to determine assets which could be damaged by threats and need protection. The lack of identifying vulnerabilities is another issue of Microsoft threat analysis and modeling. On the other hand, determining and ranking threats are done as well in this model but this detecting and ranking vulnerability that are damaged by threats are not deployed.

Practical threat analysis: The Practical Threat Analysis (PTA) proposed a model to identify system vulnerabilities, map system assets, assess the risk of the threats and define an effective risk mitigation plan for a specific system architecture, functionality and configuration (McRee, 2008). The PTA by providing free tools for threat modeling allows provider to obtain system's assets values and level of damage that is caused by the attackers. The main functionality of PTA is defining a risk mitigation plan on the base of identify vulnerabilities and assets and assess the risk of threats. PTA technologies present PTA threat modeling and risk assessment that is shown in Fig. 3. Identifying assets and their financial values is done in the first step. The second step represents system's vulnerabilities on the base of architecture, functionality and different type of users. The effect of cost is calculated on the base of implementation cost. In the last step, threat scenario has been built to identify various threat and level of damage.

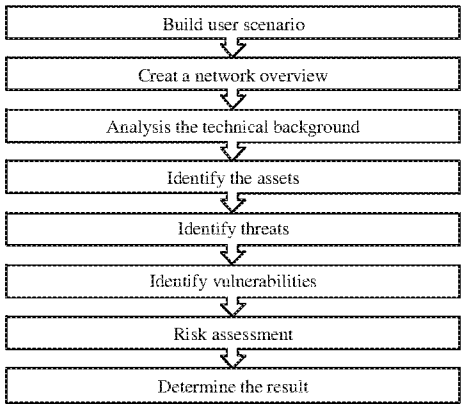


Fig. 4: Threat model for PNs

No risk evaluation method or methodology to calculate vulnerability's cost is an important issue. Actually, the cost of replacement, restoration or system down times is required to calculate the cost of vulnerabilities and thereby it gives idea to avoid system from different threats. The second disadvantage of this model is unranking vulnerability sizes. In the pervasive computing environment all known and credible vulnerabilities should be addressed. Anyway, vulnerabilities have to be ranked because known vulnerabilities can be eliminated and for unknown, some solutions should be deliberated. In PTA model, threats are ranked only by their financial values not by technical, functional, strategic or reputational values. Cloud computing as a complex multi-tenant environment, provides ability for users to be a member of different domains at the same times. In this situation, different factors and levels of security on the base of user's role should be considered to address threats not only financial values.

Threat model framework and methodology for Personal Networks (PNs): The PN as an intelligent and user-centric network consists of devices belonging to user for personal purpose, develop new application, telecommunications, entertainment and improve services (Jehangir and de Groot, 2012). Figure 4 shows primary steps of proposed threat model for PNs. Each of these main steps consists of some sub-steps (Prasad, 2007).

In the first step, the use case has been derived to describe everything in network from user's view. In the next step, the network requirements have been retrieved from use case diagrams and also network architecture, environment and technologies have been explained. In the third step, UML sequence diagram has been used to clarify data flows and determine which actors and devices are involved. The assets or everything that could be damaged should be determined and protected from harmful attacks in the next step. The fifth step involves all threats and expected result of them. The main outcome is determining threat scenario to extract fundamental threats and likelihood. After discovering threats, is possible to

IriusRisk
Ex. 1002 - Page 484

see and understand vulnerabilities that are affected by terrible threats. In the seventh step, highest risks have been determined by threat and vulnerabilities profile. The final step consists of the reports that involve threats and vulnerabilities rank on the base of risk and size.

This framework gives a good view of system by describing all network details, covering all functionalities and deploying use case. As mentioned before, in cloud computing user is faced by many services and domains that may have different and unknown vulnerabilities so proposed framework unlike PN's threat model need to be updated by new threats and vulnerability profiles.

Threat modeling in pervasive computing paradigm: By cloud computing as an on-demand service, provides reliability to user's daily life. Each user has multi identifies for different security domain so implementing security schema is a big challenge for cloud computing. This model (Fig. 5) presented as a new threat modeling to incorporate the problem of pervasive computing environment (Malik *et al*., 2008).

In the first step all user's roles need to be established with their service usage and authentication mechanisms. The use case scenario and diagrams can be used in this method to identify users and their level of access to different services. In the computing environment, each user can be a part of various domains with different roles. Identifying security domains that is mentioned in step two, presents the way that user interacts with applications inside the domain. Various type of user same as authorized user, admin and unregistered user have different trust levels. Identifying trust level in the third level, assists user to access the resources depending upon its trust level. In the next step, all vulnerabilities need to be detected. Known vulnerabilities can be eliminated and unknown vulnerabilities should be consider to protect system from harm attacks. Risk evaluation as a forth step, presents fair idea to avoid computing environment from threats. Threats have been qualified on the base of cost expectancy that is involved restoration and replacement cost. In the last step, new vulnerabilities and threats have been identified and post them to forth step for keeping system secure.

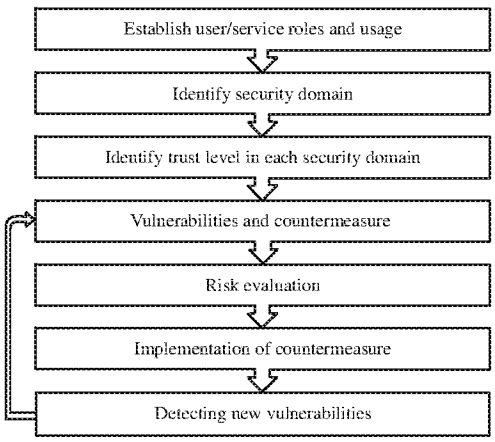


Fig. 5: Threat modeling process for pervasive computing

Today, by developing small and medium enterprises and growing the number of assets, the traditional assets tracking methods cannot be used anymore. In proposed framework unlike above model, cloud based assets management has been developed rather than traditional infrastructure to manage data, software, hardware and etc.

Before have a closer look at proposed model, we illustrate all details of discussed models in Table 1 to analyze and compare existing models by proposed model. Table 1 is based on characteristics of threat model to be adopted by cloud computing. In the next part of this study, each characteristic of new threat modeling schema was discussed to identify and rank threats and vulnerabilities.

RESULTS AND DISCUSSION

Proposed threat model: The proposed model for cloud computing security is shown in Fig. 6. This model consists of four main steps that each step involves some sub steps. The first step is identifying assets and clarifying who or what has access to assets. Trust represented by the term reliability that is using for multiple redundant web services to qualified trust between user and service provider or between different providers. Further, the second step defines capability of provider to provide user requirements literally. Identifying unique threats and addressing them by developing suitable

Table 1: Comparison of existing threat models by proposed model

	Existing threat models					
Characteristics of threat modeling	Microsoft threat modeling	TAM	PTA	Personal network threat modeling	Pervasive computing threat modeling	Proposed threat model
Identifying and classifying assets	✓		✓	✓	✓	✓
Establishing user's roles						✓
Identifying security domains					✓	✓
Estimating trustworthiness					✓	✓
Scanning domain security						✓
Identifying threat	✓	✓		✓	✓	✓
Identifying vulnerability				✓	✓	✓
Implementing countermeasure					✓	✓
Ranking and measuring threats	✓	✓		✓	✓	✓
Ranking and measuring vulnerabilities				✓		✓
Defining new assets, threats or vulnerabilities					✓	✓

IriusRisk
Ex. 1002 - Page 485

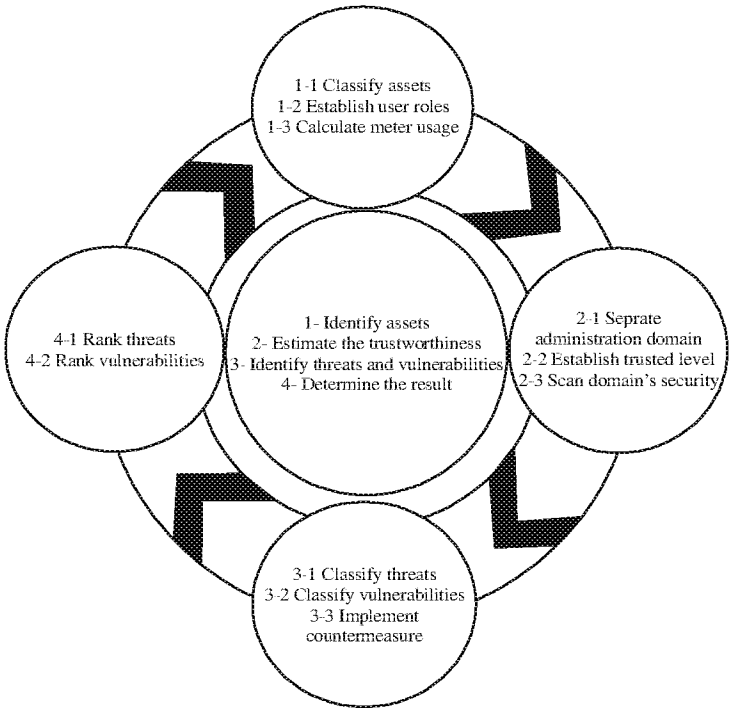


Fig. 6: Proposed threat modeling framework for cloud computing

countermeasure has been demonstrated in the third step. The main goal of this step is detecting and identifying new subsequently threats to improve security. The final step, presents system rating to identify more dangerous and effected threats and vulnerabilities.

Identify assets: The IT asset is data, software or hardware that is owned by company and used for business activities. The organizations have to be ensured these assets are accessed by authorized users and they are configured with latest security technologies to protect against security threats and vulnerabilities. The most unauthorized access or effective change during configuration management is being made by unofficial user to system. Therefore, knowing enterprises machines and their location is required to protect system from malicious attacks. Today, organizations use feasible and efficient asset management tools (SAM) to analyze their own data, software and hardware. Anyway, these tools by optimizing and managing IT assets help organization to identify what they have, control cost and risk and improve security. For discovering and monitoring assets and identifying users' roles in our suggested framework, some substeps (Fig. 6) have been proposed.

Classify assets: To implement threat modeling framework is necessary to define assets that can be damaged and need protection. The defined assets should be categorized on the base of their value that is measured by loss of availability, confidentiality and integrity types. Moreover, assets classification allows companies to store and manage their data by sensitivity and determine risks. For instance, a company has computing resource (memory, storage or CPU), encryption key, applications and customers' personal information that can be classified by different values, damage costs and trust levels. Today, most of organizations are aware of data classification significant. Every enterprise has different assets that have to be managed and classified successfully so can not specify some special and unique assets classifications for all businesses. The below mentioned and discussed assets categorizations are the most important data that by protecting them, each organization meet its final targets.

Private information: By developing social networks, personal data can be exchanged between various enterprises and users for sharing and interacting more experiences. However, private data management is a key to protect data from malicious threats. The most significance threats to offend private information have been shown in Fig. 7. The major challenges and harms for private information are collecting and extracting information by unauthorized access, using fake data for registration and lacking of controls and standards. By deploying private information manager applications more private and confidential data has to be kept secure to help enterprises to meet the acceptable level of confidentiality, integrity and availability.

Financial assets: In recent years, by developing dramatically web service technologies, more and more financial and e-commerce values are shared and accessed by users and enterprises.

IriusRisk
Ex. 1002 - Page 486

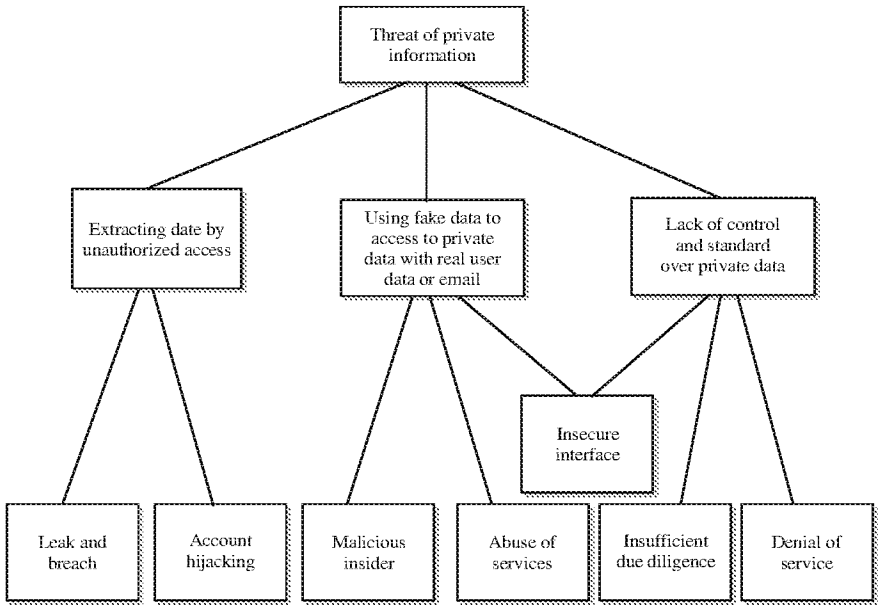


Fig. 7: Significance threats to offend privacy data

Unfortunately, this opportunity becomes the primary target for attackers to access financial assets by deploying cyber threats. For instance, Phishing and DDoS attacks are the major cyber threats that steal user financial assets same as password, bank account detail, credit card number and other confidential enterprises information. However, understanding and analyzing financial assets during threat modeling process allows enterprises to protect data from data stolen through on-line services.

Computing and network assets: This asset is an entity that is capable of creating, sustaining or destroying value at any stage in its life-cycle (Amadi-Echendu, 2004). Analyzing actual network devices or tangible assets (data center, server, desktop computer, router, switch and etc.) and providing security controls is necessary to protect system from malicious insider threats or attacks that are targeted physical assets for terrible damages. The network assets management as an on-demand service allows organization to manage risks and threats, reduce cost and increase the reliability of system.

Application assets: Applications are usually used to store, transfer and process of valuable information. For this reason, enterprises have to be sure about implementing and developing application securely. Application threat modeling by looking application as attacker's view helps enterprises to manage security risks and threats. Each application utilizes different data that allow organization to determine risk and level of expected security. For example an application consists of financial, personal information or users' account numbers. Eventually, analyzing application assets in threat modeling approach can improve security and reduce costs. Human error, malfunction, spoofing, spamming and malware as the major threats should be detected to protect other assets from malicious attacks.

