

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SOPHOS LTD. and SOPHOS INC.,
Petitioners,

v.

WEBROOT INC.,
Patent Owner.

Case No. IPR2023-00528
U.S. Patent No. 10,599,844

**Declaration of Dr. Gene Tsudik in Support of
Petition for *Inter Partes* Review of U.S. Patent No. 10,599,844**

TABLE OF CONTENTS

PETITIONERS' EXHIBIT LIST	vi
TABLE OF ABBREVIATIONS	viii
I. Qualifications.....	2
II. Scope of Opinions.....	5
III. Materials Reviewed and Considered	5
IV. Level of Ordinary Skill in the Art	6
V. Statement of Legal Principles.....	7
A. Claim Construction.....	7
B. Obviousness.....	8
VI. Summary of the '844 Patent	9
A. Challenged Claims	12
B. Prosecution History of the '844 Patent	13
VII. State of the Art.....	20
A. Malware.....	20
B. Machine Learning / Support Vector Machines	21
C. Software Encryption and Packing (Compression)	22
D. Static and Dynamic Malware Analysis	23
VIII. Application of the Prior Art to the Claims	23
A. Effective Filing Date	23
B. Ground 1: Claims 1-20 are Obvious Over Ideses	24
1. Claim 1	25

2. Claim 2: “The computer-implemented method according to claim 1, wherein the extracting the plurality of static data points further comprises classifying the extracted static data points according to a type of data extracted from the executable file, and encoding the extracted static data points for the classifier based on the classification.” 42
3. Claim 3: “The computer-implemented method according to claim 2, wherein generating the at least one feature vector comprises selectively setting features of the classifier based on the plurality of extracted static data points.” 45
4. Claim 4: “The computer-implemented method according to claim 2, wherein the classifying the at least one static data point comprises classifying the plurality of extracted static data points into categories comprising numeric values, nominal values, string or byte sequences, and Boolean values.” 47
5. Claim 5: “The computer-implemented method according to claim 4, wherein analyzing the executable file further comprises generating at least one feature vector from the plurality of extracted static data points, wherein features of the generated feature vector are weighted based at least upon the classified categories and the plurality of extracted static data points.” 49
6. Claim 6: “The computer-implemented method according to claim 1, wherein determining whether the executable file is harmful further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value.” 51
7. Claim 7: “The computer-implemented method according to claim 6, wherein the threshold value is set based on predetermined false positive range data.” 51

8. Claim 8: “The computer-implemented method according to claim 1, wherein the plurality of static data points is extracted using a machine learning technique.” 53
9. Claim 9: “The computer-implemented method according to claim 1, wherein a determination of whether the executable file is harmful is used to retrain the classifier.” 53
10. Claim 10..... 54
11. Claim 11: “The computer-readable storage device according to claim 10, wherein the extracting of the plurality of static data points executed by the processor further comprises classifying the extracted static data points according to a type of data extracted from the executable file, and encoding the extracted static data points for the classifier based on the classifying.” 56
12. Claim 12: “The computer-readable storage device according to claim 11, wherein generating the at least one feature vector comprises selectively setting features of the classifier based on the plurality of extracted static data points.” 56
13. Claim 13: “The computer-readable storage device according to claim 11, wherein the classifying the at least one static data point comprises classifying the plurality of extracted static data points into categories comprising numeric values, nominal values, string or byte sequences, and Boolean values.” 57
14. Claim 14: “The computer-readable storage device according to claim 13, wherein analyzing the executable file further comprises generating a feature vector from the plurality of extracted static data points, wherein features of the generated feature vector are weighted based at least upon the classified categories and the plurality of extracted static data points.” 57
15. Claim 15: “The computer-readable storage device according to claim 10, wherein determining whether the

	executable file is harmful further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value.”	57
16.	Claim 16: “The computer-readable storage device according to claim 15, wherein the threshold value is set based on predetermined false positive range data.”	58
17.	Claim 17	58
18.	Claim 18: “The system according to claim 17, wherein the determining further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value, and wherein the threshold value is set based on predetermined false positive range data.”	61
19.	Claim 19: “The system according to claim 17, wherein the extracting the plurality of predefined character strings further comprises classifying the extracted plurality of predefined character strings according to a type of data extracted from the executable file, and encoding the extracted plurality of predefined character strings for the classifier based on the classification.”	61
20.	Claim 20: “The system according to claim 17, wherein determining whether the executable file is harmful further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value.”	61
C.	Ground 2: Claims 1, 8-10, and 17 are Obvious Over Mori	62
1.	Claim 1	63
2.	Claim 8: “The computer-implemented method according to claim 1, wherein the plurality of static data points is extracted using a machine learning technique.”	73

3.	Claim 9: “The computer-implemented method according to claim 1, wherein a determination of whether the executable file is harmful is used to retrain the classifier.”.....	73
4.	Claim 10.....	74
5.	Claim 17.....	75
D.	Ground 3: Claims 1-20 are Obvious Over Ideses in View of Mori.....	77
IX.	Secondary Considerations of Nonobviousness	81
X.	Conclusion.....	83

PETITIONERS' EXHIBIT LIST

Exhibit No.	Document
1001	U.S. Patent No. 10,599,844 (“the ’844 Patent”)
1002	File History for U.S. Patent No. 10,599,844
1003	Defendant Sophos Ltd.’s Stipulation Regarding Invalidity Contentions for U.S. Patent No. 10,599,844
1004	Declaration of Dr. Gene Tsudik
1005	U.S. Patent Appl. Pub. No. 2015/0213376 (“Ideses”)
1006	Japanese Patent Appl. Pub. No. 2012027710A (“Mori”) and Certified Translation Thereof
1007	Defendants’ Opening Claim Construction Brief, D.I. 86, <i>Webroot Inc. et al. v. AO Kaspersky Lab et al.</i> , No. 6:22-cv-00243 (W.D. Tex. Oct. 28, 2022)
1008	Order Granting Third Amended Scheduling Order, D.I. 142, <i>Webroot Inc. et al. v. AO Kaspersky Lab et al.</i> , No. 6:22-cv-00243 (W.D. Tex. Dec. 27, 2022)
1009	Wang <i>et al.</i> , <i>Detecting Unknown Malicious Executables Using Portable Executable Headers</i> , Fifth International Joint Conference on INC, IMS and IDC, 278-84 (2009)
1010	Michael Sikorski & Andrew Honig, <i>Practical Malware Analysis</i> (2012)
1011	Murugiah Souppaya & Karen Scarfone, <i>Guide to Malware Incident Prevention and Handling for Desktops and Laptops</i> , NIST Spec. Pub. 800-83 (Rev. 1) (July 2013)
1012	Monnappa K A, <i>Learning Malware Analysis</i> (2018)
1013	V. Kecman, <i>Support Vector Machines – An Introduction</i> , StudFuzz 177, 1–47 (2005)
1014	<i>Virus Bulletin</i> (Nov. 1990)
1015	U.S. Patent No. 8,709,924
1016	U.S. Patent No. 9,465,940
1017	D. Devi & S. Nandi, <i>Detection of Packed Malware</i> , in Proceedings of the First International Conference on Security of

Exhibit No.	Document
	Internet of Things (SecurIT '12), Association for Computing Machinery, New York, NY, USA, 22–26 (2012)
1018	Excerpt from Transcript of Motions Hearing, <i>Webroot Inc. et al. v. AO Kaspersky Lab et al.</i> , No. 6:22-cv-00243 (W.D. Tex. Dec. 9, 2022)

TABLE OF ABBREVIATIONS

Abbreviation	Term
'844 Patent	U.S. Patent No. 10,599,844
AIA	America Invents Act
Challenged Claims	Claims 1–20 of U.S. Patent No. 10,599,844
Ex.	Exhibit
Fig.	Figure
Ideses	U.S. Patent Appl. Pub. No. 2015/0213376 (Ex. 1005)
IPR	<i>inter partes</i> review
Mori	Japanese Patent Appl. Pub. No. 2012027710A (Ex. 1006)
Patent Owner	Webroot, Inc.
Petitioners	Sophos Ltd. and Sophos Inc.

I, Dr. Gene Tsudik, declare as follows:

1. I have been retained by counsel for Sophos Ltd. and Sophos Inc. (“Petitioners”) to submit a declaration in support of the Petition for *Inter Partes* Review of U.S. Patent No. 10,599,844 (“Petition”), which I understand has been assigned to Webroot Inc. (“Webroot” or “Patent Owner”).

2. I have been asked to opine on whether the ’844 Patent is rendered obvious by the prior art. My opinions are based on my experience, knowledge, and skill in computer/network security and applied cryptography. In preparation for this declaration, I have reviewed the ’844 Patent, its prosecution history, and all exhibits cited in this declaration and in the Petition.

3. I may rely upon these materials, my experience and knowledge, and/or additional materials to rebut arguments raised by the Patent Owner. Further, I may also consider additional documents and information in forming any necessary opinions, including documents that may not yet have been provided to me.

4. My analysis of the materials produced in this investigation is ongoing and I will continue to review any new material as it is provided. This declaration represents only those opinions I have formed to date. I reserve the right to revise, supplement, and/or amend my opinions stated herein based on new information and on my continuing analysis of the materials already provided.

5. I am being paid at my normal hourly rate for my work on this matter. I have no personal or financial stake or interest in the outcome of the Petition.

I. Qualifications

6. I am a Distinguished Professor of Computer Science and Vice-Chair for Graduate Studies in the Computer Science Department at the University of California, Irvine (“UCI”). I earned a Bachelor of Science degree in Computer Science from the University of Houston – University Park in 1985. I also earned Master of Science and Ph.D. degrees from the University of Southern California in 1987 and 1991, respectively.

7. Throughout my research career (started in 1987), I conducted research mostly in two key areas: (1) computer and network security/privacy, as well as (2) applied cryptography. Nowadays, these areas are typically covered by a single term: **Cybersecurity**, which is used hereafter. Cybersecurity is not only a very important area within Computer Science (CS); it is cross-cutting in nature, i.e., it permeates most other CS areas and beyond. In terms of external recognition, I am a Fellow of IEEE, ACM, IFIP and AAAS – the four major scientific societies that award a status of Fellow based on rigorous analysis of one’s research results. I am also a *foreign member* of The Academy of Europe (Academia Europaea). I have been a Fulbright Scholar and Fulbright Specialist (thrice). Here are some highlights of my research history:

- In 1987-1989, my dissertation research first explored Internet firewalls and virtual private networks: the need for them was not recognized until at least 6-7 years later.
- In 1991-1992, I designed a very simple tamper-resistant token smartcard architecture used for personal authentication: the need was not realized until 5-6 years later.
- In 1994-1995, I identified major security flaws in mobile and cellular networks; recognition of these flaws and design goals did not become apparent until 8-10 years later.
- In 1999-2000, I pioneered anonymous communication techniques, which laid the groundwork for **Tor**, the preeminent means of anonymous Internet communication since early 2000-s.
- In 1996-2000, I proposed several techniques for privacy-preserving authentication, aka “group signatures”; 8-10 years later Intel incorporated a version of one of my techniques into its TPM chips to support so-called “direct anonymous attestation.” It is still present in 100-s of millions of devices.
- In 2003-2004, I developed several seminal techniques for secure cloud storage; 5-6 years later cloud storage became reality and my work served as the foundation for deployed solutions.

- In 2010-2013, I came up with a suite of secure and practical techniques for several privacy-preserving genomic tests. This research anticipated the advent of personal genomics and applications (e.g., personalized medicine) that it would engender. This dream became today's reality, owing to increasing affordability and ubiquity of whole genome sequencing, as well as a range of emerging threats.
- From 2011 on, I produced numerous advances in securing of (and identifying flaws in) Content-Centric Networking (CCN) – a network architecture that is likely to become the model for the next-generation Internet. The benefits of this work are already being reaped with on-going CCN trial deployment efforts.
- Also, though imperfect, area-specific and subjective, well-known H-Index is often used as a measure of a researcher's overall impact. Based on Google Scholar (01/01/23), I have nearly 40,000 citations and an H-Index of 102.

8. Additional information concerning my professional experience is set forth in my current Curriculum Vitae, a copy of which is attached as Appendix A. I am not currently retained as an expert in any other litigation matters. In the past, I have been retained as an expert in several litigation matters, e.g.:

- Safeclick v. Visa (2004-2005)

- SellerBid v. Groupon (2011)
- Computer Software Protection LLC v. Adobe Systems Inc (2013-2014)
- Safeclick v. Visa (2004-2005)
- ContentGuard v. Amazon (2015)

9. If asked, I will testify regarding my qualifications, background, and experience in the field of Computer Science, and specifically regarding computer security, privacy, and applied cryptography.

II. Scope of Opinions

10. I have been asked to provide my opinions regarding whether the Challenged Claims of the '844 patent would have been obvious to one of ordinary skill in the art at the time of the effective filing date in view of Ideses (Ex. 1005) (Ground 1) (all challenged claims); Mori (Ex. 1006) (Ground 2) (claims 1, 8-10, and 17); and Ideses and Mori in combination (Ground 3) (all challenged claims).

11. This declaration, including the exhibits cited herein, sets forth my opinions on these topics. It is my opinion that the Challenged Claims of the '844 patent would have been obvious based on the grounds listed above.

III. Materials Reviewed and Considered

12. The materials I considered and relied upon in preparing my declaration and forming my opinions include all exhibits cited in this declaration and the

Petition, including the '844 patent, the prosecution history of the '844 patent, and all of the relevant prior art.

13. I also have relied on my professional experience, knowledge, and skill in computer security, privacy, and applied cryptography in developing the opinions expressed in this declaration.

IV. Level of Ordinary Skill in the Art

14. In providing the opinions set forth in this declaration, I was asked to consider the patent claims and the prior art through the eyes of one of ordinary skill in the art at the effective filing date of the claimed invention of the '844 patent.

15. I understand that certain factors may help establish the level of skill in the art, such as: (1) the education level of those working in the field; (2) the sophistication of the technology; (3) the types of problems encountered in the art; (4) prior art solutions to those problems; and (5) the speed at which innovations are made.