Establish user's roles: Cloud computing by providing data ubiquity, flexibility of access and relicense, gives opportunity to users for accessing the required service anywhere, anytime and pay per their usage (Ryan, 2013). Each user might assume different roles in various domains, for instance a user might be an administrator with high level privilege in one domain and be a user with low level privilege in another domain. However, establishing the user roles with authentication, authorization and access control mechanisms is necessary for improving data confidentiality, integrity and availability. Nowadays, by progressing online services, organizations same as bank and insurance industry attract huge number of users. For using this excellent opportunity organization have to determine their users' roles to control accesses and improve security. In such enterprises same as bank is impossible to assign users to roles manually so some techniques like as Role Based Access Control (RBAC) assigns organization roles to user, defines privileges and supports security policies (Takabi *et al.*, 2007). Furthermore, domain administrator should assign or remove roles for user or group of users on the base of requested service. For example a developer can be assigned as cloud database developer role to develop and deploy application or he/she can be assigned as a cloud user to implement an application.

By migrating more and more applications and services to the cloud as a centralized data center, CC providers are faced by more access control problems. In fact, cloud computing opportunities same as diversity and dynamicity make

IriusRisk
Ex. 1002 - Page 487

authentication and authorization processes more complex because each cloud user has different access permissions in the same cloud. By the same token, using multiple services on the same cloud by the same user, the authentication and login time increase and also managing more identity process by user make system unsecure. Organization's purpose is managing different roles and assigning them to user to optimize their access control system. Eventually, some access control method same as Cloud Optimized RBAC (coRBAC) (Tianyi *et al.*, 2011) merge distributed authentication services, manage organization role and user in internal network, provide individual authentication service and improve efficiency of access control systems.

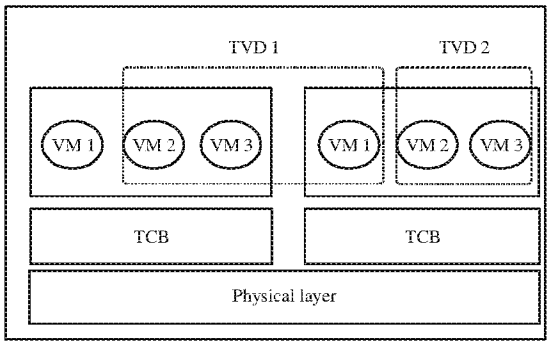


Fig. 8: TVD architecture

Meter usage: Metering and billing are two important issues that should be addressed during threat modeling framework. In computing area storage and network services have to be metered by duration and amount of used. Service Level Agreement (SLA) is a fixed cost or static method that is used as a metering mechanism. Particularly, by developing multi-tenancy and on-demand services, providers have to be used dynamic pay-per-use mechanisms in cloud computing to make bill based on dynamic tariff plan and real-time pricing (Narayan *et al.*, 2012). Furthermore, dynamic metering of cloud services is to prevent wastage of compute resources and improve overall utilization of the cloud infrastructure.

The metering application improves utilization of the infrastructures, cost and management of computing resources by developing dynamic pay-per-use pricing model. These great opportunities are the major interesting factors for organizations to adopt their resources and services to cloud computing. Eventually, attackers are interested to attack system by insufficient due diligence threat and metering and billing vulnerabilities to create the critical issues on security control of cloud computing. Due to the great importance of billing and metering, we proposed to use dynamic mapping pricing and effective billing mechanisms for cloud services based on the current load on the using of cloud infrastructures.

Estimate the trustworthiness: Trust is reliability in distributed systems, among two entities that relay on their credibility to deploy trustworthiness and improve security. Nowadays, trust plays an important role to integrate heterogeneous environments for estimating trustworthiness by confidentiality, integrity, availability, reliability and authenticity (Kallath, 2005). Separating administrative domains, identifying system's vulnerabilities and developing confidence level of trust are required for determining and estimating trust among different distributed systems.

Separating administration domains: Each domain as an administrative boundary collects assets and user roles that share common trust requirements with the same system's expectations. Particularly, different domains have different security levels and different information types so is necessary to isolate domains for security reason and isolate risks from different domains (Dong *et al.*, 2009). Eventually, the major challenge of security domain is an unauthorized access that prevents other domains to establish a secure relation with harmed domain.

Virtualization as the main feature and crucial technique allows computing duties, run under the virtual environment rather than physical hardware (He and Liang, 2012). Furthermore, leveraging virtualization and trusted computing technologies are required to develop threat modeling framework in distributed multi domains environment. Trusted Virtual Domains (TVD) (Fig. 8) as a coalition of virtual machines that trusted each other provides isolated computing environment, obvious trusted relationships, secure communications and transparent policy enforcement (Lohr *et al.*, 2010). On the other hand, TVDs provides limited boundaries to protect distributed environment over several physical platforms. The Trusted Computing Base (TCB) by responding to present domains trustworthiness, security and TVD policy, guarantees data integrity and confidentiality in execution environment. The most important critical and complex issues that have to be addressed by TVD is satisfying business level of security by simplifying management, improving level of trust and containment.

Establish trust level: Trust level is based on varied and wide necessary trust criteria same as user's roles, offered services, assets, organization structure and technology issues in each virtual domain that offer trustworthiness as a final result. Furthermore, is difficult to understand which trust perspective can be used by domain to asses trust. The trust level in a specific environment or domain always is helpful for manager or director to make their decision. On the other hand, establishing trust between different domains is based on assessment of trust level and balance of trustworthiness among different individual domains (Msanjila and Afsarmanesh, 2007a). However, measuring trustworthiness is complex with single values or metrics so several constraints have to be met to measure trust level of each domain. Nevertheless, this study tried to present the most important elements for modeling trust level in various domains with different trust perspectives.

IriusRisk
Ex. 1002 - Page 488

Users: Different users with different accesses create various trust levels in each domain. For instance, different users same as administrator, authorized users, unregistered users have different trust levels.

Trust perspectives: Each character has a unique trust perspective for deciding on useful information to assess trust level. The organization, social, financial, technology and behavior are the more identified perspectives (Msanjila and Afsarmanesh, 2007b).

Time: Trust level in each virtual domain is dynamic and depending on changes of assessment approaches same as users, assets or other factors. Eventually, time as an important factor should be addressed for modeling trust level in each domain.

Trust level assessment in each domain is the fundamental step for establishing trust relationships among different domains. The main aims of this step is ensuring the acceptable level of trust in each domain, each domain's member conform to the specified trust level, trustworthiness for each object is met and allows trust's experts to more focus on low level trust domain to identify threat and vulnerabilities. For presenting trust level in an understandable format, we can rate trustworthiness in five rating namely Strongly less trustworthy, less trustworthy, average trustworthy, more trustworthy and strongly more trustworthy (Msanjila and Afsarmanesh, 2007b).

Scanning domain security: Scanning of distributed system for management purpose is a complex task by heterogeneity and distribution of resources and tools (Sladic *et al*., 2011). By developing web services and distributed environments managing and protecting system against vulnerability is a crucial task that seeks out security faults and weaknesses by using different tools same as vulnerability scanning and firewalls. Detecting vulnerability by scanning distributed and virtual environment is used in this research to detect some dangerous gaps and weaknesses before exploiting by threats. Traditional vulnerability control center and several scanning agents in limited area. These models are obsoleted for scanning same as Nessus, X-Scan and agent-based, are using only one functional scanning entity or one scanning domain-based vulnerability scanning. In other words, only one control center handle huge amount of tasks so bottle-neck as the main issue affects system by poor accuracy and time performance (Wang *et al*., 2013). By developing distributed environment more efficient security management method is required. For this reason, it seems necessary to develop domain-oriented distributed scanning mechanism to satisfy system by high scanning efficiency, simple service providing, flexible deployment and incredible scalability (Chao *et al*., 2009). In this model, each administration domain has own scanning resources to detect vulnerabilities and all scanning resources are managed by controlling domain or integrated center. These principles allows huge amount of tasks have to be analyzed without any bottleneck and satisfy load balancing.

Identify threats and vulnerabilities: By developing information and communication technology, organizations are more interested to outsource their computational resources on virtual domains. In fact, this huge amount of data can be damaged by threats from different resources (employees' activities or malicious attacks by attackers). Therefore, management and risk assessment need to understand and analyze threats and vulnerabilities as a crucial security issues. Thus, effective security classification is necessary to identify and organize threats and vulnerabilities into classes based on the intended effect of attacks and develop solutions to prevent system from effective threats (Jouini *et al*., 2014).

There are several threat classification methods on the base of attack and attacker achievements like unauthorized access.

Threat classification: The classification of threat allows manager to protect system by identify characteristics of threats and select efficient security solution. The most of classification's approaches are based on threat definitions. On the other hand, threats are defined as a technique that is used by attacker to exploit vulnerabilities or impact system. For instance, three dimensional model (Ruf *et al*., 2003) is represented on the base of attack technique by dividing threats to three subdivisions: motivation, localization and agent (Fig. 9).

- Threat agent is the actor that imposes the threat on specific assets and is divided to human, technological and force majeure
- The threat motivation is sufficient to classify threats to answer the question 'why' a threat is created. Threat motivation is classified into two classes: Deliberate and accidental
- Threat localization represents the origin of threats, internal or external

Hybrid model or information system security threat cube classification model (Geric and Hutinski, 2007) uses classification criteria that are necessary for constantly changing environments (Fig. 10). The three main steps of this model are:

- Security threat frequency shows the frequency of security threat occurrence
- Area of security threat activity shows the main domain that is effected by the threat same as personal, physical, communication and operational security

Security threat source represents the type of threat's resource same as insider and outsiders sources.

IriusRisk
Ex. 1002 - Page 489

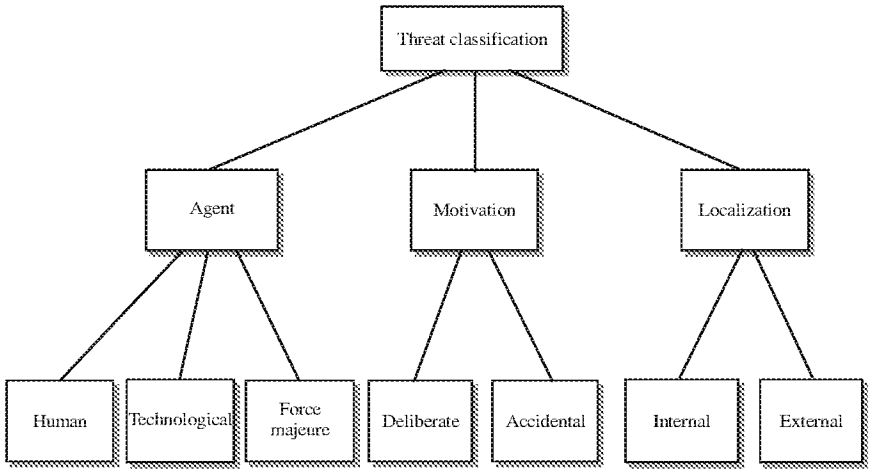


Fig. 9: Three dimension model

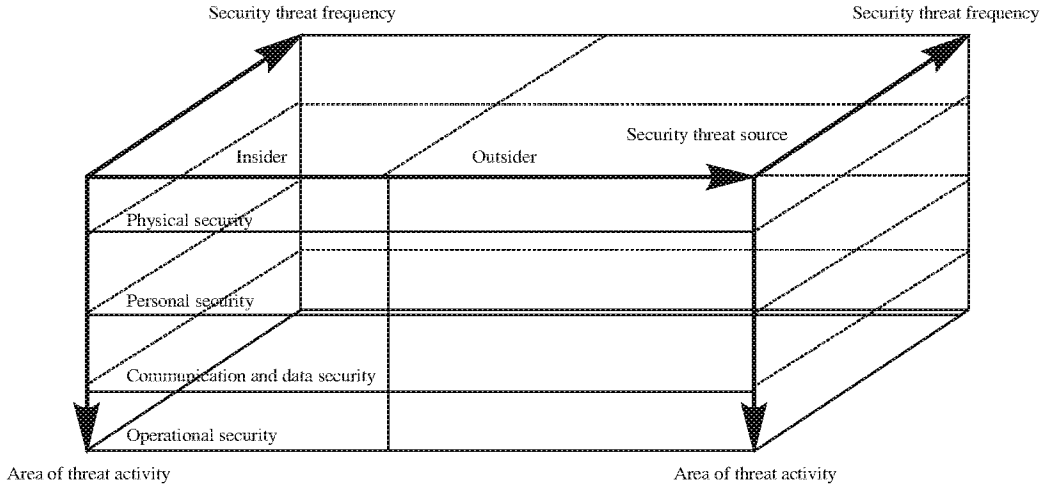


Fig. 10: Threat cube classification model

Information security threats classification pyramid (Alhabeeb *et al*., 2010) has been presented to classify threats in a dynamic environment and deliberate threats based on three factors:

- Attackers' prior knowledge about system represents the attacker knowledge about system includes system hardware, software, employees and users information
- Critically of the area represents the parts of system that are more critical than other parts of system and they can be affected by threats
- Loss represents any loss that can occur in the system by effecting privacy, integrity, business repudiation and finance loss

The proposed threat classification models are usually limited on use of criteria so, they are sufficient for stable environment where security threats are stable. Nowadays, by developing computing and virtual environments, organizations are faced by huge dynamic environment that allows different threats to attack system constantly. In this situation, dynamic classification model allows organizations to understand threats and their influence on assets and also realize area which threat could affect.