16. Taking into account the aforementioned factors and based upon my years of experience in Computer Science, it is my opinion that one of ordinary skill in the art would have a Bachelor's degree in Computer Science, Computer Engineering, Electrical Engineering, or an equivalent discipline. In addition, it is my opinion that one of ordinary skill in the art would have had at least one year's worth of experience relating to static data analysis and malware detection at the time of the

effective filing date of the claimed invention, May 12, 2015. In the alternative, a person of skill in the art could have had two or more years of work experience relating to static data analysis and malware detection in lieu of the education requirements.

V. Statement of Legal Principles

A. Claim Construction

17. I understand that terms appearing in the patent claims are to be interpreted according to their “ordinary and customary meaning” in an IPR proceeding. In determining the ordinary and customary meaning, the words of a claim are first given their plain meaning as they would have been understood by one of skill in the art at the time of the effective filing date of the claimed invention, in light of the specification and prosecution history. I understand that, under limited circumstances, evidence external to the patent and its prosecution may be used to determine the meaning attributed by one of skill in the art at the time of the effective filing date of the claimed invention. I have followed this approach in my analysis, and have applied the ordinary and customary meaning of all terms of the Challenged Claims throughout my analysis in this declaration.

18. I understand that the words of the claims should be interpreted as they would have been understood by a person of ordinary skill in the art at the time of the

effective filing date of the claimed invention. For purposes of my analysis, I have applied a date of May 12, 2015, as the effective filing date.

B. Obviousness

19. I have been informed that a patent claim is invalid as “obvious” under AIA 35 U.S.C. § 103 if the differences between the claimed invention and the prior art are such that the claimed invention as a whole would have been obvious before the effective filing date of the claimed invention to a person having ordinary skill in the art to which the claimed invention pertains. I have reviewed AIA 35 U.S.C. § 103.

20. I have also been informed that the framework for determining obviousness involves considering the following factors: (1) the scope and content of the prior art; (2) the differences between the prior art and the claimed subject matter; (3) the level of ordinary skill in the art; and (4) any objective evidence of non-obviousness as set forth in the case of *Graham v. John Deere Company*. I have been informed that in the case of *KSR International Company v. Teleflex*, the Supreme Court reaffirmed the framework for determining obviousness that was previously set forth in the case of *Graham v. John Deere Company*.

21. I have further been informed that the “objective” (or “secondary”) evidence of non-obviousness includes things such as: (1) a long-felt but unmet need in the prior art that was satisfied by the claimed invention; (2) commercial success

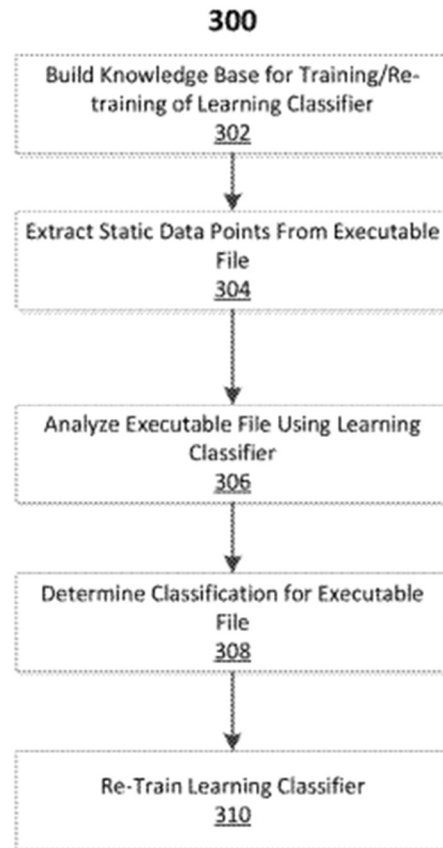
or the lack of commercial success of the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise of the claimed invention by others skilled in the art; (5) taking of licenses under the patent by others; (6) deliberate copying of the claimed invention; and (7) contemporaneous and independent invention by others.

VI. Summary of the '844 Patent

22. The '844 Patent is entitled “Automatic Threat Detection of Executable Files Based on Static Data Analysis.” Ex. 1001, Cover. I understand that the '844 patent was filed on May 12, 2015 as application number 14/709,875. *Id.* For purposes of my analysis, I have assumed that the effective filing date of the '844 Patent is the application filing date, May 12, 2015.

23. The '844 Patent relates generally to “threat detection of executable files.” Ex. 1001, Abstract. More specifically, the '844 Patent describes and claims methods and systems wherein “[a] plurality of static data points are extracted from an executable file without decrypting or unpacking the executable file.” *Id.* at 1:21-23. Purportedly, the executable file “may then be analyzed without decrypting or unpacking the executable file.” *Id.* at 1:23-25.

24. Figure 3 of the '844 Patent, reproduced below, “illustrates an exemplary method 300 for implementation of threat detection systems and methods” described in the patent:

FIG. 3

Id. at 1:54-56, 11:41-52, Fig. 3.

25. At step 302, “a knowledge base is built for training/retraining of a learning classifier used to detect threats in executable files.” *Id.* at 11:53-55. The knowledge base is built “from data collected and evaluated related to known malicious executable files, known benign executable files and potentially unwanted executable files.” *Id.* at 11:55-60.

26. At step 304, “one or more static data points are extracted from an executable file” that has been “identified for evaluation.” *Id.* at 12:4-6. Specifically,

“an executable file is analyzed using machine learning processing . . . to collect/extract static data points from an executable file for evaluation.” *Id.* at 12:6-10. The ’844 Patent notes that “[i]n examples, operation 304 occurs without decrypting or unpacking the executable file.” *Id.* at 12:10-12. The ’844 Patent also notes that “the actions performed at operation 304 may be used to classify the plurality of static data points extracted into categories (categorical values) comprising numeric values, nominal values, string or byte sequences, and Boolean values.” *Id.* at 12:26-30.

27. At step 306, “the executable file is analyzed for threats.” *Id.* at 12:36-37. The ’844 Patent explains that step 306 “comprises applying a learning classifier (e.g., the learning classifier generated during performance of operation 302) to the plurality of static data points extracted from the file,” and also “comprises generating at least one feature vector.” *Id.* at 12:41-44, 12:48-51. The ’844 Patent further explains that “generation of the feature vector may comprise selectively setting features of the learning classifier based on the one or more of static data points extracted,” and that features “may be weighted based on classified categories.” *Id.* at 12:54-60. As a specific example, “one or more features of the feature vector may be selectively turned on or off based on evaluation of whether a value of a static data point is within a predetermined range.” *Id.* at 12:60-63.

28. At step 308, “a determination is made as to a classification of the executable file,” such as “whether the executable file is harmful (e.g., malicious, malware) or not based on the results of the analysis of the executable file (e.g., using machine learning processing by a learning classifier).” *Id.* at 13:5-11. The ’844 Patent notes that the an example of a “learning classifier” is an “SVM,” i.e., a Support Vector Machine. *Id.* at 13:11-14; *see id.* at 9:41-67.

29. The operation at step 310, wherein “a learning classifier used for threat detection processing is re-trained,” may occur “[a]t any point in time.” *Id.* at 13:34-63. The ’844 Patent explains that “[c]ontinuously re-training of the learning classifier may ensure that the threat detection application/service is up to date and able to accurately detect new threats.” *Id.* at 13:36-38.

30. As discussed in more detail below, all of these basic concepts of static malware analysis, feature extraction and classification, and machine learning were well known and used prior to the effective filing date of the ’844 Patent.

A. Challenged Claims

31. The ’844 Patent has three independent claims, and 17 dependent claims. Claim 1 is representative of the independent claims, and is provided below:

[1.pre] A computer-implemented method comprising:

[1.a] extracting a plurality of static data points from an executable file without decrypting or unpacking the executable

file, wherein the plurality of static data points represent predefined character strings in the executable file;

[1.b] generating a feature vector from the plurality of static data points using a classifier trained to classify the plurality of static data points based on a collection of data comprising known malicious executable files, known benign executable files, and known unwanted executable files, wherein the collection of data comprises at least a portion of the plurality of static data points, and wherein one or more features of the feature vector are selectively turned on or off based on whether a value of one or more static data points from the plurality of extracted static data points is within a predetermined range; and

[1.c] evaluating the feature vector using support vector processing to determine whether the executable file is harmful.

B. Prosecution History of the '844 Patent

32. U.S. application no. 14/709,875 was filed on May 12, 2015, and issued as the '844 Patent on March 24, 2020. *See* Ex. 1001, Cover. The application for the '844 Patent was filed with 20 original claims. *See* Ex. 1002 at 49-52.

33. The independent claims were rejected several times during examination, and multiple amendments to the independent claims were made in response. The examiner first rejected the independent claims as anticipated by the Wojnowicz reference, U.S. Patent No. 9,465,940. *Id.* at 92-97. I note that in this (and

subsequent) patent office actions, the examiner incorrectly referred to “U.S. Patent No. 8,709,924,” which does not list Wojnowicz as an inventor (and describes subject matter unrelated to the ’844 patent), but the applicant’s responses use the correct patent number, which does list Wojnowicz as an inventor: U.S. Patent No. 9,465,940. *See, e.g., id.* at 94, 133; Ex. 1015, Cover; Ex. 1016, Cover.

34. The Wojnowicz reference describes “systems, methods and computer program products that address the issue of detecting malicious commands within portable executable files that are encrypted or compressed.” Ex. 1016 at 2:37-40. Wojnowicz “provides for automatic quantification of the extent to which the patterned variations in a file’s entropy time series makes it suspicious.” *Id.* at 2:45-48. Wojnowicz explains that its “subject matter is particularly useful for malware files that contain encrypted or compressed (‘packed’) segments which conceal malicious commands.” *Id.* at 3:1-3. In the rejection, the examiner noted that in Wojnowicz, “it is apparent that static data points are used without decrypting or unpacking.” Ex. 1002, 94.

35. In response to the rejection, the applicant amended the independent claims to add “wherein the static data points correspond to one or more points of reference used to evaluate the executable file” to the “extracting a plurality of static data points” step:

1. (Currently Amended) A computer-implemented method comprising:
extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the static data points correspond to one or more points of reference used to evaluate the executable file;
analyzing the executable file without decrypting or unpacking the executable file, wherein analyzing the executable file comprises applying a classifier to the plurality of static data points, wherein the classifier is trained from data that comprises known malicious executable files, known benign executable files and known unwanted executable files; and
determining whether the executable file is harmful based on the analysis.

Id. at 127-32. The applicant explained that, in their view, Wojnowicz did not teach “file chunks corresponding to points of reference in a file or organizing file chunks . . . into categories to evaluate a file.” *Id.* at 133-36.

36. The examiner disagreed with the applicant’s arguments, and rejected the claims again as anticipated by Wojnowicz. In the examiner’s view, Wojnowicz described “one or more points of reference” through the use of “string information” at a static location within the executable file, and evaluating an executable file through its disclosure of the calculation of “a suspiciously structured entropy score” in a malware classifier. *Id.* at 147-50. In response, the applicant amended the independent claims and added “wherein the one or more points of reference are defined by the executable file” to the “extracting a plurality of data points” step:

1. (Currently Amended) A computer-implemented method comprising:
extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the static data points correspond to one or more points of reference used to evaluate the executable file, and wherein the one or more points of reference are defined by the executable file;
analyzing the executable file without decrypting or unpacking the executable file, wherein analyzing the executable file comprises applying a classifier to the plurality of static data points, wherein the classifier is trained from data that comprises known malicious executable files, known benign executable files and known unwanted executable files; and
determining whether the executable file is harmful based on the analysis.

Id., 178-83. The applicant argued that although points of reference “may exist in the executable files processed in Wojnowicz,” that Wojnowicz did not teach using points of reference “corresponding to the format/layout of an executable file.” *Id.* at 184-87.

37. The examiner again rejected the independent claims as anticipated by Wojnowicz. The examiner determined that Wojnowicz’s description of splitting a file into 256-byte sections was the same as defining points of reference located “at each consecutive 256 byte location.” *Id.* at 211. In response, the applicant amended the independent claims to replace portions of the previously amended language to recite that the “one or more of the plurality of static data points” are “points of reference representing sections of the executable file”:

1. (Currently Amended) A computer-implemented method comprising:
 - extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein one or more of the plurality of static data points correspond to are one or more points of reference representing sections of the executable file, used to evaluate the executable file, and wherein the one or more points of reference are used to evaluate defined by the executable file;
 - analyzing the executable file without decrypting or unpacking the executable file, wherein analyzing the executable file comprises applying a classifier to the plurality of static data points, wherein the classifier is trained from data that comprises known malicious executable files, known benign executable files and known unwanted executable files; and
 - determining whether the executable file is harmful based on the analysis.

Id. at 241-48. The applicant explained that Wojnowicz’s teaching of fixed-length file chunk boundaries “do not correspond to particular sections of the file,” such as “a DOS header section, a fields section, and a data directories section” and that Wojnowicz’s file chunks “may comprise data from different sections of the file.” *Id.* at 249.

38. The examiner disagreed with the applicant’s arguments, and rejected the independent claims as anticipated by Wojnowicz once more. *Id.* at 265-69. The examiner explained that the independent claims (as amended) did not require that the sections of the claimed “‘points of reference’ are split at particular sections” that perform a specific function (or have a specific purpose), therefore, Wojnowicz taught the limitations of the independent claims. *See id.* at 267-68. In response, the applicant amended the independent claims and removed the “points of reference” limitations, added “wherein the plurality of static data points represent predefined

character strings in the executable file” to the “extracting” step, and modified the “analyzing” step to recite “analyzing the plurality of static data points using a classifier . . . trained to classify the plurality of static data points based on a collection of data comprising . . . wherein the collection of data comprises the plurality of static data points”:

1. (Currently Amended) A computer-implemented method comprising:

extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the plurality of static data points represent predefined character strings in the executable file ~~one or more of the plurality of static data points are points of reference representing sections of the executable file, wherein the points of reference are used to evaluate the executable file;~~

analyzing the plurality of static data points using a classifier ~~executable file without decrypting or unpacking the executable file, wherein analyzing the executable file comprises applying a classifier to the plurality of static data points, wherein the classifier is trained to~~ classify the plurality of static data points based on a collection of data comprising ~~from data that comprises~~ known malicious executable files, known benign executable files and known unwanted executable files, wherein the collection of data comprises the plurality of static data points; and

determining whether the executable file is harmful based on the analysis.