Multi-dimensions model (Jouini *et al*., 2014) (Fig. 11) is a hybrid model for information system security threat classification to combine most criteria of threats classifications. In this model threat classification is based on five basic criteria as source, agent, motivation and impact. The first level of classification consists of threat's source. In addition, environmental threat is divided to internal and external threats. The human (insider or hacker), environment (flood, fire, earthquake, wind and etc.) and technological threats (power supplies) are known as a threat agent. Actually, threat agent is the actor that affects system by brings in threat to the system. The next level is threat motivation that is the main goal of attacker to attack system. Particularly, viruses, Trojans and worms are malicious threat that is caused by

IriusRisk
Ex. 1002 - Page 490

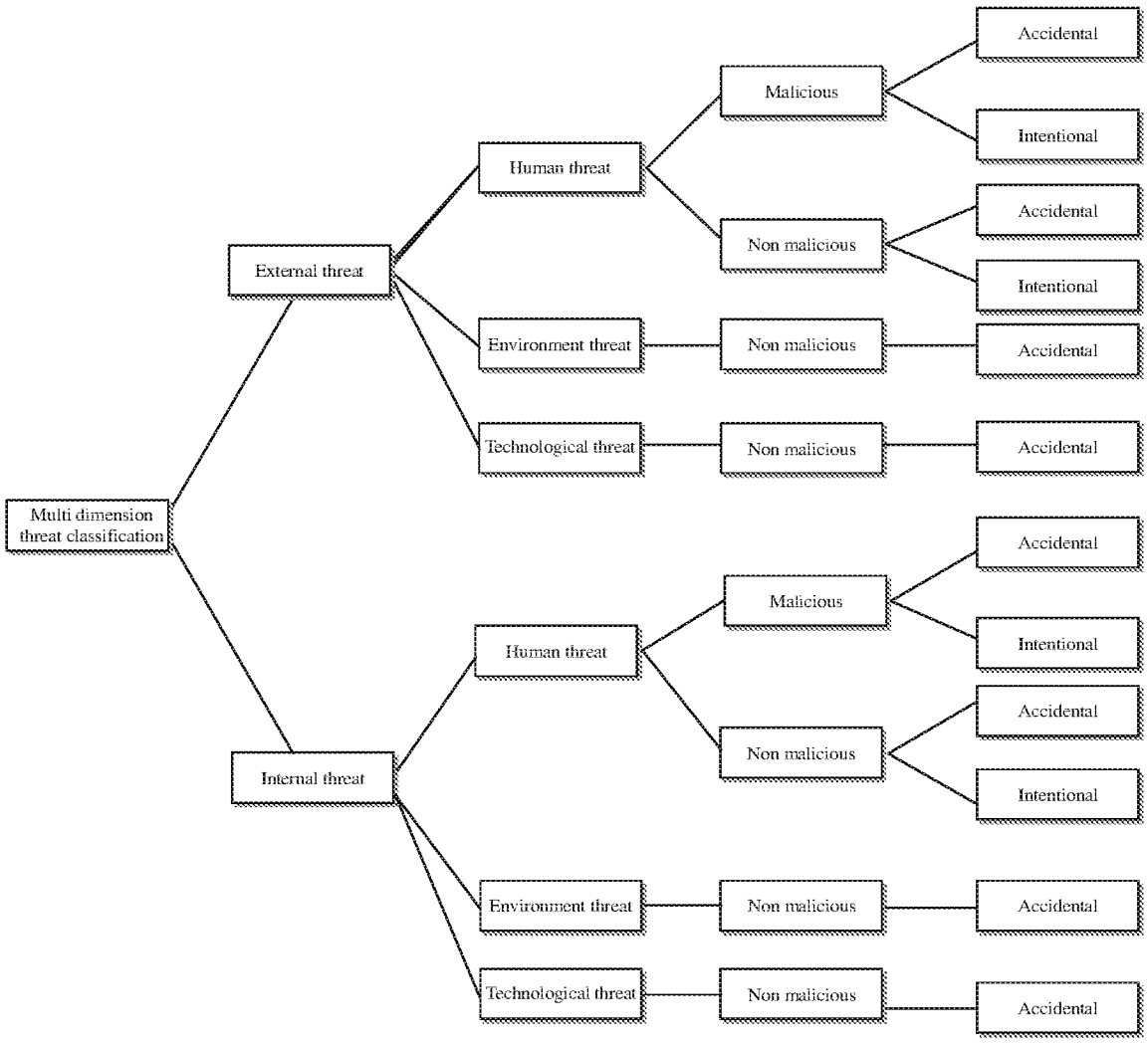


Fig. 11: Multi-dimensions model

attacker for disrupting or harming system. The vulnerabilities and errors are taking place in system because of poor security control or protocols and they are known as a none-malicious attack. The forth level of multi-dimensions model is related to user's intent that who caused the threat. International threat as a harmful decision (damage information deliberately, identity theft, ATM card crime) and accidental threat as a damage without awareness (operator mistake, coding error) are two kinds of this threat. The final part of this threat classification model is impacting threat. Each threat can cause one or many damaging impact to system same as information corruption, data loosing, disclosure of information, denial of use and data destruction.

This threat classification divides threats to different criteria to cover all security risks and allows organization to adopt large scale systems. For moving IT system to the cloud, new classes of threats have to be identified. Some of new available threats are inside trust boundary that should be considered by cloud's tenant and some else are related to cloud provider. Finally, all constraints and security issues propel managers and providers to design and adopt dynamic and hybrid threat classification model with their system to cover security risk and protect system from exploitation of vulnerabilities by threats.

Vulnerability classification: The main effect of vulnerabilities is incurring high cost for businesses by undermining degree of integrity, confidentiality, safety, availability and reliability. Classification of vulnerability helps businesses to publish, identify and analysis known and unknown vulnerabilities. For this reason, an acceptable and sufficient classification method has to follow below principles (Jin *et al*., 2009):

- **Public acceptance:** A classification model should be accepted publicly
- **Comprehensibility:** A classification has to be understood by expert and ordinary users
- **Completeness:** A classification should classify most of known and unknown vulnerabilities

IriusRisk
Ex. 1002 - Page 491

Table 2: Relation between threats, vulnerabilities and their counter measure

Threats	Vulnerabilities	Countermeasure
T1	V1, V2, V3, V4, V5, V6	Data Encryption, data signature, DLP as a service
T2	V1, V3, V5, V6	Identity and Access Management (IAM) services (CSA., 2012)
T3	V1,V3,V4	Authentication (Soares *et al*., 2013), Access control (Khan, 2012)
T4	V1, V2,V4, V5, V6, V7	Apply security patches, use an IPS for monitoring, configuring firewall, minimize IP spoofing
T5	V2,V5	Cryptography, separation of duties, logging and auditing, legal contracts and insider detection models (Kandias *et al*., 2012)
T6	V1, V3, V4, V5	Stop incursions, data protection policies, automating periodic check, security event management, monitoring technologies and authenticate identities
T7	V1, V4, V5	Stricter initial registration, credit card fraud monitoring, black list monitoring (CSA., 2010)
T8	V8	Trusted Third Party (TTP), Rating cloud service provider
T9	V2	Trusted Cloud Computing Platform (TCCP) (Santos *et al*., 2009), secure protocol and live migration

- **Determinism:** A classification should have comprehensive process
- **Mutual exclusion:** A classification has to classify vulnerabilities into minimum number of classes
- **Repeatability:** A classification model can be repeatable

Now-a-days, vulnerability is known as a new opportunity for attacker to access and exploit valuable business services. Cloud computing as a multi-tenancy and distributive environment increases substantially the possibility of vulnerability exploitation. In this situation, deploying online classification of vulnerability assists businesses to protect system from reported vulnerabilities. Web Services (WS) as a core cloud technology and XML implementation of Service-Oriented Architecture (SOA) plays an important role to classify and taxonomy vulnerabilities so we should have a closer look at SOA and WS based classification approaches. On the other hand, vulnerabilities those are associated with existing technologies of cloud computing same as SOA, WS and virtualization have significant impact on the cloud. The main goals of vulnerability analysis as avoiding, finding, fixing and monitoring vulnerability leads to vulnerability classification (Lowis and Accorsi, 2009).

Berghe *et al*. (2006) proposed a vulnerability classification model on the base of system components relation and their influence on vulnerabilities. Seacord and Householder (2005) suggested a classification view based on attributes and values to protect system from vulnerability exploitation. Parrend and Frenot (2008) deployed java Service Oriented Programming (SOP) platform to classify vulnerabilities on the base of vulnerability's categorizes and goal of attacks that are based on these vulnerabilities. The ATLIST as a vulnerability analysis method was represented by Lowis and Accorsi (2009) to detect known vulnerabilities rather than new vulnerabilities. However, for categorizing vulnerabilities in cloud computing the three different services of cloud computing as SaaS, PaaS, IaaS (SPI model) have to be considered. On the base of SPI model, insecure interface, unlimited allocation of resources, data related vulnerabilities, virtual machine, SOA and WS are the main impressive vulnerabilities in cloud computing (Hashizume *et al*., 2013).

Cloud computing as a combination of different technologies is faced by huge amount of threats and vulnerabilities that force organization to consider more complex classification mechanisms than traditional method. Furthermore, traditional web services and virtualization techniques offer some solutions that are unfledged for computing environment. Nevertheless, numerating cloud's vulnerabilities may not enough and based on existing new threats and security techniques can identify more vulnerability to make system more robust.

Implementing countermeasure: The main goal of countermeasure as an action, device, procedure, technique or other measure, is reducing vulnerability and protecting assets against threats (Laorden *et al*., 2010). Regardless cloud computing services, data confidentiality, integrity and availability as the main security challenges of cloud computing present the base of countermeasures to protect critical information. On the other hand, each security countermeasure is based on the measured threat, being offered as a service in the cloud. Countermeasure as a service is identified on the base of relationship between threats and vulnerabilities (Table 2).

Determine the result: Estimating and ranking the severity of threats and vulnerabilities to inform decision about what to do to protect system from harmful effects is the last step of proposed model. Having a system to ranking threats and vulnerability is needed to save time and prevent system from more serious security risks. The result of this step can be a list or database of threat and vulnerability profiles that consist of sorted security risks.

Threat ranking: Ranking threats as a complex task is depended on various factors same as existing vulnerabilities, likelihood of attacks and potential impact of attacks. The first step for ranking threats is considering different risk factors that are posed by identified threats. On the base of threat model, various criteria have to be considered as the main exploitation factors of risks. For instance, Microsoft Threat Modeling introduces Damage potential, Reproducibility, Exploitability, Affected users and Discoverability (DREAD) as different influencing agents of a threat (Malik *et al*., 2008). The equation that is used to compute a risk value has been shown as follows:

$$\text{Risk value} = (D+R+E+A+D)/5$$

The amount of risk value is a number between 0 and 10. It means the higher number is more serious risk. The Open Web Application Security Project (OWASP) as a universal

IriusRisk
Ex. 1002 - Page 492

risk rating system estimates all risks for all organization accurately. This model introduces a rating system to estimate the severity of all security risks on the base of likelihood, impact and combination of them to determine threats as the main factor of security risk. The equation that is used in OWASP is shown as follows:

$$\text{Risk} = \text{Likelihood} \times \text{Impact}$$

In this rule the risk consists of threat agents and their impact on system. The likelihood factor is estimated from a successful attack to exploit vulnerabilities. The amount of likelihood is low, medium or high (0-9). The impact as an effect of successful attack, divided to technical and business. These impacts are related to each other and each one has a set of options with different impact rating from 0-9. In the final stage, impact and likelihood estimations are putting together to calculate the overall accuracy for the risk. The OWASP by defining a dynamic system is more suitable for web services. On the other hand, this model is not based on some predefined threats same in DREAD or Trike. In any case, the customizability and adoptability of this model help businesses to produce result based on their perception about more risky threats.

Vulnerability ranking: Estimating and ranking vulnerability as a complex multi-step mechanism allows businesses to protect system from harsh attacks. The Common Vulnerability Scoring System (CVSS) (Hanford, 2007) is a universal open and standardized method for rating IT vulnerabilities. The CVSS offers evaluation metrics; base metric, temporal metric and environment metric for evaluating security status. Base metric provides original and basic attribute of vulnerabilities, environment metric evaluate vulnerabilities' metrics that are depended on the environment and temporal metric deploys dynamic aspect of vulnerabilities.

Applying CVSS to a business process cannot fully deliver optimal security measure for dynamic environments same as SOA and Web-Services (Zhao *et al*., 2011). VRank and hybrid ranking have been developed to satisfy business requirements. The VRank (Jiang *et al*., 2012) as a dynamic framework is able to integrate existing vulnerability databases with the specific detail context of business process. Accordingly, the VRank provides more specific vulnerability assessment for SOA. The hybrid ranking model proposes a combination of CVSS rating and numerical estimation of vulnerability that influence the global network (Zhao *et al*., 2011). By developing dynamic environments estimating vulnerability in an influential level and aggregating other rating models with high-level rating technics provides a hybrid model to precise measurement of vulnerabilities more economic and efficient.

CONCLUSION

This study reviewed the different existing threat modeling approaches and proposes a dynamic threat modeling method to be adopted by cloud computing. The proposed threat model allows organization to identify threats and vulnerabilities, justify countermeasures and document the more effected security risks. Unlike other models, proposed dynamic model is not based on some predefine threats. On the other hand, we plan to deploy a novel model to define new threats which are more critical than expected threats in static models. Eventually, in this research we effort to expand a dynamic threat model by respecting cloud essence as a reliability, mobility, scalability and flexibility. By considering the contribution of cloud computing in IT industries, it's obvious that cloud computing is a leading and innovative strategy to attract much attentions. However, developing and designing in-depth security techniques same as threat modeling is needed to satisfy cloud users and providers.

ACKNOWLEDGMENTS

First I would like to thank to my supervisor Dr. Norziana BteJamil for guidance and advances. She inspired and contributed me greatly to work in this study. Beside, I should like to thank the authority of University Tenaga Nasional (UNITEN) for providing good environment and facilities to complete this research. Special thanks to Assoc. Prof. Abdul Rahim Bin Ahmad, for sharing their invaluable assistance. Finally, I would specially thank to my family that support me for completing this study. Without helps of the person that mentioned above, I couldn't be success to publish this paper.

REFERENCES

Alhabeeb, M., A. Almuhaideb, P.D. Le and B. Srinivasan, 2010. Information security threats classification pyramid. Proceedings of the IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, April 20-23, 2010, Perth, WA., pp: 208-213.

Amadi-Echendu, J.E., 2004. Managing physical assets is a paradigm shift from maintenance. Proceedings of the IEEE International Engineering Management Conference, Volume 3, October 18-21, 2004, Singapore, pp: 1156-1160.

Berghe, C.V., J. Riordan and F. Piessens, 2006. A vulnerability tax-onomy methodology applied to web services. Proceedings of the 10th Nordic Workshop on Secure IT Systems, October 20-21, 2005, Tartu, Estonia.

Bertino, E., L. Martino, F. Paci and A. Squicciarini, 2010. Security for Web Services and Service-Oriented Architectures. Springer, New York, USA., ISBN: 978-3-540-87742-4, Pages: 226.

CSA., 2010. Top threats to cloud computing V1.0. Cloud Security Alliance (CSA), March, 2010, pp: 1-14. https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf

CSA., 2012. Identity and access management. Cloud Security Aliance (CSA), September 2012, pp: 1-45. https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_1_IAM_Implementation_Guidance.pdf

IriusRisk
Ex. 1002 - Page 493

Chao, D., Y. Danfeng, Y. Yun and Y. Fangchun, 2009. A domain-oriented distributed vulnerability scanning mechanism. Proceedings of the 2nd IEEE International Conference on Broadband Network and Multimedia Technology, October 18-20, 2009, Beijing, China, pp: 831-836.