Id. at 304-09. The applicant explained that Wojnowicz’s “fixed-length file chunks” do not represent “predefined character strings” as recited in the amended claims, which was discussed in an interview with the examiner. *Id.* at 312.

39. The examiner subsequently rejected the independent claims as anticipated by a different reference entitled “Detection of Packed Malware” by Devi *et al.* in an office action dated July 17, 2019. *Id.* at 320-23. The Devi reference is

directed to applying “a pattern recognition technique for detection of packed malware binaries” with an objective to “take out the best set of features from Windows Portable executable files in order to pass it to our classification model.” Ex. 1017 at 1. Devi’s classification model “works in two phases, in the first phase it classifies the packed and non-packed executables,” and “[o]nce an executable is classified as packed, the second phase of classification concludes whether it is packed benign or packed malware executable.” *Id.* The examiner used a combination of Devi with other references to reject some of the dependent claims, including “Efficient Malicious Code Detection using N-Gram Analysis and SVM” by Choi et al. (claims 3 and 12), Choi with U.S. Patent Application Publication 2009/0013405 to Schipka (claims 4 and 13), U.S. Patent Application Publication 2012/0079596 to Thomas et al. (claims 6 and 15), Thomas with U.S. Patent 8,510,836 to Nachenberg (claims 7 and 16), and 9,197,663 to Gilbert (claims 8-9, 17-18, and 20). *See* Ex. 1002 at 323-27.

40. In the July 17, 2019 office action, the examiner explained, however, that pending dependent claims 5 and 14, which recited “wherein features of the feature vector are selectively turned on or off based on evaluation of whether a value of a static data point from the plurality of extracted static data points is within a predetermined range,” would be allowable if rewritten in independent form. *Id.* at 305, 308, 327.

41. Ultimately, once the allegedly allowable subject matter of claims 5 and 14 (and intervening claims) was incorporated into the independent claims, the examiner allowed the claims, and the '844 patent issued on March 24, 2020. *See id.* at 425-42, 456.

VII. State of the Art

A. Malware

42. Malware, or malicious software, refers to computer programs developed to modify, steal, or destroy data on a computer or a computer network. Furthermore, some malware surreptitiously uses computer and/or network resources for the benefit of the author/distributor, e.g., crypto-mining malware. Another type of malware encrypts data and holds it for ransom, i.e., so-called ransomware. Yet another type of malware uses the infected computer as a “zombie” (of which there might be a great multitude) to mount so-called DDoS (Distribute Denial-of-Service) attacks against some target computer(s).

43. Malware varies based on the means of its propagation. Viruses and worms self-replicate, meaning that they constantly look for opportunities to infect more and more computers. They can do so via the network or via portable/removable storage devices (e.g., USB sticks or microSD cards). The chief difference between viruses and worms is that the latter generally does not exfiltrate, corrupt, or erase data; instead, a worm generally consumes computer and network resources, e.g.,

network bandwidth, CPU cycles, as well as storage and memory space. Meanwhile, a virus almost always aims to exfiltrate, corrupt, or erase data (or any combination thereof). They continuously self-replicate and spread to other computers on a network. Whereas, a trojan horse is a program designed to appear as a legitimate computer program (e.g., a software update) that can be used surreptitiously to gain access to a victim computer. For example, a trojan horse may pose as a Windows update from Microsoft. However, once installed on a computer, it may damage, disrupt, steal, or inflict some other harmful action upon local data or a computer network. It will be noted that there is no commonly accepted precise distinction between viruses, worms, and trojan horses.

44. Generally, malware is introduced into a computer or computer network via the Internet or via an external storage device. It may be hidden within bundles of software, concealed in the firmware of an external storage device, or disguised as a legitimate (benign) program.

B. Machine Learning / Support Vector Machines

45. Machine learning (“ML”) is a sub-field of artificial intelligence (“AI”) that is focused on techniques that “learn” from (usually large amounts of) data to improve performance of certain tasks. Within ML, Support Vector Machines (“SVMs”) are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. Supervised learning is

defined by its use of labeled datasets to train algorithms that classify data or predict outcomes accurately. Supervised learning uses a training dataset to teach models to yield the desired output. The training dataset includes inputs and correct outputs, which allow the model to learn over time. The algorithm measures its accuracy through a loss (or cost) function until the error has been sufficiently minimized.

C. Software Encryption and Packing (Compression)

46. Since software (initial installation, updates, or patches) can be very large in size, it is often compressed to minimize its transportation size. Packing (or compression) aims to minimize the size of software by intricate packing of redundant components.

47. Software may be encrypted in order to make it more difficult for a third party to reverse engineer, decompile, or disassemble the executable. Encrypted software must be decrypted before the underlying object code can be analyzed or executed. Further, whenever software is a paid and/or restricted commodity, it is usually encrypted *after* compression such that only its duly authorized recipients can decrypt, then un-compress and install it.

48. Malware authors have utilized encryption techniques for at least the past three decades in order to prevent malware code from being subject to static analysis, which typically requires disassembly of the code. *See* Ex. 1014 at 13. For example, encrypted viruses typically consist of two parts: (1) a decryption program;

and (2) the encrypted main body of the virus code. When an infected executable is run, the decryption program decrypts the actual virus code, which is then executed.

D. Static and Dynamic Malware Analysis

49. Static analysis is the analysis (usually automated, by another program) of a computer program performed without executing it. This contrasts with dynamic analysis, which is performed upon a program during its (usually sandboxed, meaning *carefully isolated and monitored*) execution.

50. One basic static malware analysis technique is to search through a strings contained in a target executable file, which are typically stored in ASCII or Unicode format. *See, e.g.*, Ex. 1010 at 8-10. Although legitimate programs usually include many strings, malware files that include packed or encrypted portions may contain fewer strings. *See id.* at 10. However, strings may still be extracted from PE headers, file metadata, and/or stub decryption/unpacking code even where the main body of the executable file is encrypted/compressed. *See id.* at 10, 12.

VIII. Application of the Prior Art to the Claims

A. Effective Filing Date

51. As noted above, I have assumed that the effective filing date of the '844 Patent is May 12, 2015, which is the date the application was filed.

B. Ground 1: Claims 1-20 are Obvious Over Ideses

52. I have analyzed claims 1-20 of the '844 patent, and it is my opinion that each of these claims is obvious in light of Ideses.

53. Ideses (Ex. 1005) is a United States patent application publication that was filed on January 29, 2015 and published on July 30, 2015. Ex. 1005, Cover Page. Ideses is entitled “Methods and Systems for Generating Classifiers for Software Applications,” and the applicant is listed as Shine Security Ltd. *Id.* I understand that Ideses is available as prior art because it was filed prior to the effective filing date of the '844 patent (May 12, 2015).

54. At a high level, Ideses discloses a method for providing “run-time classification of software applications, . . . to detect if the software application for installation [on a mobile device] is malware, adware, or benign (e.g., fine for installation).” *Id.* ¶ 123.

55. Ideses generally describes the performance of feature extraction during the execution of a software application (that is, at run time). However, feature extraction at run time is an *optional* feature, and Ideses also discloses static analysis through the “off-line” extraction of features that “may contain static meta data regarding the application, for example, icon, name, rating or other features,” as well as “information regarding the executable code and/or software package, for example,

in the form of byte code, resources, permissions, or other features.” *Id.* ¶¶ 74, 89, 133.

56. Ideses explains that “the extracted features are collated into feature vectors,” and that “the software application is classified . . . by a run-time classification module,” which can label the software application, for example, “as benign or adware.” *Id.* ¶¶ 134-35.

57. As shown in the element-by-element analysis of the Challenged Claims below, Ideses discloses (or renders obvious) every claimed feature of the “threat detection” method and system of the ’844 patent.

58. Based on my review, it is my opinion that a POSITA reading Ideses would have been able to implement Ideses’ method for training a classifier for classifying software applications with little or no experimentation required. Ideses provides sufficient detail regarding its classifier training method such that a POSITA would have been able to implement it without undue experimentation.

1. Claim 1

59. It is my opinion that Ideses renders claim 1 obvious, for the reasons discussed below.

a. [1.pre]: “A computer-implemented method comprising:”

60. I understand that a preamble typically does not limit a claim. However, to the extent that the preamble of claim 1 is limiting, it is my opinion that it is disclosed by Ideses. I interpret “computer-implemented” as meaning “automated” using a computer program.

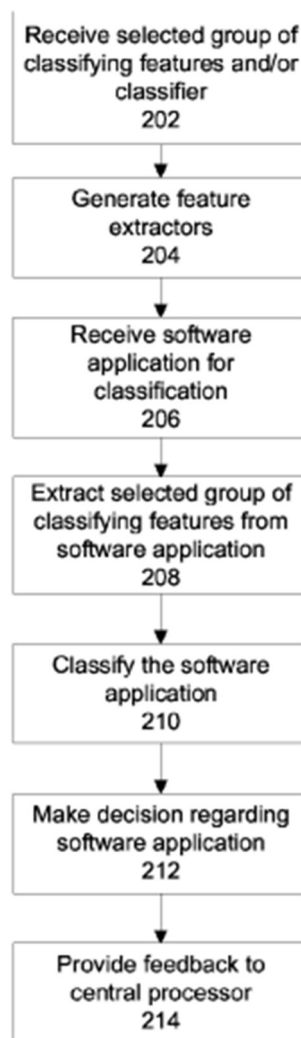
61. Ideses is directed to “methods and systems for generating classifiers for software applications.” Ex. 1005, ¶ 2. From this statement alone, one of skill in the art would have understood that Ideses discloses a “computer-implemented method” as required by the preamble of claim 1.

62. More specifically, Ideses states that the invention “may be a system, a method, and/or a computer program product.” *Id.* ¶ 63. Ideses also explains that the “computer program product may include a computer readable storage medium (or media) having computer readable program instructions thereon for causing a processor to carry out aspects of the present invention.” *Id.* Ideses describes assorted hardware that could be used as a “computer readable storage medium,” and provides additional details about the “computer readable program instructions,” that “carry[] out the operations of the present invention.” *See id.* ¶¶ 64-69.

- b. [1.a]: “extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the plurality of static data points represent predefined character strings in the executable file;”**

63. It is my opinion that Ideses discloses each part of this limitation.

64. Limitation [1.a] first requires “extracting a plurality of static data points from an executable file.” This “extracting” step is shown in Ideses’ Figure 2, which “is a flowchart of a method of run-time execution of a classifier based on a pruned feature list.” Ex. 1005 ¶ 123.



Id., Fig. 2.

65. Step 206 of Figure 2 discloses that “a software application,” which one of ordinary skill in the art would have understood to be an “executable file,” is received at a mobile device, such as having been downloaded from the internet. *Id.* ¶ 131. Step 208 discloses that “features are extracted” either “based on the software application,” “based on the group of classifying features and/or trained classifier received from the central server,” or “by the generated feature extractors.” *Id.* ¶ 132.

66. The '844 patent indicates that “static data points” are “point[s] of reference used to evaluate an executable file,” that include, but are not limited to: “header information, section information, import and export information, certificate information, resource information, string and flag information, legal information, comments, and/or program information (e.g., APIs and DLLs), among other examples.” Ex. 1001 at 5:62-6:1. Table 1.1 of the '844 Patent, reproduced below, “identifies some of the static data points identified for analysis of an executable file, where the static data points are organized by category”:

TABLE 1.1

Numeric values	Nominal values	Strings/Byte sequences	Boolean values
File size	initialize	Comments	Address Of Entry Point Anomaly
linker version	Un-initialize	company name	Image Base Anomaly
code size	entry point	file description	Section Alignment Anomaly
OS version	subsystem	internal name	Size Of Code Mismatch Anomaly
image version	file subtype	legal copyright	Low Import Count Anomaly
subsystem	language	original file	Entry Point Anomaly
version			
file version	file flags masks	private build	certificate Validity
number			
product version	file flags	product name	Certificate Exception
number			
size of heap	file OS	special build	Code Characteristics Anomaly
size of stack	file type	product version	Code Name Anomaly
size of image	machine type	file version	Count Anomaly
PE header time	PE type	package code	Data Characteristics Anomaly
Section Entropy	section counts	product code	Data Name Anomaly
Sections count	DLL count	export DLL name	Export Exception
	DLL functions	assembly version	Large Number of DLLs Anomaly
	data directory	Certificate Issuer	flag DLL Name Anomaly
	export count	Certificate Subject	Number of Functions Anomaly
		Imports	Function Name Anomaly
	Earliest Data Byte	Exports	PE Header Anomaly
	resources	Section Names	High Section Count Anomaly
	resources language	Non-resource	PE Magic Validity
	resource Encoding	section strings	
	resource code		Resource Exception
	page		
	resource size		VR Code Ratio Anomaly
	DLL		Import Exception
	characteristics		

Id. at 6:1-6, 6:19-22, Table 1.1.

67. Ideses’ description of the “extracted features” is similarly broad, and “may be *any feature* that describes the software application.” Ex. 1005 ¶ 89 (emphasis added). For example, “the extracted features may contain *static meta data* regarding the application, for example, icon, name, rating or other features,” and may further contain “information regarding the executable code and/or software package, for example, in the form of byte code, resources, permissions, or other

features.” *Id.* (emphasis added). In fact, the examples of “static meta data” listed in Ideses correspond to one (or more) of the examples of “static data points” in the ’844 Patent, as shown in the table below:

Ideses (Ex. 1005)	’844 Patent (Ex. 1011)
“icon”	“resource information”
“name”	“string . . . information,” “company name,” “internal name,” “product name”
“information regarding the executable code and/or software package . . . [e.g.,] byte code, resources”	“resource information,” “program information (e.g., APIs and DLLs),” “package code”
“permissions”	“file flags”

One of ordinary skill in the art would have understood these “extracted features” that contain “static meta data” to comprise “a plurality of static data points,” as required by limitation [1.a].

68. Limitation [1.a] also requires that the “extracting” is performed “without decrypting or unpacking the executable file.” As I explained above, Ideses primarily describes the performance of feature extraction during the execution of a software application (that is, at run time). Ideses explains that “behavioral features, for example, temporal information regarding system calls, system usage, CPU usage, [and] network utilization.” *Id.* ¶ 89. Such run-time feature extraction of “behavioral

features” would necessarily require that the executable file first be decrypted and/or unpacked prior to execution.

69. However, Ideses explains that run-time extraction is an *optional* feature, and also discloses the use of “off-line” *static analysis* through the extraction of features that “may contain *static meta data regarding the application*, for example, icon, name, rating or other features,” as well as “information regarding the executable code *and/or software package*, for example, in the form of byte code, resources, permissions, or other features.” *Id.* ¶¶ 74, 89-90, 133 (emphasis added).

70. In my opinion, Ideses’ disclosure of the off-line extraction of “static meta data regarding the application” would have been understood by one of skill in the art to mean that these features are extracted “without decrypting or unpacking the executable file,” because decryption/unpacking of the executable file would typically happen at run-time via bootstrap decryption/decompression code. *See* Section VII.C above. As I explained above, off-line static malware analysis techniques were well-known to one of skill in the art, and would have been used by a malware analyst as a first step performed prior to unpacking or decrypting an executable. *See* Section VII.D above.

71. Limitation [1.a] further requires that “the plurality of static data points represent predefined character strings in the executable file.” As I noted above, Ideses’ “extracted features” “may be *any feature* that describes the software

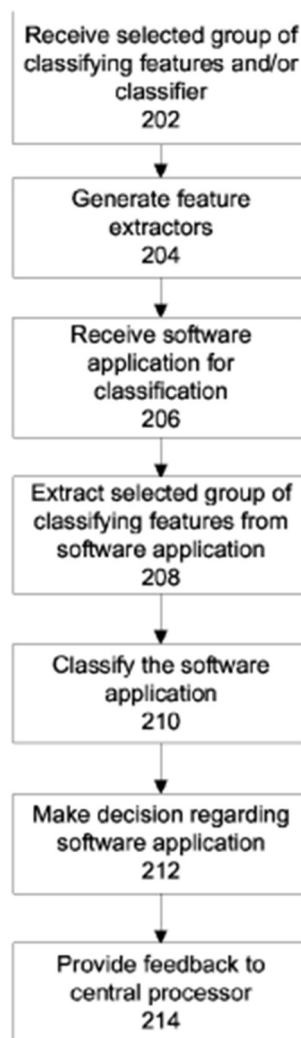
application,” such as “static meta data regarding the application, for example, icon, name, rating or other features,” as well as “information regarding the executable code and/or software package, for example, in the form of byte code, resources, permissions, or other features.” *Id.* ¶¶ 89, 133.

72. It would have been obvious to a person of skill in the art reading Ideses that “any feature,” including “static meta data” would encompass the “predefined character strings in the executable file” required by claim 1 of the ’844 patent. As just a pair of examples, the “name” and “information regarding . . . the software package” described in Ideses would have been understood by one of skill in the art to be “predefined character strings” in the executable file that could have been extracted without decrypting or unpacking the executable file. For example, executable files in the well-known Windows Portable Executable (PE) format typically include headers that include resource strings that represent, for example, an “Internal Name,” “Company Name,” and “Product Name” of a given executable file. *See, e.g.*, Appendix B (metadata information extracted from the PE-format notepad.exe executable file using ExifTool); Ex. 1009 at 1 (“The goal of this research is to propose a novel model to detect unknown malware in Portable Executable (PE) . . . format since it is the most [widely] distributed type of malicious codes.”).

- c. **[1.b]: “generating a feature vector from the plurality of static data points using a classifier trained to classify the plurality of static data points based on a collection of data comprising known malicious executable files, known benign executable files, and known unwanted executable files, wherein the collection of data comprises at least a portion of the plurality of static data points, and wherein one or more features of the feature vector are selectively turned on or off based on whether a value of one or more static data points from the plurality of extracted static data points is within a predetermined range; and”**

73. It is my opinion that Ideses discloses each part of this limitation.

74. Limitation [1.b] first requires “generating a feature vector from the plurality of static data points using a classifier trained to classify the plurality of static data points.” This “generating” step is shown in Ideses’ Figure 2, which “is a flowchart of a method of run-time execution of a classifier based on a pruned feature list.” Ex. 1005 ¶ 123.

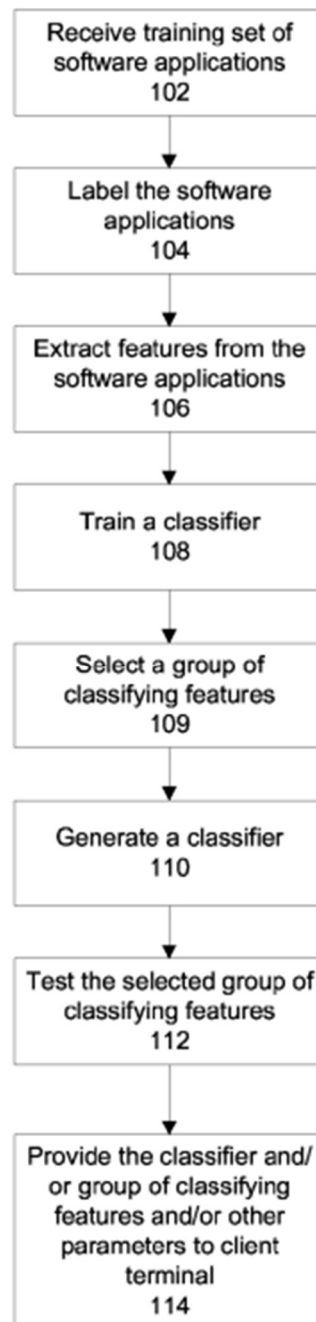


Id., Fig. 2.

75. Step 202 of Figure 2 shows that a “trained classifier . . . is received by the mobile device.” *Id.* ¶ 124. Ideses explains that the “features extracted” at step 208 of Figure 2 (“the plurality of static data points” required by limitation [1.a]), “are based on . . . the trained classifier received from the central server” in block 202. *Id.* ¶ 132. Thereafter, “the extracted features are collated into feature vectors.” *Id.* ¶ 134. Accordingly, Ideses discloses “generating a feature vector from the

plurality of static data points using a classifier trained to classify the plurality of static data points.”

76. Limitation [1.b] also requires that the “classifier” is “trained to classify the plurality of static data points based on a collection of data comprising known malicious executable files, known benign executable files, and known unwanted executable files.” Ideses’ Figure 1 shows “a method of generating a selected group of classifying features (e.g., feature vector) for classification of software applications.” *Id.* ¶ 71.



Id., Fig. 1.

77. Ideses explains that at step 102 of Figure 1, “multiple software applications for training the classifier are received,” which may be “manually

provided by an operator of the system (e.g., the manufacturer), provided by software application manufacturers, automatically downloaded from the internet, provided by updates from mobile devices, . . . and/or provided by other methods.” *Id.* ¶ 80.

78. At step 104 of Figure 1, “the multiple software applications are labeled with a classification type based on desired software application classification categories,” which “denote desired or undesired software applications.” *Id.* ¶ 81. A person of ordinary skill in the art would have understood Ideses’ disclosure of “malicious” software as corresponding to “known malicious executable files,” Ideses’ disclosure of “undesired software applications” as corresponding to “known unwanted executable files,” and “desired” software applications as corresponding to “known benign executable files.” Ideses also discloses other classification types, including “Adware, Goodware, [and] Intrusive Adware,” as well as “harmful, malware, bothersome, adware, intrusive, spyware, or other non-desirable classification types.” *Id.* ¶¶ 83, 141.

79. At step 106 of Ideses’ Figure 1, “features are extracted from the software applications,” including, for example, “data and/or meta data.” *Id.* ¶¶ 86-89. Subsequently, at step 108, “a classifier is trained based on the set of extracted features (block 106) and software classification labeling (block 104).” *Id.* ¶ 94. Thus, Ideses discloses “a collection of data comprising known malicious executable files,

known benign executable files, and known unwanted executable files” as required by limitation [1.b].

80. Limitation [1.b] also requires that “the collection of data comprises at least a portion of the plurality of static data points.” As I explained above, the features that are extracted in step 106 of Ideses’ Figure 1 “may be any feature that describes the software application,” including “static meta data regarding the application, for example, icon, name, rating or other features,” as well as “information regarding the executable code and/or software package, for example, in the form of byte code, resources, permissions, or other features.” *Id.* ¶¶ 86-89. These features can be “stored as feature vectors, *which may be used for training the classifier*,” as shown in step 108 of Ideses’ Figure 1. *Id.* ¶¶ 91, 94 (emphasis added).

81. Further, with respect to step 204 of Figure 2 (“Generate feature extractors”), Ideses explains that “the feature extractor modules are automatically built and coded based on the pruned features that have been selected by the central processor.” *Id.* ¶ 127. Ideses similarly explains that “[o]ptionally, the features extracted are based on the group of classifying features and/or trained classifier received from the central server (block 202).” *Id.* ¶ 132. Thus, Ideses discloses that “the collection of data comprises at least a portion of the plurality of static data points” as required by limitation [1.b].

Lastly, limitation [1.b] requires that “one or more features of the feature vector are selectively turned on or off based on whether a value of one or more static data points from the plurality of extracted static data points is within a predetermined range.” Ideses explains that in the method depicted in Figure 1, “[t]he full feature list is optionally selectively pruned,” that is, features are “selectively turned off” as required by limitation [1.b], “to generate a pruned feature list.” *Id.* ¶ 71. Further, “each feature of the group is selected from-the-ground-up; added to the group as opposed to removed from or retained within a larger group,” that is, features can also be “selectively turned on” as required by element [1.b]. *Id.*

Ideses explains that “[t]he group of classifying features may be selected based on one or more methods,” and that “the group of classifying features is selected to reduce the dimensionality of the feature vector.” *Id.* ¶¶ 104-05. For example, the group of classifying features may be selected “based on the significance of the extracted features,” such that significant features, or “those that have a large effect on the classification outcomes, may be retained,” whereas insignificant features, or “those that have a small, negligible or no effect on classification outcome may be pruned.” *Id.* ¶ 106.

Ideses discloses two approaches to pruning (selectively turning off) insignificant features. First, “features that correspond to coefficients having a value of zero are pruned.” *Id.* ¶ 107. Second, “features with a low magnitude value of the

coefficient are identified for pruning and/or are not included.” *Id.* Ideses explains that selection of the group of classifying features based on coefficients with zero/low values may be denoted by the following expressions, respectively:

$$\text{Feature}_i \in FV \text{ if } |(\text{Coefficient}_i)| \neq 0$$

$$\text{Feature}_i \in FV \text{ if } |(\text{Coefficient}_i)| \geq \alpha$$

Id. ¶¶ 109-10. Both approaches to selectively including/excluding features are based on whether a feature coefficient “is within a predetermined range” as required by limitation [1.b]. Further, Ideses discloses that these approaches can be used to generate a “pruned feature vector” (a vector in which features of the feature vector are selectively turned on or off) for classification by a classifier. *Id.* ¶ 157; *see also id.*, Fig. 5. Thus, Ideses discloses that “one or more features of the feature vector are selectively turned on or off based on whether a value of one or more static data points from the plurality of extracted static data points is within a predetermined range.”

d. [1.c]: “evaluating the feature vector using support vector processing to determine whether the executable file is harmful.”

82. It is my opinion that Ideses discloses this limitation.

83. As shown in Figure 2 of Ideses (reproduced above), at step 210, “the software application is classified . . . for example, as benign or adware.” *Id.* ¶ 135. Specifically, “the software application is classified by applying the trained classifier

received from the central server,” and that alternatively or additionally, “the software application is classified based on the group of classifying features that have been computed by the feature extractor” at step 208 of Figure 2. *Id.* ¶ 137.

84. Ideses discloses that “the classifier is trained based on supervised learning,” and lists “Support Vector Machines” as a specific example of a “software module[] to train the classifier.” *Id.* ¶ 96. Likewise, Ideses explains that “when classification is performed based on a support vector machine (SVM),” that “features that correspond to coefficients having a value of zero are pruned or not included.” *Id.* ¶ 107.

85. In my opinion, one of ordinary skill in the art at the time of filing of the ’844 patent would have understood Ideses’ description of the use of a “Support Vector Machine” for training a classifier for classifying software applications to be the same as “using support vector processing to determine whether the executable file is harmful,” as required by limitation [1.c].

2. Claim 2: “The computer-implemented method according to claim 1, wherein the extracting the plurality of static data points further comprises classifying the extracted static data points according to a type of data extracted from the executable file, and encoding the extracted static data points for the classifier based on the classification.”

86. It is my opinion that Ideses discloses the additional limitations of dependent claim 2.

87. As I explained above with respect to limitation [1.a], Ideses discloses that “[t]he extracted features may be any feature that describes the software application,” including “static meta data regarding the application, for example, icon, name, rating or other features.” Ex. 1005 ¶ 89. Further, “the extracted features may contain information regarding the executable code and/or software package, for example, in the form of byte code, resources, permissions, or other features.” *Id.* During classification, “the features extracted are based on the group of classifying features and/or trained classifier received from the central server,” as shown in step 208 of Figure 2 (reproduced above). *Id.* ¶ 132.

88. Further, Ideses discloses that “the features are stored as feature vectors, which may be used for training the classifier,” and that the extracted “features may be stored using other data structures, for example, a matrix, a list, or other suitable structures.” *Id.* ¶ 91.

89. Moreover, one of skill in the art would have understood that these “extracted features” corresponded to the various data types identified in the ’844 Patent, as shown in the table below:

Ideses (Ex. 1005)	'844 Patent (Ex. 1001)
“icon,” “byte code,” “resources”	“byte sequences”
“name”	“string sequences”
“rating”	“numeric values”
“permissions”	“nominal values”

Id. ¶ 89; Ex. 1001 at 6:3-6, Table 1.1.

90. In my opinion, it would have been obvious to one of skill in the art to “classify[] the extracted static data points according to a type of data extracted from the executable file” and “encod[e] the extracted static data points for the classifier based on the classification” as required by claim 2 by storing the extracted values in memory using data structures (or primitive data types) appropriate for the type of data (or category) of the extracted features. For example, byte sequences (such as the “icon,” “byte code,” or “resources” described in Ideses) would have commonly been stored using a byte array data structure. Similarly, string sequences (such as a “name”) would have been stored using an array of characters ending with a null terminator to identify the end of the string (i.e., a C-style string). Numeric values (such as an application “rating”) would have been stored using a primitive integer data type (or appropriate wrapper).

3. Claim 3: “The computer-implemented method according to claim 2, wherein generating the at least one feature vector comprises selectively setting features of the classifier based on the plurality of extracted static data points.”