Demchenko, Y., C. Ngo, M.X. Makkes, R. Strijkers and C. de Laat, 2012. Defining inter-cloud architecture for interoperability and integration. Proceeding of the 3rd International Conference on Cloud Computing, GRIDs and Virtualization, July 22-27, 2012, Nice, France, pp: 174-180.

Dong, G., Z. Liu and D. Zhao, 2009. A security domain isolation and data exchange system based on VMM. Proceedings of the 3rd International Conference on Signal Processing and Communication Systems, September 28-30, 2009, Omaha, NE., pp: 1-5.

Geric, S. and Z. Hutinski, 2007. Information system security threats classifications. J. Inform. Organiz. Sci., 31: 51-61.

Grobauer, B., T. Walloschek and E. Stocker, 2011. Understanding cloud computing vulnerabilities. IEEE Secur. Privacy, 9: 50-57.

Hamza, Y.A. and M.D. Omar, 2013. Cloud computing security: Abuse and nefarious use of cloud computing. Int. J. Comput. Eng. Res., 3: 22-27.

Hanford, S., 2007. Common Vulnerability Scoring System (CVSS-SIG): New version of common vulnerability scoring system released. http://www.first.org/cvss

Hashizume, K., D.G. Rosado, E. Fernandez-Medina and E.B. Fernandez, 2013. An analysis of security issues for cloud computing. J. Internet Services Applic. 10.1186/1869-0238-4-5

He, Z. and G. Liang, 2012. Research and evaluation of network virtualization in cloud computing environment. Proceedings of the 3rd International Conference on Networking and Distributed Computing, October 21-24, 2012, Hangzhou, pp: 40-44.

Jehangir, A. and S.M.H de Groot, 2012. Securing inter-cluster communication in personal networks. Proceedings of the 4th Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, August 6-10, 2007, Philadelphia, PA., pp: 1-6.

Jiang, J., D. Liping, E. Zhai and Y. Ting, 2012. VRank: A context-aware approach to vulnerability scoring and ranking in SOA. Proceedings of the IEEE 6th International Conference on Software Security and Reliability, June 20-22, 2012, Gaithersburg, MD., USA., pp: 61-70.

Jin, S., Y. Wang, X. Cui and X. Yun, 2009. A review of classification methods for network vulnerability. Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, October 11-14, 2009, San Antonio, pp: 1171-1175.

Jouini, M., L.B.A. Rabai and A.B. Aissa, 2014. Classification of security threats in information systems. Procedia Comput. Sci., 32: 489-496.

Kallath, D., 2005. Trust in trusted computing the end of security as we know it. Comput. Fraud Secur., 2005: 4-7.

Kandias, M., N. Virvilis and D. Gritzalis, 2012. The insider threat in cloud computing. Proceedings of the 6th International Workshop on Critical Information Infrastructure Security, September 8-9, 2011, Switzerland, pp: 99-103.

Kenyon, H., 2010. Closing an overlooked vulnerability. Government Computer News, June 9, 2010, pp: 25-27. https://www.drivesaversdatarecovery.com/images/pdf/GCN+09_06_10.pdf

Khan, A.R., 2012. Access control in cloud computing environment. ARPN J. Eng. Applied Sci., 7: 613-615.

Laorden, C., B. Sanz, G. Alvarez and P.G. Bringas, 2010. A threat model approach to threats and vulnerabilities in on-line social networks. Proceedings of the 3rd International Conference on Computational Intelligence in Security for Information Systems, November 11-12, 2010, Leon, Spain, pp: 135-142.

Lohman, T., 2011. DDoS is cloud's security achilles heel. http://www.computerworld.com.au/article/401127/ddos_cloud_security_achilles_heel/

Lohr, H., T. Poppelmann, J. Rave, M. Steegmanns and M. Winandy, 2010. Trusted virtual domains on OpenSolaris: Usable secure desktop environments. Proceedings of the 5th ACM Workshop on Scalable Trusted Computing, October 4-8, 2010, Chicago, IL., USA., pp: 91-96.

Lowis, L. and R. Accorsi, 2009. On a classification approach for SOA vulnerabilities. Proceedings of the 33rd Annual IEEE International Computer Software and Applications Conference, Volume 2, July 20-24, 2009, Seattle, WA., pp: 439-444.

Malik, N.A., M.Y. Javed and U. Mahmud, 2008. Threat modeling in pervasive computing paradigm. Proceedings of the Mobility and Security, New Technologies, November 5-7, 2008, Tangier, pp: 1-5.

McRee, R., 2008. PTA: Practical threat analysis. Information Systems Security Association, September 2008, pp: 37-40.

Msanjila, S.S. and H. Afsarmanesh, 2007a. Modelling trust relationships in collaborative networked organisations. Technol. Transfer Commer., 6: 40-55.

Msanjila, S.S. and H. Afsarmanesh, 2007b. Towards Establishing Trust Relationships among Organizations in VBEs. In: Establishing the Foundation of Collaborative Networks: IFIP TC 5 Working Group 5.5 Eighth IFIP Working Conference on Virtual Enterprises September 10-12, 2007, Guimaraes, Portugal, Camarinha-Matos, L.M., H. Afsarmanesh, P. Novais and C. Analide (Eds.). Springer-Verlag, Berlin, Germany, ISBN: 978-0-387-73798-0, pp: 3-14.

Narayan, A., S. Rao, G. Ranjan and K. Dheenadayalan, 2012. Smart metering of cloud services. Proceedings of the IEEE International Systems Conference, March 19-22, 2012, Vancouver, BC., pp: 1-7.

IriusRisk
Ex. 1002 - Page 494

Parrend, P. and S. Frenot, 2008. Classification of component vulnerabilities in Java Service Oriented Programming (SOP) platforms. Proceedings of the 11th International Symposium, Component-Based Software Engineering, October 14-17, 2008, Karlsruhe, Germany, pp: 80-96.

Prasad, N.R., 2007. Threat model framework and methodology for Personal Networks (PNs). Proceedings of the 2nd International Conference on Communication Systems Software and Middleware, January 7-12, 2007, Bangalore, pp: 1-6.

Rai, P.K. and R.K. Bunkar, 2014. Study of security risk and vulnerabilities of cloud computing. Int. J. Comput. Sci. Mobile Comput., 3: 490-496.

Rouse, M., 2009. Virtual machine escape. http://whatis.techtarget.com/definition/virtual-machine-escape

Ruf, L., A. Thorn, T. Christen, B. Gruber and R. Portmann, 2003. Threat modeling in security architecture: The nature of threats. https://www.isss.ch/fileadmin/publ/agsa/ISSS-AG-Security-Architecture__Threat-Modeling_Lukas-Ruf.pdf

Ryan, M.D., 2013. Cloud computing security: The scientific challenge and a survey of solutions. J. Syst. Software, 86: 2263-2268.

Santos, N., K.P. Gummadi and R. Rodrigues, 2009. Towards trusted cloud computing. Proceedings of the Conference on Hot Topics in Cloud Computing, June 14-19, 2009, Berkeley, CA, USA.

Seacord, R.C. and A.D. Householder, 2005. A structured approach to classifying security vulnerabilities. Technical Note CMU/SEI-2005-TN-003, Camegie Mellon Software Engineering Institute, Pittsburgh, PA., USA., January 2005.

Sladic, G., M. Vidakovic and Z. Konjovic, 2011. Agent based system for network availability and vulnerability monitoring. Proceedings of the IEEE 9th International Symposium on Intelligent Systems and Informatics, September 8-10, 2011, Subotica, pp: 53-58.

Soares, L.F.B., D.A.B. Fernandes, M.M. Freire and P.R.M. Inacio, 2013. Secure user authentication in cloud computing management interfaces. Proceedings of the IEEE 32nd International Performance Computing and Communications Conference, December 6-8, 2013, San Diego, CA., USA., pp: 1-2.

Takabi, H., M. Amini and R. Jalili, 2007. Trust-based user-role assignment in role-based access control. Proceedings of the IEEE/ACS International Conference on Computer Systems and Applications, May 13-16, 2007, Amman, pp: 807-814.

The Open Group, 2009. Risk Taxonomy. The Open Group, London, UK., ISBN: 1-931624-77-1, Pages: 39.

Tianyi, Z., L. Weidong and S. Jiaxing, 2011. An efficient role based access control system for cloud computing. Proceedings of the 11th IEEE International Conference on Computer and Information Technology, August 31-September-2, 2011, Paphos, Cyprus, pp: 97-102.

Wang, S., Y. Gong, G. Chen, Q. Sun and F. Yang, 2013. Service vulnerability scanning based on service-oriented architecture in web service environments. J. Syst. Architecture, 59: 731-739.

Zhao, F., H. Huang, H. Jin and Q. Zhang, 2011. A hybrid ranking approach to estimate vulnerability for dynamic attacks. Comput. Math. Applic., 62: 4308-4321.

IriusRisk
Ex. 1002 - Page 495

Electronic Acknowledgement Receipt

EFS ID:	38606546
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	17-FEB-2020
Filing Date:	15-AUG-2019
Time Stamp:	19:31:39
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Amendment/Req. Reconsideration-After Non-Final Reject	OAR1.pdf	91345 7aceba25f4f74221508d7fbc19b9d35d62594692	no	12

Warnings:

IriusRisk

Information:	
Total Files Size (in bytes):	91345

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

AMENDMENT/REPLY

Applicant: ThreatModeler Software Inc. **Docket No:** THREATM010

Examiner: Theodore C. Parsons **Ser. No:** 16542263

Filed: August 15, 2019 **Group Art Unit:** 2494

Title: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

RESPONSE TO OFFICE ACTION DATED NOVEMBER 18, 2019

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Examiner:

This is filed in response to the Office Action mailed November 18, 2019. This response is being filed within the shortened statutory period and therefore no extension of time fees are applicable. Reconsideration of the application is requested in view of the following:

Claim Amendments begin on page 2.

Remarks/Arguments begin on page 8.

IriusRisk
Ex. 1002 - Page 498

CLAIM AMENDMENTS

Please amend claim 16 as shown in the complete set of claims below.

1. (Original) A threat modeling method, comprising:

providing one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

providing one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components"); and

in response to receiving one or more user inputs, using one or more user interfaces displayed on one or more computing devices communicatively coupled with the one or more databases:

using the one or more computing devices, reading a data file generated by the third party software application, the data file comprising a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components;

determining using the one or more computing devices, for the subset of third party diagram components, correlated threat model components as defined in the one or more mapping files;

displaying on the one or more user interfaces a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, the relational diagram defining a threat model; and

IriusRisk
Ex. 1002 - Page 499

generating, using the one or more computing devices, and displaying, on the one or more user interfaces, a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

2. (Original) The method of claim 1, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

3. (Original) The method of claim 1, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

4. (Original) The method of claim 1, wherein the data file comprises one of a VSD file, a VSDX file, a TMT file, an XML file, a JSON file, a PNG file, and a JPEG file.

5. (Original) The method of claim 1, wherein determining, for the subset of third party diagram components, correlated threat model components, comprises first determining whether a user-defined name for one of the subset of third party diagram components has a match in the one or more mapping files and, if no match is found, then determining whether a default title for the third party diagram component, as defined by the third party software application, has a match in the one or more mapping files.

6. (Original) The method of claim 5, further comprising, if no match is found for the default title, matching the third party diagram component with a placeholder component and displaying the placeholder component on the relational diagram to indicate that no match was found.

IriusRisk
Ex. 1002 - Page 500

7. (Original) The method of claim 1, further comprising determining, for the one or more relationships defined by the data file, correlated threat model components as defined in the one or more mapping files.

8. (Original) A threat modeling system, comprising:

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with visual diagram components of a third party software application (hereinafter "third party diagram components");

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by one of the one or more computing devices, the data file generated by the third party software application, the data file comprising a subset of the third party diagram components, the data file defining one or more relationships between the subset of third party diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of third party diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of one of a system, an application, and a process, using visual representations of the threat model components correlated with the subset of third party diagram components, the relational diagram defining a threat model; and

IriusRisk
Ex. 1002 - Page 501

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

9. (Original) The system of claim 8, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

10. (Original) The system of claim 8, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

11. (Original) The system of claim 8, wherein the data file comprises one of a VSD file, a VSDX file, a TMT File, an XML file, a JSON file, a PNG file, and a JPEG file.

12. (Original) The system of claim 8, wherein the one or more mapping files includes a file correlating user-defined names of the third party diagram components with the threat model components.

13. (Original) The system of claim 8, wherein the one or more mapping files includes a file correlating default titles of the third party diagram components with the threat model components.

14. (Original) The system of claim 8, wherein, for each of the subset of third party diagram components for which no correlated threat model component is found in the one or more mapping files, the diagram interface displays a placeholder component indicative of a match not being found.

IriusRisk
Ex. 1002 - Page 502

15. (Original) The system of claim 8, further comprising one or more mapping interfaces displayed on the one or more computing devices and configured to, in response to receiving user input, modify the one or more mapping files.

16. (Currently Amended) A threat modeling system, comprising:

one or more computing devices communicatively coupled with one or more databases, the one or more databases comprising:

a plurality of threat model components stored therein; and

a plurality of threats stored therein, wherein each of the threats is associated with at least one of the threat model components through the one or more databases;

one or more mapping files communicatively coupled with the one or more databases, the one or more mapping files correlating the threat model components with ~~VISIO~~ visual diagram components of a third party software application (hereinafter "third party diagram components");

an import interface displayed on one of the one or more computing devices, the import interface configured to, in response to receiving user input, initiate reading of a data file by the one or more computing devices, the data file comprising a subset of the ~~VISIO~~ third party diagram components, the data file defining one or more relationships between the subset of ~~VISIO~~ third party diagram components;

an interface configured to, in response to receiving user input, initiate a determination of which threat model components are correlated with the subset of ~~VISIO~~ third party diagram components in the one or more mapping files;

a diagram interface, displayed on the one or more computing devices, displaying a relational diagram of a computing network, using visual representations of the threat model components correlated with the subset of ~~VISIO~~ third party diagram components, the relational diagram defining a threat model; and

IriusRisk
Ex. 1002 - Page 503

a threat report interface, displayed on the one or more computing devices, including a threat report displaying each threat that is associated through the one or more databases with one of the threat model components included in the threat model.

17. (Original) The system of claim 16, wherein the one or more mapping files are not stored in the one or more databases but are stored on the one or more computing devices.

18. (Original) The system of claim 16, wherein the one or more mapping files comprise one or more tables stored within the one or more databases.

19. (Original) The system of claim 16, wherein the one or more mapping files includes a file correlating user-defined names of the third party diagram components with the threat model components.

20. (Original) The system of claim 16, wherein the one or more mapping files includes a file correlating default titles of the third party diagram components with the threat model components.

IriusRisk
Ex. 1002 - Page 504

Remarks/Arguments

Claims 1-20 are pending. Reconsideration is requested in light of the following.

Examiner Interview

Examiner Parsons and undersigned counsel for applicant participated in a telephonic interview on 1/7/20. The examiner is thanked for his time and suggestions.

In the interview it was indicated that the drawing objection (referenced below) could be withdrawn without amendment.

After clarification in the interview of the "third party diagram components" and "visual representations of the threat model components" being distinct elements (the third party diagram components being components of an independent software application and the visual representations of the threat model components being components of the native software, stored in databases/servers locally or remotely but coupled with the system), the examiner indicated that the 112(b) rejections of claims 1-20 based on this language could be withdrawn without amendment.

With regards to the 112(b) rejection of claims 1, 5, 6, 8, 12-14, 19-20 on the basis of the "third party" language of those claims, the examiner and counsel agreed that a clarification on the record of "third party" not necessarily indicating present ownership of software applications (but rather indicating independently-created software applications not created with a common schema, such that a mapping file or the like as claimed in the claims would be useful to map between elements of the two software applications), would be sufficient to overcome the 112(b) rejections of the "third party" language.

In the interview counsel for applicant further agreed to remove VISIO from the claims.

Drawing Objections

IriusRisk
Ex. 1002 - Page 505

Figure 1 was objected to on the basis of boxes and shapes needing descriptive text labels. In the 1/7/20 examiner interview, referenced above, the examiner indicated that this objection could be withdrawn without amendment and, accordingly, FIG. 1 is left unamended but it is requested that the objection of FIG. 1 be withdrawn.

Rejections under 35 U.S.C. 112(b)

Claims 1-20 were rejected under 35 U.S.C. 112 for indefiniteness. Specifically, the examiner indicated that the relationship was unclear between the "threat model components," the "visual representations of the threat model components" and the "third party diagram components" (which the examiner pointed out were likely themselves visual representations). In the above-referenced interview it was clarified that the threat model components are stored in the database(s) of the system, as are the visual representations of the threat model components, and that the "third party diagram components" are components (possibly visual) of a third party software application. The mapping file correlates the internal threat model components with the external third party diagram components so that the system can read a file created with the third party software program and generate its own diagram using internal components that are, through the mapping file, correlated to the external third party components. With this clarification, the examiner indicated that this rejection could be withdrawn without amendment and, accordingly, the claims are not amended with regards to this language, but it is requested that the 112(b) rejection based on this language be withdrawn.

Claim 16 was also rejected under 112(b) for including the trademark VISIO. This term has been removed from claim 16 herein, and it is accordingly requested that the 112(b) rejection of claim 16 based on this language be withdrawn.

Claims 1, 5, 6, 8, 12-14, 19-20 were rejected under 112(b) for including the term "third party diagram components." The examiner argued that "third party" refers to ownership, which can change at arbitrary times, arguing that this rendered the claims indefinite as a

IriusRisk
Ex. 1002 - Page 506

potential infringer may not know who owns the software applications at any given time. Applicant clarifies that "third party" as used in the claims and in the specification does not necessarily refer to present ownership of the software applications, but rather is a shorthand way the applicant refers to distinct software applications created without a common schema so that a mapping file or the like is useful to correlate components of one software program with the other. While this may not be explicitly pointed out in the specification, it is believed that this is implicit based on the application never referring to ownership or custody of the software programs but, rather, frequently focusing on the need for a mapping file to correlate the two software programs. For example, the terms "owner," "ownership," and "custody" do not appear in the application, while the term "mapping" is included 94 times in the application.

It is true that ownership of software programs can change. For example if OWNER A owned a threat modelling software (SOFTWARE A) that read on the claims of the present application, OWNER B owned a separate diagramming software program (SOFTWARE B) created using a different schema, and SOFTWARE A was configured to import files created in SOFTWARE B and, using a mapping file, generate a diagram and threat model, in that scenario SOFTWARE B would be considered a "third party" software and its components would be considered "third party diagram components." However, if OWNER A then purchased SOFTWARE B from OWNER B so that SOFTWARE A and SOFTWARE B were jointly owned, a mapping file of the type discussed in the claims would still be useful to allow a user to import files created in SOFTWARE B and, using the mapping file, generate a diagram and threat model. Accordingly, in this second scenario, SOFTWARE B would still be considered a "third party" software file even though, at the time of common ownership, there is no "third party" involved. Indeed, even if SOFTWARE A and SOFTWARE B were originally created and owned by a single party (so that, in the traditional sense, there was no "third party" to begin with), if SOFTWARE A and SOFTWARE B are created using different schemas so that a mapping file would be useful for correlating elements of SOFTWARE A with elements of

IriusRisk
Ex. 1002 - Page 507

SOFTWARE B, then SOFTWARE B would still be a "third party software" as that term is broadly used in the application which is, as indicated above, a shorthand way of referring to software applications created without a common schema and benefitting from a mapping file to correlate elements. With this clarification, it is requested that the rejection of the "third party" language under 112(b) be withdrawn.

It is believed that the 112(b) rejection of claims 19-20 for lack of antecedent basis with regards to "the third party diagram components" is overcome by the amendment to claim 16 which now introduces the third party diagram components, and it is accordingly requested that this rejection be withdrawn.

Note regarding Doctrine of Equivalents

Any amendments herein not specifically made for patentability have been made for other purposes, such as clarity, and no such amendments should be interpreted as limiting claim scope or application of the doctrine of equivalents.

IriusRisk
Ex. 1002 - Page 508

CONCLUSION

In conclusion, it is respectfully requested that the application be allowed. If it will in any way speed examination and/or allowance, the Examiner is invited to contact an attorney of record using the contact information listed below.

Respectfully submitted,

Date: February 17, 2020 By: /Paul B. Johnson/

Paul B. Johnson
Reg. No. 59375

Law Office of Paul B. Johnson
8483 S 1275 E
Sandy, UT 84094-1363
Ph: 801.719.5229
Fax: 801.317.8889
paul@pauljohnsonlaw.com

IriusRisk
Ex. 1002 - Page 509

PTO/SB/06 (09-11)
Approved for use through 1/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 16/542,263	Filing Date 08/15/2019	☐ To be Mailed

ENTITY: ☐ LARGE ☑ SMALL ☐ MICRO

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A		N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A		N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*		x $50 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*		x $230 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).				
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))					
* If the difference in column 1 is less than zero, enter "0" in column 2.				TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT	02/17/2020	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0		x $50 =	0
	Independent (37 CFR 1.16(h))	* 3	Minus	*** 3	= 0		x $230 =	0
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=		x $0 =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=		x $0 =	
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.	LIE
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".	/VANESSA D HODGES/
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".	
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.	

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

145042 7590 03/05/2020
Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

EXAMINER
PARSONS, THEODORE C

ART UNIT	PAPER NUMBER
2494	

DATE MAILED: 03/05/2020

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/542,263	08/15/2019	Anurag Agarwal	ThreatM010	7756

TITLE OF INVENTION: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$500	$0.00	$0.00	$500	06/05/2020

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the ENTITY STATUS shown above. If the ENTITY STATUS is shown as SMALL or MICRO, verify whether entitlement to that entity status still applies.

If the ENTITY STATUS is the same as shown above, pay the TOTAL FEE(S) DUE shown above.

If the ENTITY STATUS is changed from that shown above, on PART B - FEE(S) TRANSMITTAL, complete section number 5 titled "Change in Entity Status (from status indicated above)".

For purposes of this notice, small entity fees are 1/2 the amount of undiscounted fees, and micro entity fees are 1/2 the amount of small entity fees.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Maintenance fees are due in utility patents issuing on applications filed on or after Dec. 12, 1980. It is patentee's responsibility to ensure timely payment of maintenance fees when due. More information is available at www.uspto.gov/PatentMaintenanceFees.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), by mail or fax, or via EFS-Web.

By mail, send to: Mail Stop ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

By fax, send to: (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

145042 7590 03/05/2020

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being transmitted to the USPTO via EFS-Web or by facsimile to (571) 273-2885, on the date below.

(Typed or printed name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/542,263	08/15/2019	Anurag Agarwal	ThreatM010	7756

TITLE OF INVENTION: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$500	$0.00	$0.00	$500	06/05/2020

EXAMINER	ART UNIT	CLASS-SUBCLASS
PARSONS, THEODORE C	2494	726-025000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

❑ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

❑ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-09 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list
(1) The names of up to 3 registered patent attorneys or agents OR, alternatively,
(2) The name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 _____
2 _____
3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document must have been previously recorded, or filed for recordation, as set forth in 37 CFR 3.11 and 37 CFR 3.81(a). Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ❑ Individual ❑ Corporation or other private group entity ❑ Government

4a. Fees submitted: ❑ Issue Fee ❑ Publication Fee (if required) ❑ Advance Order - # of Copies _____

4b. Method of Payment: *(Please first reapply any previously paid fee shown above)*

❑ Electronic Payment via EFS-Web ❑ Enclosed check ❑ Non-electronic payment by credit card (Attach form PTO-2038)

❑ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment to Deposit Account No. _____

5. **Change in Entity Status** (from status indicated above)

❑ Applicant certifying micro entity status. See 37 CFR 1.29

❑ Applicant asserting small entity status. See 37 CFR 1.27

❑ Applicant changing to regular undiscounted fee status.

NOTE: Absent a valid certification of Micro Entity Status (see forms PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.
NOTE: If the application was previously under micro entity status, checking this box will be taken to be a notification of loss of entitlement to micro entity status.
NOTE: Checking this box will be taken to be a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: This form must be signed in accordance with 37 CFR 1.31 and 1.33. See 37 CFR 1.4 for signature requirements and certifications.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

IriusRisk
Ex. 1002 - Page 512

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/542,263	08/15/2019	Anurag Agarwal	ThreatM010	7756

145042 7590 03/05/2020

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

EXAMINER	
PARSONS, THEODORE C	
ART UNIT	PAPER NUMBER
2494	

DATE MAILED: 03/05/2020

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(Applications filed on or after May 29, 2000)

The Office has discontinued providing a Patent Term Adjustment (PTA) calculation with the Notice of Allowance.

Section 1(h)(2) of the AIA Technical Corrections Act amended 35 U.S.C. 154(b)(3)(B)(i) to eliminate the requirement that the Office provide a patent term adjustment determination with the notice of allowance. See Revisions to Patent Term Adjustment, 78 Fed. Reg. 19416, 19417 (Apr. 1, 2013). Therefore, the Office is no longer providing an initial patent term adjustment determination with the notice of allowance. The Office will continue to provide a patent term adjustment determination with the Issue Notification Letter that is mailed to applicant approximately three weeks prior to the issue date of the patent, and will include the patent term adjustment on the patent. Any request for reconsideration of the patent term adjustment determination (or reinstatement of patent term adjustment) should follow the process outlined in 37 CFR 1.705.

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

IriusRisk
Ex. 1002 - Page 513

OMB Clearance and PRA Burden Statement for PTOL-85 Part B

The Paperwork Reduction Act (PRA) of 1995 requires Federal agencies to obtain Office of Management and Budget approval before requesting most types of information from the public. When OMB approves an agency request to collect information from the public, OMB (i) provides a valid OMB Control Number and expiration date for the agency to display on the instrument that will be used to collect the information and (ii) requires the agency to inform the public about the OMB Control Number's legal significance in accordance with 5 CFR 1320.5(b).

The information collected by PTOL-85 Part B is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 30 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450. Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

IriusRisk
Ex. 1002 - Page 514

Notice of Allowability	Application No. 16/542,263	Applicant(s) Agarwal, Anurag	
	Examiner THEODORE C PARSONS	Art Unit 2494	AIA (FITF) Status Yes

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☑ This communication is responsive to 2020-02-17.
   ☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2. ☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

3. ☑ The allowed claim(s) is/are 1-20 . As a result of the allowed claim(s), you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see **http://www.uspto.gov/patents/init_events/pph/index.jsp** or send an inquiry to **PPHfeedback@uspto.gov.**

4. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

   Certified copies:

   a) ☐ All b) ☐ Some *c) ☐ None of the:

   1. ☐ Certified copies of the priority documents have been received.
   2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
   3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____ .

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____ .

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☑ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 2020-02-17, 2020-01-01, 2019-11-26.
3. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material _____.
4. ☐ Interview Summary (PTO-413), Paper No./Mail Date. _____.
5. ☐ Examiner's Amendment/Comment
6. ☑ Examiner's Statement of Reasons for Allowance
7. ☐ Other _____.

/THEODORE C PARSONS/
Primary Examiner, Art Unit 2494

DETAILED ACTION

Response to Amendment

1. The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

2. This is in reply to papers filed on 2020-02-17. Claims 1-20 are pending. Claims 1, 8, 16 is/are independent.

3. The objections to the specification are withdrawn in view of Applicant's arguments.

4. The objections to the drawings are withdrawn in view of Applicant's arguments.

5. The rejection(s) of claims under 35 U.S.C. § 112 are withdrawn in view of Applicant's amendments and arguments.

Reasons for Allowance

6. The prior art of record (in particular, U.S. Publication 20150033346 to Hebert et al. (hereinafter "Herbert '346") and U.S. Patent 8413249 to Chou (hereinafter "Chou '249")) does not disclose, with respect to claim 1, "determining using the one or more computing devices, for the subset of third party diagram components, correlated threat model components as defined in the one or more mapping files" in the recited context. Rather, Herbert '346 discloses using an architecture model to generate a threat model by mapping threats to components of the model [Herbert '346 ¶ 0031]; storing mappings in a database [Herbert '346 ¶ 0031]; manual user inputs to refine mappings [Herbert '346 ¶ 0032]; and generating a visual diagram of the threat model [Herbert '346 ¶ 0087-0088, 0097-0100, Fig. 5, Fig. 8]. Similarly, Chou '249

discloses using an architecture model to generate a threat model [Chou '249 col. 6, l. 3-29; Fig. 2C] in which architecture components and relationships are correlated to known vulnerabilities [Chou '249 col. 7, l. 1-44]. However, neither Chou '249 nor Herbert '346 discloses third-party control of the diagram components of the architecture model, wherein "the one or more mapping files correlating the threat model components with visual diagram components of a third party software application" (claim 1).