91. It is my opinion that Ideses discloses the additional limitation of dependent claim 3.

92. As I explained above with respect to limitation [1.b], Ideses discloses that a client device can generate a “pruned feature vector” (i.e., a feature vector in which features of the feature vector are selectively turned on or off) for classification by a classifier. *See* Ex. 1005 ¶ 157, Fig. 5. In describing Figure 1, Ideses discloses that the “full feature list is optionally selectively pruned,” that is, “selectively turned off” as required by limitation [1.b], in order “to generate a pruned feature list.” *Id.* ¶ 71. Likewise, “each feature of the group is selected from-the-ground-up; added to the group as opposed to removed from or retained within a larger group,” that is, features can also be “selectively set[.]” (i.e., turned on) as required by element [1.b]. *Id.*

93. Claim 3 also requires “selectively setting features of the classifier *based on the plurality of extracted data points*,” which is also disclosed by Ideses. Ideses discloses selecting “a group of classifying features” with the goal of “reducing client terminal processor usage and/or client terminal memory requirements.” *Id.* ¶ 16. Because “a certain feature may have strong classification effectiveness and/or

discrimination capability, but may place large requirements in terms of memory and/or CPU usage,” such a feature may be omitted (or pruned) from the feature set in order to “reduc[e] processor usage . . . and/or reduc[e] memory requirements of the mobile device.” *Id.* ¶¶ 53-55. Conversely, a different feature with “slightly less classification effectiveness and/or discrimination capability, but significantly lower CPU and/or memory usage” may be selected (i.e., maintained in the feature set). *Id.*

94. Additionally, and as I explained above with respect to limitation [1.b], Ideses discloses two approaches to pruning insignificant features based on whether a feature coefficient is within a predetermined range. *See id.* ¶¶ 107-110. Ideses discloses other pruning methods, for example, where “the group of classifying features is selected based on solving a cost function” and where “the dimensionality reduction of the group of classifying features and/or feature pruning is selectively performed based on the tradeoff of improving classification ability, while reducing CPU requirements and/or while reducing memory usage.” *Id.* ¶¶ 111-14. Ideses explains that such a “cost function denotes a combination of classifier quality (i.e., ability to accurately classify) and the measure of complexity attributed to the features of the feature vector.” *Id.* ¶ 111.

95. One of skill in the art would have recognized that features having certain variable-length data types (such as byte sequences or character strings) would demand a greater computational cost if used in a feature vector, whereas other fixed-

length data types (such as numbers) would have a lower cost if used in a feature vector. For this reason, it would have been obvious to one of skill in the art reading Ideses to selectively turn off certain features of a classifier (such as potentially large byte sequences like icons or other application resources) while selectively turning on other features (such a numeric features like a rating, or relatively short string sequences like a software version or name) based on the type of data of the extracted feature (i.e., the “classification” of the extracted feature) in order to limit the use of scarce resources available to a client device. Indeed, Ideses suggests that where the “client . . . is a *resource-limited device*, having limited CPU availability, limited CPU power, and/or limited memory,” that the “set of extracted features is selected to take place during run-time, under the available resources of [the] client.” *Id.* ¶¶ 155-57 (emphasis added).

4. Claim 4: “The computer-implemented method according to claim 2, wherein the classifying the at least one static data point comprises classifying the plurality of extracted static data points into categories comprising numeric values, nominal values, string or byte sequences, and Boolean values.”

96. It is my opinion that Ideses renders the additional limitation of dependent claim 4 obvious.

97. As I explained above with respect to claim 2, Ideses discloses that “[t]he extracted features may be any feature that describes the software application,” including “static meta data regarding the application, for example, icon, name, rating

or other features.” *Id.* ¶ 89. Further, “the extracted features may contain information regarding the executable code and/or software package, for example, in the form of byte code, resources, permissions, or other features.” *Id.* During classification, “the features extracted are based on the group of classifying features and/or trained classifier received from the central server,” as shown in step 208 of Figure 2 (reproduced above). *Id.* ¶ 132.

98. Likewise, and similar to my explanation above with respect to claim 2, it would have been obvious to one of skill in the art to “classify[] the plurality of extracted static data points into categories comprising numeric values, nominal values, string or byte sequences, and Boolean values” by storing the extracted values in memory using data structures (or primitive data types) appropriate for the type of data (or category) of the extracted features. The four categories listed in claim 4 – numeric values, nominal values, string or byte sequences, and Boolean values – each correspond to data types that would have been well-known to one of skill in the art. As I explained above, byte sequences (such as the “icon,” “byte code,” or “resources” described in Ideses) would have commonly been stored using a byte array data structure, string sequences (such as a “name”) would have been stored as C-style strings, and numeric values (such as an application “rating”) would have been stored using a primitive integer data type.

99. In addition, it would have been obvious to one of skill in the art to “classify[] the plurality of extracted static data points into categories comprising . . . nominal values, . . . and Boolean values,” for example, by storing “permissions” as bit fields wherein each bit that is set to a value of ‘1’ indicates a granted permission (i.e., by using individual bits as flags). Boolean values have only two possible values: true or false. It would have been obvious to one of skill in the art to store such a value using as little as a single bit, or more simply by using a primitive unsigned integer type and according a value of ‘0’ to mean “false,” and any other numeric value as “true.”

5. Claim 5: “The computer-implemented method according to claim 4, wherein analyzing the executable file further comprises generating at least one feature vector from the plurality of extracted static data points, wherein features of the generated feature vector are weighted based at least upon the classified categories and the plurality of extracted static data points.”

100. It is my opinion that Ideses discloses the additional limitations of dependent claim 5.

As I explained above with respect to limitation [1.b], Ideses discloses that “the extracted features are collated into feature vector” as required by dependent claim 5. *Id.* ¶ 134. And, as I explained above, Ideses discloses two approaches to pruning (selectively turning off) insignificant features. First, “features that correspond to coefficients having a value of zero are pruned.” *Id.* ¶ 107. Second, “features with a

low magnitude value of the coefficient are identified for pruning and/or are not included.” *Id.* Ideses explains that selection of the group of classifying features based on coefficients with zero/low values may be denoted by the following expressions, respectively:

$$\text{Feature}_i \in FV \mid |(\text{Coefficient}_i)| \neq 0$$

$$\text{Feature}_i \in FV \mid |(\text{Coefficient}_i)| \geq \alpha$$

Id. ¶¶ 109-10.

101. The coefficient values described by Ideses are weights that indicate the significance of a given feature. For example, features having non-zero/high coefficient values are “significant” in the sense that they may “have a large effect on the classification outcomes,” and for that reason are retained in the feature vector. *See id.* ¶ 106. Conversely, features having zero/low coefficient values either “do not contribute (or do not significantly contribute) to the classification process” in the sense that their omission from the feature vector “does not affect the quality of the classification.” *Id.* ¶ 107. Thus, one of ordinary skill in the art would have understood Ideses’ disclosure of the selection of features based on these coefficients to have included features “weighted based at least upon the classified categories and the plurality of extracted static data points” as required by dependent claim 5.

- 6. Claim 6: “The computer-implemented method according to claim 1, wherein determining whether the executable file is harmful further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value.”**

102. It is my opinion that Ideses discloses the additional limitation of dependent claim 6.

103. Ideses discloses that at step 210 of Figure 2, “a certainty of the classification [of the software application] is provided, for example, the estimated probability that the classification is correct.” *Id.* ¶ 140. Ideses explains that the “decision to install or remove the software application may be based on the certainty level,” and that “high probability malicious software may be automatically (or manually) removed.” *Id.* ¶ 143. In my opinion, one of skill in the art would have understood the removal of “high probability malicious software” to be the same as “preventing execution of the executable file,” as required by dependent claim 6.

- 7. Claim 7: “The computer-implemented method according to claim 6, wherein the threshold value is set based on predetermined false positive range data.”**

104. It is my opinion that Ideses discloses the additional limitation of dependent claim 7.

105. As I explained above with respect to dependent claim 6, “a certainty of the classification [of the software application] is provided, for example, the estimated probability that the classification is correct.” *Id.* ¶ 140.

106. This “estimated probability that the classification is correct” is a measure of how accurate the classifier is. One of skill in the art would have known that one measure of classifier accuracy is the false positive rate (FPR), which is the proportion of negative cases that are incorrectly identified as positive cases, and is commonly and routinely used to measure classifier accuracy. In the context of Ideses, this would be the proportion of benign files incorrectly classified as malware, or the probability that a false positive will be raised by the classifier. The FPR is defined as the total number of negative cases incorrectly identified as positive cases (false positives) divided by the total number of actual negative cases (false positives + true negatives). Accordingly, it would have been obvious to one of skill in the art reading Ideses that “the estimated probability that the classification is correct” would have been based on the well-known FPR, which is “predetermined false positive range data” as required by dependent claim 7.

107. Additionally, one of skill in the art would have known that another measure of classifier accuracy is the false negative rate (FNR), which is the proportion of positive cases that are incorrectly identified as negative cases, and is also commonly and routinely used to measure classifier accuracy. In the context of Ideses, this would be the proportion of actual malware files incorrectly classified as benign, or the probability that a false negative will be raised by the classifier. The FNR is defined as the total number of positive cases incorrectly identified as negative

cases (false negatives) divided by the total number of actual positive cases (false negatives + true positives). It would have been further obvious to one of skill in the art reading Ideses that “the estimated probability that the classification is correct” would have been based not only on the FPR, but on the combination of the FNR and FPR (i.e., $1 - (FPR + FNR)$), which is also “predetermined false positive range data” as required by dependent claim 7.

8. Claim 8: “The computer-implemented method according to claim 1, wherein the plurality of static data points is extracted using a machine learning technique.”

108. It is my opinion that Ideses discloses the additional limitation of dependent claim 8.

109. As I explained above with respect to limitation [1.c], Ideses discloses the use of “Support Vector Machines,” which is a machine learning technique. *See id.* ¶ 96. More generally, Ideses discloses that “a learning module . . . of a machine learning algorithm is applied to train the classifier.” *Id.* ¶ 94. Likewise, Ideses explains that “the learning module performs the machine learning and/or classifier training.” *Id.* ¶ 95.

9. Claim 9: “The computer-implemented method according to claim 1, wherein a determination of whether the executable file is harmful is used to retrain the classifier.”

110. It is my opinion that Ideses discloses the additional limitation of dependent claim 9.

111. Ideses explains that following classification (step 210 of Figure 2), at step 214, “the mobile device reports back to the server,” and “the classification type outcome” (i.e., “harmful, malware, bothersome, adware, intrusive, spyware,” etc.) is reported. *Id.* ¶ 144. Ideses explains that “the central server learns about the existence of new software applications based on the feedback provided from the mobile device,” and that “the central server re-labels or confirms the labeling of existing software applications based on the provided feedback.” *Id.* ¶¶ 145-46. As an example, “classification results with high certainty received from multiple mobile devices may cause the central server to change the existing classification type, or to retain the existing classification type.” *Id.* ¶ 146.

112. Moreover, I note that it was well-known in the art at the time of filing of the ’844 patent that a machine learning model could (and would) be updated with new data. Such retraining can improve a model’s performance metrics and can help ensure that the training data remains a good representation of the live data to be classified.

10. Claim 10

113. It is my opinion that Ideses renders claim 10 obvious, for the reasons discussed below.

- a. **[10.pre]: “A computer-readable storage device containing instructions, that when executed on at least one processor, causing the processor to execute a process comprising:”**

114. I understand that a preamble typically does not limit a claim. However, to the extent that the preamble of claim 10 is limiting, it is my opinion that it is disclosed by Ideses for the same reasons I expressed above with respect to claim limitation [1.pre].

- b. **[10.a]: “extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the plurality of static data points represent predefined character strings in the executable file;”**

115. It is my opinion that Ideses discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.a].

- c. **[10.b]: “generating a feature vector from the plurality of static data points using a classifier trained to classify the plurality of static data points based on a collection of data comprising known malicious executable files, known benign executable files and known unwanted executable files, wherein the collection of data comprises at least a portion of the plurality of static data points, and wherein one or more features of the feature vector are selectively turned on or off based on whether one or more values of one or more static data points from the plurality of extracted static data points is within a predetermined range; and”**

116. It is my opinion that Ideses discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.b].

- d. **[10.c]: “evaluating the feature vector using support vector processing to determine whether the executable file is harmful.”**

117. It is my opinion that Ideses discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.c].

11. **Claim 11: “The computer-readable storage device according to claim 10, wherein the extracting of the plurality of static data points executed by the processor further comprises classifying the extracted static data points according to a type of data extracted from the executable file, and encoding the extracted static data points for the classifier based on the classifying.”**

118. It is my opinion that Ideses discloses these additional limitations of dependent claim 11 for the same reasons I expressed above with respect to dependent claim 2.

12. **Claim 12: “The computer-readable storage device according to claim 11, wherein generating the at least one feature vector comprises selectively setting features of the classifier based on the plurality of extracted static data points.”**

119. It is my opinion that Ideses discloses the additional limitation of dependent claim 12 for the same reasons I expressed above with respect to dependent claim 3.

- 13. Claim 13: “The computer-readable storage device according to claim 11, wherein the classifying the at least one static data point comprises classifying the plurality of extracted static data points into categories comprising numeric values, nominal values, string or byte sequences, and Boolean values.”**

120. It is my opinion that Ideses discloses the additional limitation of dependent claim 13 for the same reasons I expressed above with respect to dependent claim 4.

- 14. Claim 14: “The computer-readable storage device according to claim 13, wherein analyzing the executable file further comprises generating a feature vector from the plurality of extracted static data points, wherein features of the generated feature vector are weighted based at least upon the classified categories and the plurality of extracted static data points.”**

121. It is my opinion that Ideses discloses these additional limitations of dependent claim 14 for the same reasons I expressed above with respect to dependent claim 5.

- 15. Claim 15: “The computer-readable storage device according to claim 10, wherein determining whether the executable file is harmful further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value.”**

122. It is my opinion that Ideses discloses the additional limitation of dependent claim 15 for the same reasons I expressed above with respect to dependent claim 6.

16. Claim 16: “The computer-readable storage device according to claim 15, wherein the threshold value is set based on predetermined false positive range data.”