7. For the reasons described above, the prior art of record does not disclose, with respect to claim(s) 8 and 16, features corresponding to those of claim 1 in the respective context(s).

8. Dependent claims 2-7, 9-15, 17-20 are allowed in view of their respective dependence from claims 1, 8, 16.

Information Disclosure Statement PTO-1449

9. The Information Disclosure Statement(s) submitted by applicant on 2020-02-17, 2020-01-01, 2019-11-26 has/have been considered. The submission is in compliance with the provisions of 37 CFR § 1.97. Form PTO-1449 signed and attached hereto.

Conclusion

10. Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled “Comments on Statement of Reasons for Allowance.”

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to THEODORE C PARSONS whose telephone number is (571)270-1475. The examiner can normally be reached on MTWRF 7:30-4:30.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jung Kim can be reached on (571) 272-3804. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/THEODORE C PARSONS/
Primary Examiner, Art Unit 2494

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	Art Unit
	THEODORE C PARSONS	2494

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

CLAIMS

☑ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIM		DATE								
Final	Original	10/28/2019	03/01/2020							
	1	✓	=							
	2	✓	=							
	3	✓	=							
	4	✓	=							
	5	✓	=							
	6	✓	=							
	7	✓	=							
	8	✓	=							
	9	✓	=							
	10	✓	=							
	11	✓	=							
	12	✓	=							
	13	✓	=							
	14	✓	=							
	15	✓	=							
	16	✓	=							
	17	✓	=							
	18	✓	=							
	19	✓	=							
	20	✓	=							

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	Art Unit
	THEODORE C PARSONS	2494

CPC - Searched*		
Symbol	Date	Examiner
G06F21/577; G06F16/2282; G06F16/248; G06F9/451; G06F16/212; G06F2221/033	2019-11-12	TCP
G06F21/577; G06F16/2282; G06F16/248; G06F9/451; G06F16/212; G06F2221/033	2020-03-01	TCP

CPC Combination Sets - Searched*		
Symbol	Date	Examiner

US Classification - Searched*			
Class	Subclass	Date	Examiner

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

/THEODORE C PARSONS/ Primary Examiner, Art Unit 2494	

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	**Art Unit**
	THEODORE C PARSONS	2494

Search Notes		
Search Notes	**Date**	**Examiner**
inventor, assignee search	2019-10-28	TCP
EAST (see notes attached)	2019-10-28	TCP
Google Patents Sigma (see notes attached)	2019-10-28	TCP
IP.com InnovationQ (see notes attached)	2019-10-28	TCP
Google Scholar (see notes attached)	2019-11-12	TCP
inventor, assignee search	2019-11-12	TCP
EAST (see notes attached)	2019-11-12	TCP
foreign counterpart search	2019-11-12	TCP
inventor, assignee search	2020-03-01	TCP
EAST (see notes attached)	2020-03-01	TCP

Interference Search			
US Class/CPC Symbol	**US Subclass/CPC Group**	**Date**	**Examiner**
	PGPUB text search via EAST (see interference notes attached)	2019-11-12	TCP
	PGPUB text search via EAST (see interference notes attached)	2020-03-01	TCP

/THEODORE C PARSONS/ Primary Examiner, Art Unit 2494	

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	**Art Unit**
	THEODORE C PARSONS	2494

CPC						
Symbol					**Type**	**Version**
G06F	/	21	/	577	F	2013-01-01
G06F	/	16	/	212	I	2019-01-01
G06F	/	16	/	248	I	2019-01-01
G06F	/	9	/	451	I	2018-02-01
G06F	/	16	/	2282	I	2019-01-01
G06F	/	2221	/	033	A	2013-01-01

CPC Combination Sets							
Symbol					**Type**	**Set**	**Ranking**	**Version**
	/		/					

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	20	
/THEODORE C PARSONS/ Primary Examiner, Art Unit 2494	01 March 2020	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	30

U.S. Patent and Trademark Office

Part of Paper No.: 20200301

IriusRisk
Ex. 1002 - Page 522

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	**Art Unit**
	THEODORE C PARSONS	2494

INTERNATIONAL CLASSIFICATION				
CLAIMED				
G06F	/	21	/	57

NON-CLAIMED				
	/		/	

US ORIGINAL CLASSIFICATION	
CLASS	**SUBCLASS**

CROSS REFERENCES(S)						
CLASS	**SUBCLASS (ONE SUBCLASS PER BLOCK)**					

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	20	
/THEODORE C PARSONS/ Primary Examiner, Art Unit 2494	01 March 2020	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	30

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	16/542,263	Agarwal, Anurag
	Examiner	**Art Unit**
	THEODORE C PARSONS	2494

☑ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIMS

Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original
	1		10		19										
	2		11		20										
	3		12												
	4		13												
	5		14												
	6		15												
	7		16												
	8		17												
	9		18												

NONE		**Total Claims Allowed:**	
(Assistant Examiner)	(Date)	20	
/THEODORE C PARSONS/ Primary Examiner, Art Unit 2494	01 March 2020	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	30

U.S. Patent and Trademark Office

Part of Paper No.: 20200301

IriusRisk
Ex. 1002 - Page 524

Bibliographic Data

Application No: 16/542,263

Foreign Priority claimed: ○ Yes ◉ No

35 USC 119 (a-d) conditions met: ☐ Yes ☑ No ☐ Met After Allowance

Verified and Acknowledged: /THEODORE C PARSONS/ (Examiner's Signature) [ ] (Initials)

Title: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

FILING or 371(c) DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
08/15/2019	726	2494	ThreatM010
RULE			

APPLICANTS

ThreatModeler Software Inc., Jersey City, NJ, UNITED STATES

INVENTORS

Anurag Agarwal Jersey City, NJ, UNITED STATES

CONTINUING DATA

This application is a CIP of 16228738 12/20/2018

16228738 is a CIP of 15922856 03/15/2018 PAT 10200399

15922856 is a CIP of 15888021 02/03/2018 PAT 10255439

15888021 has PRO of 62530295 07/10/2017

15888021 has PRO of 62527671 06/30/2017

15888021 has PRO of 62520954 06/16/2017

15888021 has PRO of 62507691 05/17/2017

FOREIGN APPLICATIONS

IF REQUIRED, FOREIGN LICENSE GRANTED**

08/26/2019

**** SMALL ENTITY ****

STATE OR COUNTRY

UNITED STATES

ADDRESS

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

FILING FEE RECEIVED

$2,855

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S86	31	("7433829" | "7900259" | "8407801" | "20140157417" | "20160162690" | "7260844" | "8413237" | "20150033346" | "20180255084" | "20060015941" | "20110178942" | "20170213037" | "10262132" | "7624448" | "20060241991" | "20140137257" | "8413249" | "9043924" | "20010027388" | "20110126111" | "9774613" | "20160248798" | "4959015" | "9497203" | "20110277034" | "7096502" | "7891003" | "6952779" | "8413249" | "20180324207" | "8191139" | "8255995").PN.	US-PGPUB; USPAT	OR	ON	2020/03/01 08:08
S87	20	("7433829" | "7900259" | "8407801" | "8413237" | "20060015941" | "20110178942" | "20060241991" | "20140137257" | "8413249" | "9043924" | "20110126111" | "9774613" | "4959015" | "9497203" | "7096502" | "7891003" | "6952779" | "20180324207" | "8191139" | "8255995").PN.	US-PGPUB; USPAT	OR	ON	2020/03/01 08:08
S88	11	S86 not S87	US-PGPUB; USPAT	OR	ON	2020/03/01 08:08
S89	4	(US-20140157417-$ or US-20110277034-$ or US-20160248798-$).did. or (US-8413249-$).did.	US-PGPUB; USPAT	OR	ON	2020/03/01 08:08
S90	8	S88 not S89	US-PGPUB; USPAT	OR	ON	2020/03/01 08:08
S91	14	(09/853690 | 10/655062 | 11/367943 | 13/951837 | 14/958792 | 15/201186 | 15/326991 | 15/909786).APP.	US-PGPUB; USPAT; USOCR	OR	ON	2020/03/01 08:09
S92	12	S91 and (((shelf third 3rd independent separate visio existing cots! commercial$4 conventional) near20 (model$4 diagram architecture component format)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:09
S93	5	S91 and ((map mapping mapped correlat$4) same (((attack$4 threat vulnerability risk) near4 (model$4 diagram report)) near20 (component asset visio vsd vsdx shape)) same ((attack$4 threat vulnerability risk) (model$4 diagram report) (component asset visio vsd vsdx shape) (architecture model modelling) (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:09
S94	12	S91 and (((third 3rd independent	US-	OR	ON	2020/03/01

		separate visio existing known cots! commercial$4 available conventional) near20 (model$4 diagram architecture component format)))	PGPUB; USPAT			08:10
S95	5	S91 and ((((attack$4 threat vulnerability risk) near4 (model$4 diagram report)) near20 (component asset visio vsd vsdx shape)) same ((attack$4 threat vulnerability risk) (model$4 diagram report) (component asset visio vsd vsdx shape) (architecture model modelling) (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:10
S96	1	(US-20150033346-$).did.	US-PGPUB; USPAT	OR	ON	2020/03/01 08:13
S97	117164	((visual depict$5 graphical viso shape) near20 (software system process network application) near20 (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S98	35372	(G06F21/577 or G06F16/2282 or G06F16/248 or G06F9/451 or G06F16/212 or G06F2221/033).cpc.	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S99	149705	S98 S97	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S100	149685	S99 and (@pd<"20190901" @ad<"20190901" @rlad<"20190901")	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S101	138528	S99 and ((model$4 visual depict$5 graphical viso shape) near12 (software system process network application))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S102	112666	S99 and ((visual depict$5 graphical viso shape) near12 (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S103	112666	S99 and ((visual depict$5 graphical viso shape) near12 (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S104	5892	S99 and ((threat vulnerability risk) near12 (model$4 visual depict$5 graphical viso shape))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S105	17104	S99 and ((threat vulnerability risk) near12 (software system process network application))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S106	2717	S99 and ((threat vulnerability risk) near12 (visual depict$5 graphical viso shape))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S107	2149	S99 and ((threat vulnerability risk) same (visual depict$5 graphical viso shape) same (software system process network application) same (architecture model modelling))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S108	2240	S99 and (((correspond$4 map mapping mapped correlat$4 associat$5) near12 (attack$4 threat vulnerability risk)) same (visual depict$5 graphical viso shape))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S109	3243	S99 and (((map mapping mapped	US-	OR	ON	2020/03/01

		correlat$4) near20 (attack$4 threat vulnerability risk)))	PGPUB; USPAT			08:22
S110	7496	S99 and ((attack$4 threat vulnerability risk).ab,ti,clm.)	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S111	43196	S99 and ((visual$4 visualiz$5 visualis$5 depict$5 graphical viso shape).ab,ti,clm.)	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S112	92	(S110 and S111) and S108 and S109 and S100 and S101 and S102 and S103 and S104 and S105 and S106 and S107	US-PGPUB; USPAT	OR	ON	2020/03/01 08:22
S113	28	(09/500269 | 12/775120 | 12/895847 | 13/413586 | 13/559692 | 13/559707 | 13/559732 | 13/690517 | 14/078514 | 14/623333 | 15/086330 | 15/554553 | 15/888021 | 15/922856 | 15/940604 | 16/130352 | 16/228738 | 16/270276 | 16/278957).APP.	US-PGPUB; USPAT; USOCR	OR	ON	2020/03/01 08:22
S114	92	S113 S112	US-PGPUB; USPAT; USOCR	OR	ON	2020/03/01 08:22
S115	6	S114 and (@pd>"20191101" @ad>"20191101")	US-PGPUB; USPAT	OR	ON	2020/03/01 08:36
S116	6	S115 and (((correspond$4 map mapping mapped correlat$4 associat$5) near12 ((diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape) (attack$4 threat vulnerability risk))) same (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 visio vsd vsdx shape) same (attack$4 threat vulnerability risk) same ((attack$4 threat vulnerability risk) (architecture model modelling)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:36
S117	6	S115 and ((diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 viso shape) same (attack$4 threat vulnerability risk) same ((architecture model modelling) (map mapping mapped correlat$4)) same ((attack$4 threat vulnerability risk) (correspond$4 map mapping mapped correlat$4 associat$5)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:36
S118	5	S115 and ((map mapping mapped) same (diagram visual$4 visualiz$5 visualis$5 depict$5 graphical$3 viso shape) same (attack$4 threat vulnerability risk) same ((attack$4 threat vulnerability risk) (architecture model modelling) (correspond$4 map mapping mapped correlat$4 associat$5)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:36

EAST Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S119	1	"Term Removed"	US-PGPUB	OR	ON	2020/03/01 08:08
S120	82	(architecture model modelling).clm. same	US-	OR	ON	2020/03/01

		(component asset).clm. same (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm. same (threat vulnerability risk attack$4).clm.	PGPUB; USPAT			08:11
S121	27	S120 and ((map mapping mapped correlat$4 correspond$3).clm. near12 ((threat vulnerability risk attack$4).clm.))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:11
S122	18	S120 and ((map mapping mapped correlat$4 correspond$3).clm. near12 ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm.))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:11
S123	16	S122 and S121	US-PGPUB; USPAT	OR	ON	2020/03/01 08:11
S124	4	S123 and (@pd>"20191101" @ad>"20191101")	US-PGPUB; USPAT	OR	ON	2020/03/01 08:11
S125	6	"Term Removed"	US-PGPUB; USPAT	OR	ON	2020/03/01 08:13
S126	6	S125 and (((architecture model modelling).clm. same (threat vulnerability risk attack$4).clm. same ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm.)) or ((map mapping mapped correlat$4 correspond$3).clm. near12 ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm. (threat vulnerability risk attack$4).clm.)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:13
S127	58	((("AGARWAL") near3 ("Anurag"))).INV.	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S128	6	(THREATMODELER).AS,AANM.	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S129	58	S127 S128	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S130	4	S129 and ((map mapping mapped).clm.)	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S131	13	S129 and ((model$4 visual depict$5 graphic visio template component shape connector).clm. near12 ((correspond$4 map mapping mapped correlat$4 associat$5).clm. (threat vulnerability risk attack$4).clm.))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S132	11	S129 and ((threat vulnerability risk attack$4).clm. near12 (correspond$4 map mapping mapped correlat$4 associat$5).clm.)	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S133	10	S131 and (S130 S132)	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S134	2	S133 and (@pd>"20191101" @ad>"20191101")	US-PGPUB; USPAT	OR	ON	2020/03/01 08:14
S135	0	"Term Removed"	USPAT	OR	ON	2020/03/01 08:15