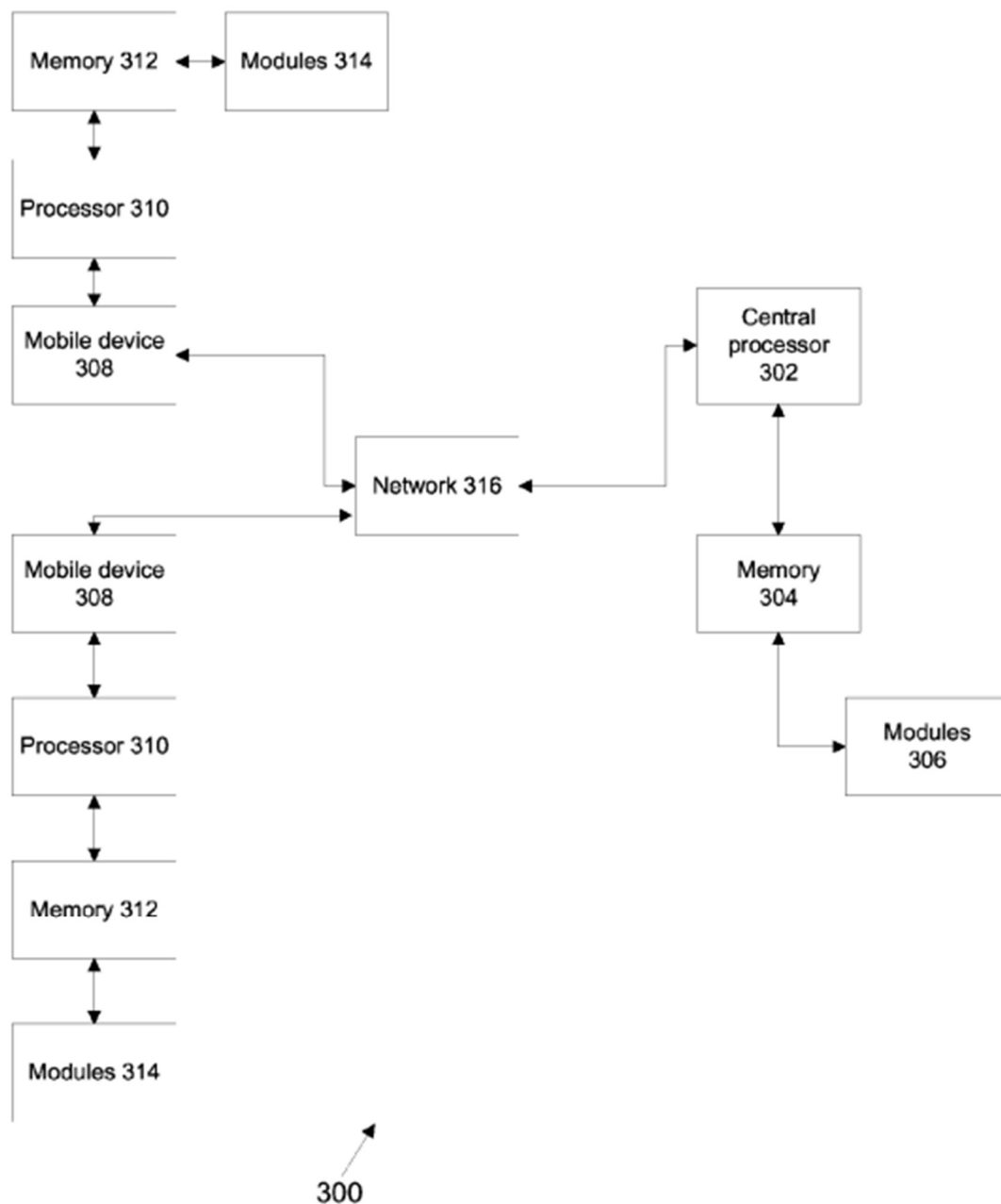
123. It is my opinion that Ideses discloses the additional limitation of dependent claim 15 for the same reasons I expressed above with respect to dependent claim 7.

17. Claim 17

a. [17.pre]: “A system comprising: at least one memory; and at least one processor operatively connected with the memory and configured to perform operation of:”

124. I understand that a preamble typically does not limit a claim. However, to the extent that the preamble of claim 17 is limiting, it is my opinion that it is disclosed by Ideses.

125. In addition to the reasons I expressed above with respect to limitation [1.pre], Ideses discloses “a system 300 for generating a selected group of classifying features for classification, and for classifying based on the selected group.” *Id.* ¶ 72. This system is depicted in Figure 3 of Ideses:



Id., Fig. 3.

126. As shown in Figure 3, “[s]ystem 300 includes a central processor 302 with a memory 304 for storing data modules 306 thereon.” *Id.* ¶ 73. Likewise, “[m]obile devices 308 may be resource limited, having smaller and/or less powerful

processor 310, and/or smaller and/or slower memory 312, [and] [m]odules 314 are stored on memory 312 for execution by processor 310.” *Id.* ¶ 75.

- b. **[17.a]: “extracting a plurality of predefined character strings from an executable file without decrypting or unpacking the executable file;”**

127. It is my opinion that Ideses discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.a].

- c. **[17.b]: “generating a feature vector from the plurality of predefined character strings using a classifier trained to classify the plurality of predefined character strings based on a collection of data comprising known malicious executable files, known benign executable files and known unwanted executable files, wherein the collection of data comprises at least a portion of one or more of the plurality of predefined character strings, and wherein one or more features of the feature vector are selectively turned on or off based on whether a value of one or more predefined character strings from the plurality of predefined character strings is within a predetermined range; and”**

128. It is my opinion that Ideses discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.b].

- d. **[17.c]: “evaluating the feature vector using support vector processing to determine whether the executable file is harmful.”**

129. It is my opinion that Ideses discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.c].

- 18. Claim 18: “The system according to claim 17, wherein the determining further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value, and wherein the threshold value is set based on predetermined false positive range data.”**

130. It is my opinion that Ideses discloses these additional limitations of dependent claim 18 for the same reasons I expressed above regarding dependent claims 6 and 7.

- 19. Claim 19: “The system according to claim 17, wherein the extracting the plurality of predefined character strings further comprises classifying the extracted plurality of predefined character strings according to a type of data extracted from the executable file, and encoding the extracted plurality of predefined character strings for the classifier based on the classification.”**

131. It is my opinion that Ideses discloses these additional limitations of dependent claim 19 for the same reasons I expressed above with respect to dependent claim 2.

- 20. Claim 20: “The system according to claim 17, wherein determining whether the executable file is harmful further comprises preventing execution of the executable file when a determined probability value that the executable file is harmful exceeds a threshold value.”**

132. It is my opinion that Ideses discloses the additional limitation of dependent claim 20 for the same reasons I expressed above with respect to dependent claim 6.

C. Ground 2: Claims 1, 8-10, and 17 are Obvious Over Mori

133. I have analyzed claims 1, 8-10, and 17 of the '844 Patent, and it is my opinion that each of these claims is obvious in light of Mori.

134. The Mori reference is a Japanese patent application publication that was published on February 9, 2012. Ex. 1006 at 14. Mori is entitled “Software Detection Methods and Devices and Programs,” and the applicants are listed as Nippon Telegraph and Telephone Corp. and Waseda University. *Id.* I understand that Mori is available as prior art because it was published prior to the effective filing date of the '844 patent (May 12, 2015).

135. Mori describes “a software detection device that detects malicious software (hereinafter referred to as ‘malware’).” *Id.* ¶ 10. Mori explains that “[m]alware often uses deliberately complex encryption and compression (called packing) to avoid detection by pattern matching or static analysis by inverse assemblers,” and that in general, “in order to perform a detailed analysis of malware, packed executable files need to be decrypted using various techniques.” *Id.* ¶¶ 3-4. Mori acknowledges, however, that methods “that use decryption, disassembly, and debuggers are time-consuming and require computational resources, and so are not suitable for use in detecting communications containing malware in real time.” *Id.* ¶ 7.

136. To solve the problem of detecting malware without the use of decryption, disassembly, and/or debuggers, Mori “focuses on readable strings contained in software files, collects information (characteristic vectors) that characterizes malware,” such as messages contained in unencrypted and uncompressed portions of “programs that constitute malware, host names, execution command names, function names and library names used for communication,” and “compares this content with strings contained in files being inspected to detect malware.” *Id.* ¶ 23.

137. Specifically, Mori discloses a malware detection method that includes a “a word extraction means that extracts as a word a string of printed and readable bytes” from a potential malware object, a “a feature extraction step wherein the feature extraction means configures feature vectors for each [potential malware] Object based on a set of words extracted by the word extraction step,” and a determination as to “whether or not the Object is malware based on newly observed feature vectors by applying a machine learning system to the feature vector.” *Id.* ¶ 16.

1. Claim 1

138. It is my opinion that Mori renders claim 1 obvious, for the reasons discussed below.

a. [1.pre]: “A computer-implemented method comprising:”

139. I understand that a preamble typically does not limit a claim. However, to the extent that the preamble of claim 1 is limiting, it is my opinion that it is disclosed by Mori.

140. Mori discloses “a software detection method, apparatus and program for detecting malware or communications transmitting and/or receiving malware using supervised machine learning systems.” Ex. 1006 ¶ 1. From this statement alone, one of skill in the art would have understood that Mori discloses a “computer-implemented method” as required by the preamble of claim 1.

141. Specifically, Mori discloses that “[t]he operation of each component of the detection device shown in FIG. 1 above can be programmatically constructed and then installed and executed on a computer utilized as the detection device.” *Id.* ¶ 67.

b. [1.a]: “extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the plurality of static data points represent predefined character strings in the executable file;”

142. It is my opinion that Mori discloses each part of this limitation.

143. Limitation [1.a] first requires “extracting a plurality of static data points from an executable file.” Mori discloses that in the first step of its malware detection

method, “the file or communication data to be sent or received (hereinafter, the file or communication data is collectively referred to as the ‘Object’) is recorded by the Object recording unit” *Id.* ¶ 28. Next, “a word extraction unit . . . extracts the printed and readable byte strings as words from the Object recorded above, and uses it as a word set.” *Id.* ¶ 29. As examples, Mori explains that “[r]eadable strings include messages contained in programs that make up malware, host names used for communication, execution command names, function names, library names, and so on, and can collect a variety of information about malware.” *Id.* ¶ 25. Mori’s method takes advantage of the fact that “[a]lthough the majority of malware today is known to develop a large number of variants after the original is developed, most of the code of the variants is not the same as the original, so it is likely that the characteristics of the printable strings will be similar to the original.” *Id.* ¶ 26.

144. Limitation [1.a] also requires that the “extracting” is performed “without decrypting or unpacking the executable file.” Mori explains that “[m]alware often uses deliberately complex encryption and compression (called packing) to avoid detection by pattern matching or static analysis by inverse assemblers,” and that “[p]acked executable files decrypt themselves at runtime and are then deployed in memory.” *Id.* ¶ 3. And, in general, “in order to perform a detailed analysis of malware, packed executable files need to be decrypted.” *Id.* ¶ 4.

145. Mori’s method is an effort to detect malware without using “decryption, disassembly, and debuggers,” as such techniques “are not suitable for use in detecting communications containing malware in real time.” *Id.* ¶ 7. Accordingly, Mori focuses on the analysis of “readable strings contained in software files,” which “can collect a variety of information about malware.” *Id.* ¶¶ 23-25. As I explained above, Mori discloses that “[r]eadable strings include messages contained in programs that make up malware, host names used for communication, execution command names, function names, library names, and so on.” *Id.* ¶ 25.

146. Limitation [1.a] further requires that “the plurality of static data points represent predefined character strings in the executable file.” Mori states that its method “focuses on readable strings contained in software files,” which “include messages contained in programs that make up malware, host names used for communication, execution command names, function names, library names, and so on, and can collect a variety of information about malware.” *Id.* ¶¶ 23-25. Mori explains that its method is useful for detecting malware “variants where only a part of the code has been modified,” because even though “the majority of malware today is known to develop a large number of variants after the original is developed, most of the code of the variants is not the same as the original, so it is likely that the characteristics of the printable strings will be similar to the original.” *Id.* ¶ 26.

147. Mori's word extracting unit uses a "word dictionary . . . prepared in advance" and "extract[s] only those words that match the words in word dictionary." *Id.* ¶ 50. One of skill in the art would have understood that the extracted words described in Mori "represent predefined character strings in the executable" to the extent that they match the words in a word dictionary "prepared in advance," or "predefined" as required by limitation [1.a].

- c. **[1.b]: "generating a feature vector from the plurality of static data points using a classifier trained to classify the plurality of static data points based on a collection of data comprising known malicious executable files, known benign executable files, and known unwanted executable files, wherein the collection of data comprises at least a portion of the plurality of static data points, and wherein one or more features of the feature vector are selectively turned on or off based on whether a value of one or more static data points from the plurality of extracted static data points is within a predetermined range; and"**

148. It is my opinion that Mori discloses each part of this limitation.

149. Limitation [1.b] first requires "generating a feature vector from the plurality of static data points using a classifier trained to classify the plurality of static data points." Mori explains that after the "word extraction unit . . . extracts the printed and readable byte strings as words from the Object recorded above, and uses it as a word set," that "a feature vector" is constructed "for each [potential malware] Object based on the above word set." *Id.* ¶¶ 29-30. Mori also discloses that "known

supervised learning methods can be used” can be used wherein “[t]eacher data . . . is given in advance,” which can then be used as “a highly accurate method for estimating [a] class . . . for the newly observed” feature vector. *Id.* ¶¶ 56-57. I note that although the English translation of Mori uses the term “teacher data,” other terms such as “*training* data” or “*training* set” are more commonly used to refer to the data used to build/train machine learning models.

150. Mori also explains that “[t]he supervised machine learning system . . . can be re-learned [i.e., re-trained] by providing feedback of new data each time an arbitrary configurable cycle arrives.” *Id.* ¶ 31.

151. Limitation [1.b] also requires that the “classifier” is “trained to classify the plurality of static data points based on a collection of data comprising known malicious executable files, known benign executable files, and known unwanted executable files.” As previously explained, Mori’s “known supervised learning methods can be used” with “[t]eacher data” (i.e., a training set) that is “given in advance.” *Id.* ¶¶ 56-57.