S136	2	"Term Removed"	US-PGPUB; USPAT	OR	ON	2020/03/01 08:15
S137	2	S136 and ((correspond$4 map mapping mapped correlat$4 associat$5).clm. near12 ((model$4 visual depict$5 graphic visio template component shape connector).clm. (threat vulnerability risk attack$4).clm.))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:15
S138	2	S136 and (((architecture model modelling).clm. same (threat vulnerability risk attack$4).clm. same ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm.)) or ((map mapping mapped correlat$4 correspond$3).clm. near12 ((component asset).clm. (diagram visio vsd vsdx shape visual visualiz$5 depict$5 graphic$3).clm. (threat vulnerability risk attack$4).clm.)))	US-PGPUB; USPAT	OR	ON	2020/03/01 08:15

3/1/2020 9:06:33 AM
C:\Users\tparsons1\Documents\EAST\Workspaces\16542263-track1.wsp

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	THREATM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20170213037	A1	2017-07-27	Toledano et al.	
	2	20180255084	A1	2018-09-06	Kotinas et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

IriusRisk
Ex. 1002 - Page 532

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P./

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	THREATM010

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1		

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature	/THEODORE C PARSONS/	Date Considered	03/01/2020

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	THREATM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☒ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2019-11-25
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20010027388	A1	2001-10-04	Beverina et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /T.C.P./

IriusRisk
Ex. 1002 - Page 536

Receipt date: 01/27/2020

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	16542263
	Filing Date	2019-08-15
	First Named Inventor	Anurag Agarwal
	Art Unit	2494
	Examiner Name	Theodore C. Parsons
	Attorney Docket Number	ThreatM010

	1	Roy, Maurer. "Top Database Security Threats and How to Mitigate Them." July 30, 2015. Retrieved from "https://www.shrm.org/resouresandtools/hr-topics/risk-management/pages/top-database-security-threats.aspx" (Year: 2015).
	2	A. Amini, N. Jamil, A.R. Ahmad and M.R. Z'aba, 2015. Threat Modeling Approaches for Securing Cloud Computing. Journal of Applied Sciences, 15: 953-967. Retrieved from "https://scialert.net/fulltextmobile/?doi=jas.2015.953.967" (Year: 2015).

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature	/THEODORE C PARSONS/	Date Considered	03/01/2020

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☒ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2020-01-27
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (02-18)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	10262132	B2	2019-04-16	Reinecke et al.	
	2	7624448	B2	2009-11-24	Coffman	
	3	7260844	B1	2007-08-21	Tidwell et al.	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20160162690	A1	2016-06-09	Reith et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

If you wish to add additional Foreign Patent Document citation information please click the Add button [Add]

NON-PATENT LITERATURE DOCUMENTS [Remove]

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1		

If you wish to add additional non-patent literature document citation information please click the Add button [Add]

EXAMINER SIGNATURE

Examiner Signature	/THEODORE C PARSONS/	Date Considered	03/01/2020

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	16542263
Filing Date	2019-08-15
First Named Inventor	Anurag Agarwal
Art Unit	2494
Examiner Name	Theodore C. Parsons
Attorney Docket Number	ThreatM010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☒ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Paul B. Johnson/	Date (YYYY-MM-DD)	2020-01-01
Name/Print	Paul B. Johnson	Registration Number	59375

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Mar 05, 2020 03:56:48 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	NOA	03/05/2020	ThreatM010
	1449	03/05/2020	ThreatM010
	1449	03/05/2020	ThreatM010
	1449	03/05/2020	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

CORRESPONDENCE

Applicant: ThreatModeler Software Inc.

Docket No: THREATM010

Examiner: Theodore C. Parsons

Ser. No: 16542263

Filed: August 15, 2019

Group Art Unit: 2494

Title: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

COMMENTS ON STATEMENT OF REASONS FOR ALLOWANCE

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Examiner:

The following is responsive to the Notice of Allowance mailed Mar. 5, 2020. Comments on the Examiner's stated reasons for allowance are given in the following section.

Remarks begin on page 2 of this paper.

IriusRisk
Ex. 1002 - Page 545

REMARKS

Claims 1-20 have been allowed in the application. Applicant wishes to thank the Examiner for confirmation of the patentable subject matter of these claims.

Comments on Examiner's Reasons for Allowance

The Applicant has reviewed the Examiner's Reasons for Allowance in the Notice of Allowance referenced above. It is pointed out that, while the examiner's stated reasons state "neither Chou '249 nor Herbert '346 discloses third-party control of the diagram components of the architecture model," that nor do the allowed claims require "third-party control" of the diagram components. As indicated on p. 8 of the applicant's comments filed Feb. 17, 2020, the term "third-party" in the claims does "not necessarily indicat[e] ownership of software applications (but rather indicate[s] independently-created software applications not created with a common schema, such that a mapping file or the like as claimed in the claims would be useful to map between the two software applications)." The claims do not recite nor do they require any "third-party control" whatsoever, of diagram components or any other element.

Nor do the claims recite or require an "architecture model" or any model specifically related to the field of architecture.

The Applicant respectfully notes that there may be additional reasons for allowance that have not been specifically cited, and which may apply to various of the allowed claims, in addition to or instead of the cited Reasons. Applicant respectfully suggests that notwithstanding the Examiner's Reasons for Allowance, it is believed that each of the allowed claims is patentable in its own right and/or for other reasons than those given by the examiner and/or for reasons raised during the prosecution and/or for reasons explained in the specification and/or for reasons not raised during prosecution and/or for reasons not explained in the specification. To the extent that any statements regarding patentability of

IriusRisk
Ex. 1002 - Page 546

any claim(s) allowed by the Examiner made by the Applicant or the Examiner in any document filed in this application are inconsistent with or not included in the Examiner's Reasons for Allowance, they are incorporated by reference herein.

The statements with regard to what is disclosed in the prior art references are not admitted by applicant.

Respectfully submitted,

Date: March 31, 2020

By: /Paul B. Johnson/
Paul B. Johnson
Reg. No. 59375

Law Office of Paul B. Johnson
8483 S 1275 E
Sandy, UT 84094-1363
Ph: 801.719.5229
Fax: 801.317.8889
paul@pauljohnsonlaw.com

IriusRisk
Ex. 1002 - Page 547

Electronic Acknowledgement Receipt

EFS ID:	39026487
Application Number:	16542263
International Application Number:	
Confirmation Number:	7756
Title of Invention:	Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files
First Named Inventor/Applicant Name:	Anurag Agarwal
Customer Number:	145042
Filer:	Paul B. Johnson
Filer Authorized By:	
Attorney Docket Number:	ThreatM010
Receipt Date:	31-MAR-2020
Filing Date:	15-AUG-2019
Time Stamp:	18:21:40
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Miscellaneous Incoming Letter	PATCommentsOnReasonsForAllowance.pdf	192943 aef6f12f638b910a7bed87671c8414a3a9b3dcaa	no	3

Warnings:

IriusRisk

Information:	
Total Files Size (in bytes):	192943

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

FEE ADDRESS CHANGE

APPLICATION #	FILING OR 371(C) DATE	REQUEST ID
16/542,263	08/15/2019	19300771

The following fields have been changed to Customer Number 145042 on 06/05/2020 via Patent Center in view of the certification copied below that authorized the change.

- Maintenance Fee Address

The address for Customer Number 145042 is:

145042 - Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UTAH 84094-1363

I certify, in accordance with 37 CFR 1.4(d)(4), that I am :

An attorney or Agent of Record registered to practice before the Patent and Trademark Office who has been given power of attorney in this application

Signature	**Name**	**Registration #**

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

FEE ADDRESS INDICATION FORM

APPLICATION #	ATTORNEY DOCKET #	FILING DATE
16542263	ThreatM010	08/15/2019

Title of Invention

Systems &Methods for Automated Threat Model Generation from Third Party Diagram Files

Application Information

ART UNIT 2494 **FIRST NAMED INVENTOR** Anurag Agarwal

EXAMINER THEODORE PARSONS

INSTRUCTIONS: In order for the fee address identified on this form to be effective, the issue fee must have been paid for the application listed on this form. Only an address represented by a customer number can be established as the fee address for maintenance fee purposes (hereafter, fee address). A fee address should be established when correspondence related to maintenance fees should be mailed to a different address than the correspondence address for the application. For more information on customer numbers, see the Manual of Patent Examining Procedure (MPEP) § 403

Maintenance Fee Address

Please recognize as the Fee Address under the provisions of 37 CFR 1.363 the address associated with:

145042 - Law Office of Paul B. Johnson

8483 S. 1275 E.
--
Sandy, UT 84094-1363
UNITED STATES

Signature

I certify, in accordance with 37 CFR 1.4(d)(4) that I am an attorney or agent registered to practice before the Patent and Trademark Office who has filed and has been granted power of attorney in this application.

Signature	Name	Registration #	Date
/Paul B. Johnson/	Paul Johnson	59375	06/05/2020

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
16/542,263	06/05/2020 03:28:29 PM ET	ThreatM010

Title of Invention

Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	7756	FILED BY	Paul Johnson
PATENT CENTER #	60033642	FILING DATE	08/15/2019
CUSTOMER #	145042	FIRST NAMED INVENTOR	Anurag Agarwal
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	-

Documents

TOTAL DOCUMENTS: 2

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
web85b.pdf	3	Issue Fee Payment (PTO-85B)	62 KB
web85bfeeaddress.pdf	1	Maintenance Fee Address Change	27 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
web85b.pdf	0655626DED2C5EDD0C490BC612EC4664EEF0BF7E9F363C21 5C8C38F54EBDD66AAFDEE73103B92C3E7C9A3F95197F200C 120270A8C4B7419217B9A3CB9BDDB543

web85bfeeaddress.pdf 3C1C9B5275686CA7361BCC2282847F6BA3DD7B160B4818C592502F83865B4097B2E900BEECFD2ECD298C17D9CA9828A05B42CE13D8391936268AC1067A8BC085

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
16/542,263	06/05/2020 03:28:29 PM ET	ThreatM010

Title of Invention

Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	7756	FILED BY	Paul Johnson
PATENT CENTER #	60033642	AUTHORIZED BY	-
CUSTOMER #	145042	FILING DATE	08/15/2019
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Anurag Agarwal

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
CARD / 6354	E202065F28563163	Paul Johnson

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
2501	UTILITY ISSUE FEE	500.00	1	500.00
			TOTAL AMOUNT:	$500.00

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage

submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ISSUE FEE TRANSMITTAL FORM

APPLICATION #	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET #
16542263	08/15/2019	Anurag Agarwal	ThreatM010

Title of Invention

Systems &Methods for Automated Threat Model Generation from Third Party Diagram Files

Application Information

APPLICATION TYPE	Nonprovisional Application under 35 USC 111(a)	DATE DUE	06/05/2020
CONFIRMATION #	7756	ISSUE FEE DUE	$ 500
EXAMINER	THEODORE PARSONS	PUBLICATION DUE	$ 0
GROUP ART UNIT	2494	TOTAL FEES DUE	$500
CLASS - SUBCLASS	726/025000	PREV. PAID FEE	$ 0
ENTITY STATUS	Small		

1. CHANGE OF CORRESPONDENCE ADDRESS AND/OR INDICATION OF FEE ADDRESS (37 CFR 1.33 & 1.363)

Correspondence Address

CURRENT ADDRESS

Maintenance Fee Address

CURRENT ADDRESS

☐ Change of correspondence address requested,system generated AIA/122-PC form attached

☑ Fee address indication requested, system generatedSB/47-PC form attached

2. ENTITY STATUS

Change in Entity Status

NEW STATUS

3. THE FOLLOWING FEES ARE SUBMITTED

☑ Issue Fee

☑ Publication Fee

☐ Advance Order - # of copies:

4. FEE AUTHORIZATION

☐ I authorize USPTO to apply my previously paid issue fee to the current fees due

☐ The Director is herby authorized to apply my previously paid issue fee to the current fee due and to charge deficient fees to Deposit Account Number:

☐ If in addition to the payment of the issue fee amount submitted with this form, there are any discrepencies in any amount(s) due, the Director is authorized to charge any deficiency, or credit any overpayment, to Deposit Account Number:

5. FIRM AND/OR ATTORNEY NAMES TO BE PRINTED

NOTE: If no name is listed, no name will be printed
For printing on the patent front page, list to be displayed as entered

Law Office of Paul B. Johnson
Paul B. Johnson

6. ASSIGNEE NAME(S) AND RESIDENCE DATA TO BE PRINTED

NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a susbstitute for filing an assignment.

ThreatModeler Software Inc. Jersey City, NJ UNITED STATES Organization

Signature

I certify, in accordance with 37 CFR 1.4(d)(4) that I am an attorney or agent registered to practice before the Patent and Trademark Office who has filed and has been granted power of attorney in this application. I also

certify that this Fee(s) Transmittal form is being transmitted to the USPTO via Patent Center on the date indicated below.

Signature	Name	Registration #	Date
/Paul B. Johnson/	Paul Johnson	59375	06/05/2020

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/542,263	07/14/2020	10713366	ThreatM010	7756

145042 7590 06/24/2020

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 0 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Application Assistance Unit (AAU) of the Office of Data Management (ODM) at (571)-272-4200.

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Anurag Agarwal, Jersey City, NJ;
ThreatModeler Software Inc., Jersey City, NJ;

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The USA offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to encourage and facilitate business investment. To learn more about why the USA is the best country in the world to develop technology, manufacture products, and grow your business, visit SelectUSA.gov.

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Jun 25, 2020 03:31:38 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	ISSUE.NTF	06/24/2020	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

PTO/SB/44 (09-07)
Approved for use through 01/31/2020. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.
(Also Form PTO-1050)

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 10713366

APPLICATION NO.: 16542263

ISSUE DATE : July 14, 2020

INVENTOR(S) : Anurag Agarwal

It is certified that an error appears or errors appear in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the drawings:

In FIG. 10, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.
In FIG. 21, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.
In FIG. 22, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.

In the specification:

Column 5, line 64, change "compensated" to -compensating-.
Column 6, line 58, change "compensatory" to -compensating-.
Column 15, line 29, change "showing" to -shows-.
Column 16, line 60, change "a cloud computing" to -cloud computing-.
Column 16, line 44, change "or a subset" to -(or a subset-.
Column 17, line 53, change "new link drawn" to -new link is drawn-.
Column 17, line 54, change "protocol selected" to -a protocol is selected-.
Column 17, line 59, change "popup threat" to -popup threat report-.
Column 18, lines 26-27, change "a similar" to -an-.
Column 18, line 27, change "to the" to -similar to the-.
Column 18, line 62, change "what" to -that-.
Column 18, line 63, change "list" to -lists-.
Column 20, line 29, change "the component" to -component-.
Column 20, line 35 change "ID a version" to -a CPE ID indicating a version-.
Column 21, line 1, change "That" to -The-.
Column 22, line 31, change "drop downs" to -dropdowns-.
Column 23, line 41, change "to" to -to/from-.
Column 24, line 17, change "some" to -a-.

MAILING ADDRESS OF SENDER (Please do not use Customer Number below):

Law Office of Paul B. Johnson
8483 S 1275 E, Sandy, UT 84094

This collection of information is required by 37 CFR 1.322, 1.323, and 1.324. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1.0 hour to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Attention Certificate of Corrections Branch, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

PTO/SB/44 (09-07)
Approved for use through 01/31/2020. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.
(Also Form PTO-1050)

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 10713366

APPLICATION NO.: 16542263

ISSUE DATE : July 14, 2020

INVENTOR(S) : Anurag Agarwal

It is certified that an error appears or errors appear in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the specification:

Column 24, line 18, change "some" to -an-.
Column 24, line 20-21, change "the asset may be sensitive client information stored within the database" to -sensitive client information may be stored within the database-.
Column 27, line 24, change "analyze" to -analyzing-.
Column 27, line 26, change "analyze" to -analyzing-.
Column 27, lines 27-28, change "measure and quantify" to -measuring and quantifying-.
Column 29, line 11, change "threat" to -threat)-.
Column 29, line 43, the number 1 should not be bolded.
Column 34, line 7, change "functionality" to -the web page having functionality-.
Column 34, line 57, change "is" to -includes-.
Column 37, line 51, change "that" to -that is-.
Column 37, line 52, change "similar elements" to -similar elements to the table of interface 2800-.
Column 42, lines 51-52, change "SQL SERVER" to -SQL DATABASE-.

MAILING ADDRESS OF SENDER (Please do not use Customer Number below):

Law Office of Paul B. Johnson
8483 S 1275 E, Sandy, UT 84094

This collection of information is required by 37 CFR 1.322, 1.323, and 1.324. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1.0 hour to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Attention Certificate of Corrections Branch, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
16/542,263	07/22/2020 10:17:04 PM ET	ThreatM010

Title of Invention

Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	10713366
CONFIRMATION #	7756	FILED BY	Paul Johnson
PATENT CENTER #	60057969	FILING DATE	08/15/2019
CUSTOMER #	145042	FIRST NAMED INVENTOR	Anurag Agarwal
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	-

Documents

TOTAL DOCUMENTS: 2

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
CertOfCorrectionRequestLtr.pdf	2	Miscellaneous Incoming Letter	120 KB
CertOfCorrectionRequest.pdf	2	Request for Certificate of Correction	126 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
CertOfCorrectionRequestLtr.pdf	D3FD12354A8C26BCCB780D673E4F55F76DB5D664936EEA25A386E9B3AB45F90A1C40FC42EF5CF740F81CA3C30E257AA2

	D5D2836BCDFF70F41BF0C99FD08A6B0D
CertOfCorrectionRequest.pdf	5697B4364AA56053C1A948DDB922D3B98760AAA50953D85824 097B1C0ABCCA197FD3C401501CC53966E149F3A5E13A634B2 CA7D217D7B374E277CDAFE6BA30B1

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
16/542,263	07/22/2020 10:17:04 PM ET	ThreatM010

Title of Invention

Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	10713366
CONFIRMATION #	7756	FILED BY	Paul Johnson
PATENT CENTER #	60057969	AUTHORIZED BY	-
CUSTOMER #	145042	FILING DATE	08/15/2019
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Anurag Agarwal

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
CARD / 6354	E20207LM17546740	Paul Johnson

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
2811	CERTIFICATE OF CORRECTION	150.00	1	150.00
			TOTAL AMOUNT:	$150.00

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PAUL B. JOHNSON
INTELLECTUAL PROPERTY LAW

Paul B. Johnson
Registered Patent Attorney
paul@pauljohnsonlaw.com

July 22, 2020

Commissioner for Patents
Office of Data Management Attention: Certificates of Correction Branch
P.O. Box 1450
Alexandria, VA 22313-1450

Certificate of Correction Request

Patent Number: 10713366
Application Number: 16542263
Inventor: Anurag Agarwal
Issue Date: July 14, 2020
Title of Invention: Systems & Methods for Automated Threat Model Generation from Third Party Diagram Files

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the drawings:

In FIG. 10, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.
In FIG. 21, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.
In FIG. 22, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.

In the specification:

Column 5, line 64, change "compensated" to -compensating-.
Column 6, line 58, change "compensatory" to -compensating-.
Column 15, line 29, change "showing" to -shows-.
Column 16, line 60, change "a cloud computing" to -cloud computing-.
Column 16, line 44, change "or a subset" to -(or a subset-.
Column 17, line 53, change "new link drawn" to -new link is drawn-.
Column 17, line 54, change "protocol selected" to -a protocol is selected-.
Column 17, line 59, change "popup threat" to -popup threat report-.
Column 18, lines 26-27, change "a similar" to -an-.

8483 S 1275 E, Sandy, UT 84094-1363
Phone: 801.719.5229 Fax: 801.317.8869

Column 18, line 27, change “to the” to -similar to the-.
Column 18, line 62, change “what” to -that-.
Column 18, line 63, change “list” to -lists-.
Column 20, line 29, change “the component” to -component-.
Column 20, line 35 change “ID a version” to -a CPE ID indicating a version-.
Column 21, line 1, change “That” to -The-.
Column 22, line 31, change “drop downs” to -dropdowns-.
Column 23, line 41, change “to” to -to/from-.
Column 24, line 17, change “some” to -a-.
Column 24, line 18, change “some” to -an-.
Column 24, line 20-21, change “the asset may be sensitive client information stored within the database” to -sensitive client information may be stored within the database-.
Column 27, line 24, change “analyze” to -analyzing-.
Column 27, line 26, change “analyze” to -analyzing-.
Column 27, lines 27-28, change “measure and quantify” to -measuring and quantifying-.
Column 29, line 11, change “threat” to -threat)-.
Column 29, line 43, the number 1 should not be bolded.
Column 34, line 7, change “functionality” to -the web page having functionality-.
Column 34, line 57, change “is” to -includes-.
Column 37, line 51, change “that” to -that is-.
Column 37, line 52, change “similar elements” to -similar elements to the table of interface 2800-.
Column 42, lines 51-52, change “SQL SERVER” to -SQL DATABASE-.

Statement of Facts Supporting the Requested Corrections

The corrections made herein are to correct grammatical errors. Since at least one of the errors was in the application as originally filed, the appropriate fee is submitted herewith. Form PTO/SB/44 is included with this correspondence. Please contact me if you have any questions.

Sincerely,

Paul

8483 S 1275 E. Sandy, UT 84094-1363
Phone: 801.719.5229 Fax: 801.317.8869

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/542,263	08/15/2019	Anurag Agarwal	ThreatM010	7756

145042 7590 08/10/2020

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363

EXAMINER
PARSONS, THEODORE C

ART UNIT	PAPER NUMBER
2494	

NOTIFICATION DATE	DELIVERY MODE
08/10/2020	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

paul@pauljohnsonlaw.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

Patent No.: 10713366
Issue Date: 07/14/2020
Appl. No.: 16/542,263
Filed: 08/15/2019

PART (A) RESPONSE FOR CERTIFICATES OF CORRECTION

This is a decision on the Certificate of Correction request filed 22 July 2020.

The request for issuance of Certificate of Correction for the above-identified correction(s) under the provisions of 37 CFR 1.322 and/or 1.323 is hereby:

(Check one)
☑ Approved ☐ Approved in Part ☐ Denied

Comments: ____

PART (B) PETITION UNDER 37 CFR 1.324 OR 37 CFR 1.48

☐ This is a decision on the petition filed ____ to correct inventorship under 37 CFR 1.324.

☐ This is a decision on the request under 37 CFR 1.48, petition filed ____. In view of the fact that the patent has already issued, the request under 37 CFR 1.48 has been treated as a petition to correct inventorship under 37 CFR 1.324.

The petition is hereby: ☐ Granted ☐ Dismissed

Comment: ____

The patented filed is being forwarded to Certificate of Corrections Branch for issuance of a certificate naming only the actual inventor or inventors.

/JUNG W KIM/
Supervisory Patent Examiner, Art Unit 2494
Technology Center 2400
Phone: (571)272-3804

Certificates of Correction Branch email: CustomerServiceCoC@uspto.gov CoC Central Phone Number: (703) 756-1814

To: paul@pauljohnsonlaw.com,,
From: PAIR_eOfficeAction@uspto.gov
Cc: PAIR_eOfficeAction@uspto.gov
Subject: Private PAIR Correspondence Notification for Customer Number 145042

Aug 11, 2020 03:41:46 AM

Dear PAIR Customer:

Law Office of Paul B. Johnson
8483 S. 1275 E.
Sandy, UT 84094-1363
UNITED STATES

The following USPTO patent application(s) associated with your Customer Number, 145042 , have new outgoing correspondence. This correspondence is now available for viewing in Private PAIR.

The official date of notification of the outgoing correspondence will be indicated on the form PTOL-90 accompanying the correspondence.

Disclaimer:
The list of documents shown below is provided as a courtesy and is not part of the official file wrapper. The content of the images shown in PAIR is the official record.

Application	Document	Mailroom Date	Attorney Docket No.
16542263	COCX	08/10/2020	ThreatM010

To view your correspondence online or update your email addresses, please visit us anytime at https://sportal.uspto.gov/secure/myportal/privatepair.

If you have any questions, please email the Electronic Business Center (EBC) at EBC@uspto.gov with 'e-Office Action' on the subject line or call 1-866-217-9197 during the following hours:

Monday - Friday 6:00 a.m. to 12:00 a.m.

Thank you for prompt attention to this notice,

UNITED STATES PATENT AND TRADEMARK OFFICE

PATENT APPLICATION INFORMATION RETRIEVAL SYSTEM

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 10,713,366 B2
APPLICATION NO. : 16/542263
DATED : July 14, 2020
INVENTOR(S) : Anurag Agarwal

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the Drawings

In FIG. 10, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.
In FIG. 21, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.
In FIG. 22, the rightmost line connecting the "AKAMAI DNS" component with the "FIRE WALL" component should have the end arrow pointing at the "FIRE WALL" component.

In the Specification

Column 5, Line 64, change "compensated" to -compensating-.
Column 6, Line 58, change "compensatory" to -compensating-.
Column 15, Line 29, change "showing" to -shows-.
Column 16, Line 60, change "a cloud computing" to -cloud computing-.
Column 16, Line 44, change "or a subset" to -(or a subset-.
Column 17, Line 53, change "new link drawn" to -new link is drawn-.
Column 17, Line 54, change "protocol selected" to -a protocol is selected-.
Column 17, Line 59, change "popup threat" to -popup threat report-.
Column 18, Lines 26-27, change "a similar" to -an-.
Column 18, Line 27, change "to the" to -similar to the-.
Column 18, Line 62, change "what" to -that-.
Column 18, Line 63, change "list" to -lists-.
Column 20, Line 29, change "the component" to -component-.
Column 20, Line 35, change "ID a version" to -a CPE ID indicating a version-.
Column 21, Line 1, change "That" to -The-.
Column 22, Line 31, change "drop downs" to -dropdowns-.
Column 23, Line 41, change "to" to -to/from-.
Column 24, Line 17, change "some" to -a-.
Column 24, Line 18, change "some" to -an-.

Signed and Sealed this
First Day of September, 2020



Andrei Iancu
Director of the United States Patent and Trademark Office

Column 24, Line 20-21, change "the asset may be sensitive client information stored within the database" to -sensitive client information may be stored within the database-.
Column 27, Line 24, change "analyze" to -analyzing-.
Column 27, Line 26, change "analyze" to -analyzing-.
Column 27, Lines 27-28, change "measure and quantify" to -measuring and quantifying-.
Column 29, Line 11, change "threat" to -threat)-.
Column 29, Line 43, the number 1 should not be bolded.
Column 34, Line 7, change "functionality" to -the web page having functionality-.
Column 34, Line 57, change "is" to -includes-.
Column 37, Line 51, change "that" to -that is-.
Column 37, Line 52, change "similar elements" to -similar elements to the table of interface 2800-.
Column 42, Lines 51-52, change "SQL SERVER" to -SQL DATABASE-.

AO 120 (Rev. 08/10)

TO:	REPORT ON THE FILING OR DETERMINATION OF AN ACTION REGARDING A PATENT OR TRADEMARK
Mail Stop 8 Director of the U.S. Patent and Trademark Office P.O. Box 1450 Alexandria, VA 22313-1450	

In Compliance with 35 U.S.C. § 290 and/or 15 U.S.C. § 1116 you are hereby advised that a court action has been filed in the U.S. District Court for the District of Delaware on the following

☐ Trademarks or ☑ Patents. (☐ the patent action involves 35 U.S.C. § 292):

DOCKET NO.	DATE FILED	U.S. DISTRICT COURT
	7/8/2022	for the District of Delaware

PLAINTIFF	DEFENDANT
THREATMODELER SOFTWARE INC.	IRIUSRISK, INC.

	PATENT OR TRADEMARK NO.	DATE OF PATENT OR TRADEMARK	HOLDER OF PATENT OR TRADEMARK
1	US 10,699,008 B2	6/30/2020	ThreatModeler Software Inc.
2	US 10,713,366 B2	7/14/2020	ThreatModeler Software Inc.
3			
4			
5			

In the above—entitled case, the following patent(s)/ trademark(s) have been included:

DATE INCLUDED	INCLUDED BY ☐ Amendment ☐ Answer ☐ Cross Bill ☐ Other Pleading	
PATENT OR TRADEMARK NO.	DATE OF PATENT OR TRADEMARK	HOLDER OF PATENT OR TRADEMARK
1		
2		
3		
4		
5		

In the above—entitled case, the following decision has been rendered or judgement issued:

DECISION/JUDGEMENT

CLERK	(BY) DEPUTY CLERK	DATE

Copy 1—Upon initiation of action, mail this copy to Director Copy 3—Upon termination of action, mail this copy to Director
Copy 2—Upon filing document adding patent(s), mail this copy to Director Copy 4—Case file copy