152. Mori details an example training scenario wherein 100 known malware files and 100 known “normal files” were used for machine learning. *See id.* ¶¶ 61-63. One of skill in the art would have understood Mori’s description of “malware” and “normal file” classification types to correspond to “known malicious executable files” and “known benign executable files,” respectively.

153. Mori also discloses that its machine learning technique is not limited to just two classes (i.e., “malware” and “normal”), but can also “classify other classes of malware and determine the malware type.” *Id.* ¶ 63. One of skill in the art would have understood Mori’s disclosure of “other classes of malware” to encompass, for example, “known unwanted executable files” such as adware that are not necessarily malicious or destructive, but are intrusive and therefore undesired.

154. Limitation [1.b] also requires that “the collection of data comprises at least a portion of the plurality of static data points.” As I explained above, Mori’s word extraction unit “uses [a] word dictionary . . . prepared in advance,” and “extract[s] only those words that match the words in [the] word dictionary.” *Id.* ¶ 50. Mori discloses that the feature vectors used for supervised machine learning likewise use the same “word dictionary.” *See id.* ¶¶ 59-62.

155. Lastly, limitation [1.b] requires that “one or more features of the feature vector are selectively turned on or off based on whether a value of one or more static data points from the plurality of extracted static data points is within a predetermined range.” Mori generates a feature vector for each Object based on a “word set” comprising the “printed and readable byte strings [extracted] as words from the Object” using, for example, “a static dictionary provided in advance.” *Id.* ¶¶ 29-32.

156. Mori explains that the feature vector can be defined “as whether a word existed or not by assigning it ‘0’ or ‘1.’” *Id.* ¶ 30. In other words, features are

selectively turned on or off based on whether or not a word in the word dictionary is present in the object or not.

157. Mori provides an example wherein the “[w]ord extract unit . . . uses word dictionary . . . prepared in advance for the word candidates shown in Table 2 above, *extracting only those words that match the words in [the] word dictionary.*” *Id.* ¶¶ 49-50 (emphasis added). Alternatively, “an Internet search engine is used to search for appropriate word candidates, with words extracted only when the number of search results *exceeds an arbitrary configurable threshold.*” *Id.* ¶ 50 (emphasis added).

158. Mori discloses the use of “known tf-idf values” as a configuration example for the generated feature vectors. *Id.* ¶ 53. A “tf-idf value” (term frequency-inverse document frequency) is a measure that reflects the importance or relevance of string representations in a single document amongst a collection (or corpus) of documents. “Term frequency” refers to the number of times the word appears in a document, and “inverse document frequency” represents how common (or uncommon) a word is among the corpus. Multiplying these values together results in tf-idf values that have higher scores for more important/relevant terms, and lower scores (approaching zero) as a term becomes less relevant. For example, Mori explains that “known tf-idf values can be used a configuration example of v_{ij} ,” or “ $v_{ij} = tf_{ij} * idf_{ij}$ ” for each word w_j in a file. *Id.*

159. Mori's Table 4 depicts the "result of analyzing the words that appear in each file as well as the word counts for each file." *Id.* ¶ 59.

Table 4

File name (hash value)	Appearance of word: number of times
00c4fc2fd0ac1d2b2fa43d8c5a087104	dll:4,get:2,sip:2,kernel:2,address:1,change:1,...
013a5ba10e3fc8a039b045530381d957	dll:4,get:2,kernel:2,rhd:1,bde:1,amu:1,range:1,...
notepad.exe	get:47>window:20,set:19,file:16,message:13,...
.....	

Id., Table 4.

160. Mori's Table 5 depicts "an example of configuring feature vectors" based on Table 4, wherein each word is numbered based on the word dictionary. *Id.* ¶ 60.

Table 5

File ID	Feature Vector	Class
1	119:0.0008, 196:0.0077, 307:0.0077,...	-1
2	119:0.0024, 168:0.0051, 213:0.0382,...	-1
3	84:0.006, 94:0.002, 113:0.00258,...	+1
....		

Id., Table 5.

161. As shown in Table 5, words that are present in each file are represented as "{word number: tf-idf value}" in the feature vector. *Id.* ¶ 60. One of skill in the art reading Mori would have understood that if a word is not present, the "static data

point” for that word is “turned off” since it is not included in the generated feature vector for that file (i.e., the tf-idf value is 0 because the tf portion is 0). Thus, Mori discloses that “one or more features of the feature vector are selectively turned on or off based on whether a value of one or more static data points from the plurality of extracted static data points is within a predetermined range.”

d. [1.c]: “evaluating the feature vector using support vector processing to determine whether the executable file is harmful.”

162. It is my opinion that Mori discloses this limitation.

163. Mori discloses the use of “support vector machines (SVMs)” as a supervised learning method for estimating a class from an observed feature vector, and indicates that SVMs were “widely used” and “known.” Ex. 1006 ¶¶ 57-58. For example, Mori’s Table 6 depicts an “example of the results of applying SVM to the above feature vector,” and shows 221 files classified as malware, and 1526 files classified as “normal” files. *Id.* ¶¶ 61-63.

164. In my opinion, one of ordinary skill in the art at the time of filing of the ’844 patent would have understood Mori’s description of the use of “support vector machines (SVMs)” as a supervised learning method for estimating a class from an observed feature vector to be the same as “using support vector processing to determine whether the executable file is harmful,” as required by limitation [1.c].

2. Claim 8: “The computer-implemented method according to claim 1, wherein the plurality of static data points is extracted using a machine learning technique.”

165. It is my opinion that Mori discloses the additional limitation of dependent claim 8.

166. As I explained above with respect to limitation [1.c], Mori discloses the use of “support vector machines (SVMs),” which is a machine learning technique. *See id.* ¶¶ 57-58. More generally, Mori discloses “determining whether [an] Object is malware based on newly observed feature vectors by applying a machine learning system to the feature vectors.” *Id.* ¶ 10. Likewise, Mori explains that “[b]y applying [the] supervised machine learning system . . . to the feature vector . . . , the file is classified as ‘malware,’ [or] ‘normal file.’” *Id.* ¶ 55.

3. Claim 9: “The computer-implemented method according to claim 1, wherein a determination of whether the executable file is harmful is used to retrain the classifier.”

167. It is my opinion that Mori discloses the additional limitation of dependent claim 9.

168. As I noted above with respect to limitation [1.b], Mori explains that the “[m]achine learning system . . . can also perform re-learning [i.e., re-training] via feedback of new data upon each arbitrary configurable cycle.” *Id.* ¶ 64. Mori explains that such “new data” refers to “data about files that are not real-time, such

as through prior art, or that, over time, have been determined to be malware or normal files.” *Id.*

4. Claim 10

169. It is my opinion that Mori renders claim 10 obvious, for the reasons discussed below.

- e. **[10.pre]: “A computer-readable storage device containing instructions, that when executed on at least one processor, causing the processor to execute a process comprising:”**

170. I understand that a preamble typically does not limit a claim. However, to the extent that the preamble of claim 10 is limiting, it is my opinion that it is disclosed by Mori for the same reasons I expressed above with respect to claim limitation [1.pre].

- f. **[10.a]: “extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the plurality of static data points represent predefined character strings in the executable file;”**

171. It is my opinion that Mori discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.a].

- g. [10.b]: “generating a feature vector from the plurality of static data points using a classifier trained to classify the plurality of static data points based on a collection of data comprising known malicious executable files, known benign executable files and known unwanted executable files, wherein the collection of data comprises at least a portion of the plurality of static data points, and wherein one or more features of the feature vector are selectively turned on or off based on whether one or more values of one or more static data points from the plurality of extracted static data points is within a predetermined range; and”**

172. It is my opinion that Mori discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.b].

- h. [10.c]: “evaluating the feature vector using support vector processing to determine whether the executable file is harmful.”**

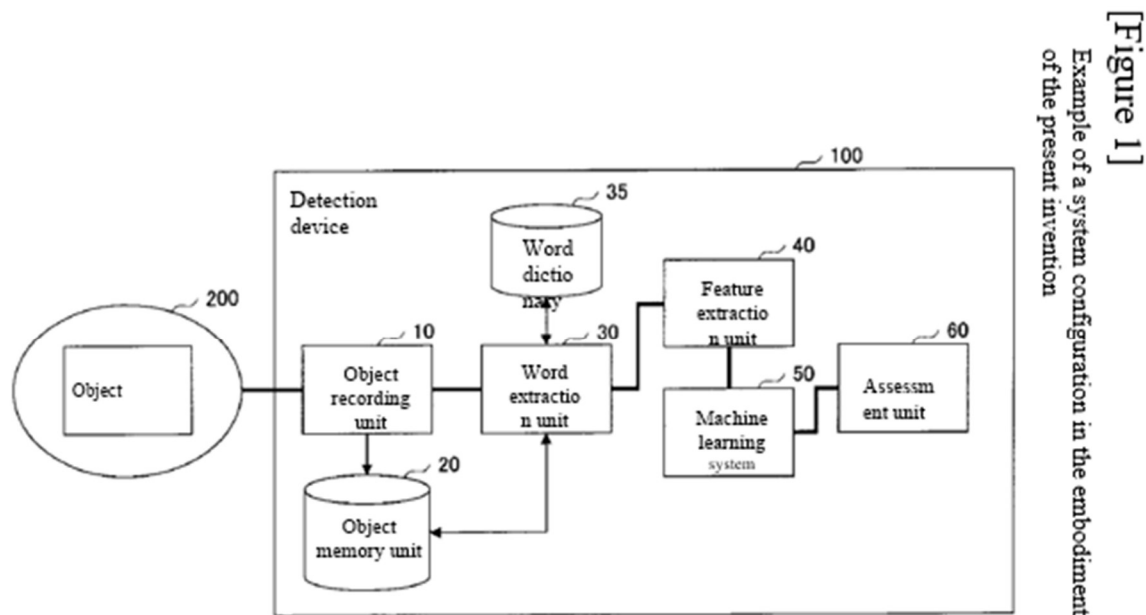
173. It is my opinion that Mori discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.c].

5. Claim 17

- i. [17.pre]: “A system comprising: at least one memory; and at least one processor operatively connected with the memory and configured to perform operation of:”**

174. I understand that a preamble typically does not limit a claim. However, to the extent that the preamble of claim 17 is limiting, it is my opinion that it is disclosed by Mori.

175. In addition to the reasons I expressed above with respect to limitation [1.pre], Mori discloses “a system configuration” that is depicted in Figure 1:



Id. ¶ 36, Fig. 1. Mori explains that “[d]etection device 100 is connected to network 200 and consists of Object recording unit 10, Object storage unit 20, word extraction unit 30, word dictionary 35, feature extract[ion] unit 40, machine learning system 50, and assessment unit 60,” and that “the Object storage unit 20 and word dictionary 35 are storage media such as a hard disk.” *Id.* ¶ 37.

- j. [17.a]: “extracting a plurality of predefined character strings from an executable file without decrypting or unpacking the executable file;”

176. It is my opinion that Mori discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.a].

- k. **[17.b]: “generating a feature vector from the plurality of predefined character strings using a classifier trained to classify the plurality of predefined character strings based on a collection of data comprising known malicious executable files, known benign executable files and known unwanted executable files, wherein the collection of data comprises at least a portion of one or more of the plurality of predefined character strings, and wherein one or more features of the feature vector are selectively turned on or off based on whether a value of one or more predefined character strings from the plurality of predefined character strings is within a predetermined range; and”**

177. It is my opinion that Mori discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.b].

- l. **[17.c]: “evaluating the feature vector using support vector processing to determine whether the executable file is harmful.”**

178. It is my opinion that Mori discloses each part of this limitation for the same reasons I expressed above with respect to claim limitation [1.c].

D. Ground 3: Claims 1-20 are Obvious Over Ideses in View of Mori

179. It is my opinion that the combination of Ideses and Mori renders claims 1-20 obvious, for the reasons discussed below.

180. It would have been obvious to one of skill in the art to combine the method of run-time classification of software applications disclosed in Ideses with Mori’s method of detecting encrypted/packed malware applications by focusing on

readable character strings. In particular, one of skill in the art would have been motivated to apply Mori's focus on readable character strings to Ideses' selection of features for extraction in order to generate a more accurate malware classifier that would have been capable of detecting malware without decrypting or unpacking an executable.

181. As I explained above in Grounds 1 and 2, it is my opinion that Ideses discloses (or renders obvious) all of the limitations of claims 1-20 and Mori discloses (or renders obvious) all of the limitations of claims 1, 8-10, and 17. However, if it is determined that Ideses does not sufficiently or explicitly disclose any part of any of the claim limitations of claims 1, 8-10, and/or 17, Mori discloses these features, as explained in Ground 2.

182. For example, with respect to [1.a], [10.a] ("extracting a plurality of static data points from an executable file without decrypting or unpacking the executable file, wherein the plurality of static data points represent predefined character strings in the executable file") or [17.a] ("extracting a plurality of predefined character strings from an executable file without decrypting or unpacking the executable file"), Mori discloses these features, as explained above in Ground 2.

183. Specifically, Mori discloses a technique that "focuses on readable strings contained in software files, collects information (characteristic vectors) that characterizes malware, such as messages contained in programs that constitute

malware, host names, execution command names, function names and library names used for communication” and that is distinguishable from other malware detection techniques that require decrypting or unpacking the executable file, which “are not suitable for use in detecting communications containing malware in real time.” Ex. 1006 ¶¶ 7, 23.

184. Mori acknowledges that “decryption, disassembly, and debuggers are time-consuming and require computational resources,” and explains that “detecting whether malware is present by executing it on an actual operating system and analyzing its behavior” is “time-consuming and requiring computational resources,” thus preventing “timely detection and [malware] determination.” *Id.*, ¶¶ 7-8.

185. Ideses acknowledges a similar problem faced by “resource limited” client devices that may be “unable to extract the full set of features during runtime, and/or unable to perform the extraction within a reasonable amount of time to allow for run-time operation.” Ex. 1005 ¶ 60.

186. In my opinion, a POSITA faced with the problem presented in Ideses (resource-limited client devices) would have been motivated to use Mori’s static analysis technique using “readable strings,” which can be used without “decryption, disassembly, or debuggers,” with the malware classification methods described in Ideses. The use of Mori’s “readable strings” technique would have helped in achieving Ideses’ stated goal of performing “feature extraction and

classification of a software application locally by a [resource limited] client terminal” by eliminating the “time-consuming” and resource-intensive decryption, disassembly, and/or debugging processes. *See* Ex. 1005 ¶ 6.

187. The combination of Ideses and Mori would have been nothing more than the use of a known technique (Mori’s focus on readable character strings for malware detection) to improve similar devices (Ideses malware detection system) in the same way to yield predictable results (more efficient malware classification, and the ability to detect malware without decrypting or unpacking an executable).

188. As a further example, and with respect to the “predefined character strings” limitations of [1.a], [1.b], [10.a], [10.b], [17.a], and [17.b], Mori discloses these features, as explained above in Ground 2.

189. Mori’s stated focus is on “readable strings contained in software files,” i.e., “predefined character strings” as required by the above-listed elements of the independent claims of the ’844 patent. Ex. 1006 ¶ 23. As examples of readable strings that can be collected from potential malware files, Mori identifies “messages contained in programs that constitute malware, host names, execution command names, function names and library names used for communication.” *Id.*; *see* Section VIII.B.1.b-c.

190. Ideses also teaches the extraction of “static meta data,” which can include readable strings. Ex. 1005 ¶ 89. As an example, Ideses identifies the application “name” as extractable static meta data. *Id.*; see Section VIII.C.1.b-c.

191. In my opinion, a POSITA would have been motivated to use Mori’s examples of predefined “readable strings,” such as messages, host names, execution command names, function names, and library names, in addition to the application “name” identified by Ideses. Indeed, Ideses does not limit its description of “extracted features,” but states that extracted features “may be *any feature* that describes the software application.” Ex. 1005 ¶ 89. The use of Mori’s exemplary “readable strings” would have improved Ideses’ system by enabling the generation of a more accurate classifier that would have been capable of detecting malware without decrypting or unpacking an executable.

IX. Secondary Considerations of Nonobviousness

192. As I explained above in Section V.B, I understand that an obviousness analysis includes consideration of “secondary” or “objective” evidence of nonobviousness, including things such as: (1) a long-felt but unmet need in the prior art that was satisfied by the claimed invention; (2) commercial success or the lack of commercial success of the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise of the claimed invention by others skilled in the art;

(5) taking of licenses under the patent by others; (6) deliberate copying of the claimed invention; and (7) contemporaneous and independent invention by others.

193. I am not aware of any secondary considerations of non-obviousness raised by Patent Owner with respect to the '844 patent. As an example, I am not aware of any showing of a long-felt but unmet need satisfied by the '844 patent. As I explained in my obviousness analysis above, any alleged “need” that would have been satisfied by the '844 patent had already been satisfied in light of Ideses, Mori, or both. As another example, I am not aware of any commercial success of any malware detection software due to its practicing of the claims of the '844 patent. I likewise am not aware of any evidence relating to the other “secondary” or “objective” considerations listed above.

194. Accordingly, the consideration of “secondary” or “objective” evidence of nonobviousness does not change my opinion that claims 1-20 of the '844 Patent would have been obvious to one of skill in the art.

X. Conclusion

195. Based on my review and above analysis of the prior art, it is my opinion that claims 1-20 of the '844 patent would have been obvious to one of skill in the art.

196. I declare under penalty of perjury that the foregoing is true and correct. I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: February 5, 2023



Gene Tsudik

Appendix A

Gene Tsudik
Distinguished Professor of Computer Science
Vice-Chair for Graduate Studies, Computer Science Department
University of California, Irvine, CA, 92697-3435 USA
tel: (949) 824-3410 email: gene.tsudik@uci.edu

Research Interests/Foci

- Computer/Network Security and Privacy
- Applied Cryptography

Education

- PhD in Computer Science, University of Southern California, April 1991.
- MS in Computer Science, University of Southern California, May 1987.
- BS in Computer Science, University of Houston - University Park, May 1985.

Employment History

- 07/19 - NOW: Distinguished Professor, Computer Science Department, University of California, Irvine.
- 07/13 - 06/19: Chancellor's Professor, Computer Science Department, University of California, Irvine.
- 07/03 - 06/13: Full Professor, Computer Science Department, University of California, Irvine.
- 01/00 - 03/03: Associate Professor, Computer Science Department, University of California, Irvine.
- 09/98 - 12/99: Research Associate Professor, Computer Science Department, University of Southern California.
- 04/96 - 12/99: Senior Research Scientist, Information Sciences Institute, University of Southern California.
- 04/91 - 03/96: Project Leader and Research Staff Member, IBM Zürich Research Laboratory.
- 08/87 - 04/91: Research Assistant, Computer Networks Laboratory, USC, University of Southern California.
- 06/85 - 05/90: Member of Technical Staff, IBM Los Angeles Scientific Center.

Visiting Positions

- 08/21 - 01/23: *Visiting Professor*, Swiss Federal Polytechnic Zürich (ETH).
- 12/17 - 12/18: *TÜV Süd Foundation Visiting Professor*, Technical University of Dresden, Germany.
- 01/16 - 12/16: *Distinguished/VIP Visiting Professor*, University of Padova, Italy.
- 05/14 - 06/14: *Astor Fellow*, Oxford University, United Kingdom.
- 09/09 - 09/11: *Cor Wit Foundation Visiting Chair*, Delft University of Technology, The Netherlands.
- 04/07 - 09/07: Fulbright Senior Scholar, University of Rome, "La Sapienza", Italy.
- 09/06 - 12/06: Senior Fellow, Institute for Pure & Applied Mathematics (IPAM), University of California, Los Angeles.

- [J1] D. Estrin, J. Mogul and G. Tsudik, *VISA Protocols for Controlling Inter-Organizational Datagram Flow*, IEEE Journal on Selected Areas in Communications, Vol. 7, No. 4, pp. 486-498, May 1989.
- [J2] G. Tsudik, *Datagram Authentication in Internet Gateways*, IEEE Journal on Selected Areas in Communications, Vol. 7, No. 4, 1989.
- [J3] D. Estrin and G. Tsudik, *An End-to-End Argument for Network Layer Inter-Domain Access Controls*, Journal of Internetworking: Research and Experience, June 1991.
- [J4] R. Summers and G. Tsudik, *AudES: An Expert System for Security Auditing*, Computer Security Journal, Vol. 6, No. 1, 1991.
- [J5] D. Estrin and G. Tsudik, *Secure Control of Transit Internetwork Traffic*, Computer Networks and ISDN Systems, October 1991.
- [J6] D. Estrin, M. Steenstrup and G. Tsudik, *Protocols for Route Establishment and Packet Forwarding Across Multi-Domain Internetworks*, ACM/IEEE Transactions on Networking, Vol. 1, No. 1, 1993.
- [J7] R. Molva, D. Samfat and G. Tsudik, *Authentication of Mobile Users*, IEEE Network, Vol. 8, No. 2, 1994.
- [J8] P. Janson and G. Tsudik, *Secure and Minimal Protocols for Authenticated Key Distribution*, Computer Communications, Vol. 18, No. 9, 1995.
- [J9] D. Chess, B. Grosz, C. Harrison, D. Levine, C. Parris and G. Tsudik, *Itinerant Programs for Mobile Computing*, IEEE Personal Communication Systems, Vol. 2, No. 5, 1995.
- [J10] R. Hauser, P. Janson, R. Molva, G. Tsudik and E. Van Herreweghen, *Robust and Secure Password/Key Change Method*, Journal of Computer Security, Vol. 4, No. 1, 1996.
- [J11] J. Gray, A. Kshemkalyani, M. Matyas, M. Peyravian and G. Tsudik, *ATM Cell Encryption and Key Update Synchronization*, Telecommunication Systems Journal, Vol. 7, No. 4, 1997.
- [J12] N. Asokan, G. Tsudik and M. Waidner, *Server-Supported Signatures*, Journal of Computer Security, Vol. 5, No. 1, 1997.
- [J13] R. Molva and G. Tsudik, *Secret Sets and Applications*, Information Processing Letters, Vol. 65, No. 1, 1998.
- [J14] R. Hauser, T. Przygienda and G. Tsudik, *Lowering Security Overhead in Link State Routing*, Computer Networks, Vol. 31, No. 8, 1999.
- [J15] G. Ateniese, A. Herzberg, H. Krawczyk and G. Tsudik, *Untraceable Mobility: How to Travel Incognito*, Computer Networks, Vol. 31, No. 8, 1999.
- [J16] G. Ateniese, M. Steiner and G. Tsudik, *New Multi-party Authentication Services and Key Agreement Protocols*, IEEE Journal on Selected Areas in Communications, Vol. 18, No. 4, 2000.
- [J17] M. Bellare, J. Garay, R. Hauser, A. Herzberg, H. Krawczyk, M. Steiner, G. Tsudik, E. Van Herreweghen, and M. Waidner, *Design, Implementation and Deployment of a Secure Account-Based Electronic Payment System*, IEEE Journal on Selected Areas in Communications, Vol. 18, No. 4, 2000.
- [J18] M. Steiner, G. Tsudik and M. Waidner, *Key Agreement in Dynamic Peer Groups*, IEEE Transactions on Parallel and Distributed Systems, Vol. 11, No. 8, 2000.
- [J19] T. Hardjono and G. Tsudik, *IP Multicast Security: Issues and Directions*, Annals de Telecom, July/August 2000.
- [J20] K. Obraczka, G. Tsudik and K. Viswanath, *Flooding for Reliable Multicast in Multi-Hop Ad Hoc Networks*, ACM Wireless Networks, Vol. 7, No. 6, 2001.
- [J21] Y. Kim, D. Mazzocchi and G. Tsudik, *Admission Control in Collaborative Groups (Abstract)*, ACM Mobile Computing and Communications Review, Vol. 6, No. 4, 2002.
- [J22] D. Boneh, X. Ding and G. Tsudik, *A Method for Fast Credentials Revocation*, ACM Transactions on Internet Technology, Vol. 4, No. 1, 2004.
- [J23] K. Bickel, G. Tsudik and B. Tung, *How to Construct Optimal One-Time Signatures*, Computer Networks, Vol. 43, No. 3, 2003.
- [J24] Y. Kim, F. Maino, M. Narasimha, K. Rhee and G. Tsudik, *Secure Group Services for Storage-Area Networks*, IEEE Communications, Vol. 41, No. 8, 2003.
- [J25] Y. Kim, A. Perrig and G. Tsudik, *Communication-Efficient Group Key Agreement*, IEEE Transactions on Computers, Vol. 53, No. 7, 2004.
- [J26] Y. Amir, Y. Kim, C. Nita-Rotaru, J. Stanton, J. Schultz and G. Tsudik, *Secure Group Communication Using Robust Contributory Key Agreement*, IEEE Transaction on Parallel and Distributed Systems, vol. 15, no. 5, 2004.
- [J27] Y. Kim, A. Perrig and G. Tsudik, *Tree-Based Group Key Agreement*, ACM Transactions on Information Systems Security, Vol. 7, No. 1, 2004.
- [J28] K. Obraczka, G. Tsudik and K. Viswanath, *Exploring Mesh- and Tree Based Multicast Routing Protocols for MANETs*, IEEE Transactions on Mobile Computing, Vol. 5, No. 1, 2006.
- [J29] Y. Amir, Y. Kim, C. Nita-Rotaru, G. Tsudik, *On the Performance of Group Key Agreement Protocols*, ACM Transactions on Information Systems Security, vol. 7, no. 3, 2004.
- [J30] K. Rhee, Y. Park and G. Tsudik, *A Group Key Management Architecture in Mobile Ad-hoc Wireless Networks*, Journal

- of Information Science and Engineering (JISE), vol. 21, no. 2, 2005.
- [J31] C. Castelluccia, S. Jarecki, J. Kim and G. Tsudik, *Secure Acknowledgment Aggregation and Multisignatures with Limited Robustness*, Computer Networks, Vol. 50, No. 10, 2006.
 - [J32] Y. Amir, C. Nita-Rotaru, J. Stanton and G. Tsudik, *Secure Spread: An Integrated Architecture for Secure Group Communication*, IEEE Transactions on Dependable and Secure Computing, Vol. 2, No. 3, 2005.
 - [J33] X. Ding, D. Mazzocchi and G. Tsudik, *Equipping "Smart" Devices With Public Key Signatures*, ACM Transactions on Internet Technology, Vol. 7, No. 1, 2007.
 - [J34] M. Narasimha, E. Mykletun and G. Tsudik, *Authentication and Integrity in Outsourced Databases*, ACM Transactions on Storage, Vol. 2, No. 2, 2006,
 - [J35] N. Saxena, G. Tsudik and J. Yi, *Threshold Cryptography in P2P and MANETs: the Case of Access Control*, Computer Networks, Vol. 51, No. 13, 2007.
 - [J36] N. Saxena, G. Tsudik and J. Yi, *Efficient Node Admission and Certificate-less Secure Communication in Short-Lived MANETs*, IEEE Transactions on Parallel and Distributed Systems, . Vol. 20, No. 2. 2009.
 - [J37] M. Goodrich, M. Sirivianos, J. Solis, C. Soriente, G. Tsudik and E. Uzun, *Loud and Clear: Human-Verifiable Authentication Based on Audio*, International Journal of Security and Networks, Vol. 4, No. 1, 2009.
 - [J38] C. Soriente, G. Tsudik and E. Uzun, *Secure Pairing of Interface-Constrained Devices*, International Journal of Security and Networks, Vol. 4, No. 1, 2009.
 - [J39] M. Narasimha, J. Solis and G. Tsudik, *Privacy-Preserving Revocation Checking*, International Journal of Information Security, Vol. 8, No. 1, 2009.
 - [J40] A. Chan, C. Castelluccia, E. Mykletun and G. Tsudik, *Efficient and Provably Secure Aggregation of encrypted data in WSN*, ACM Transactions on Sensor Networks, Vol. 5, No. 3., 2009.
 - [J41] X. Ding, G. Tsudik and S. Xu, *Leak-Free Group Signatures: Definitions and Construction*, Journal of Computer Security, Vol. 17, No. 4, 2009.
 - [J42] J. Kim and G. Tsudik, *SRDP: Secure Route Discovery for Dynamic Source Routing in MANETs*, Ad Hoc Networks, Vol. 7, No. 6, 2009.
 - [J43] D. Ma and G. Tsudik, *A New Approach to Secure Logging*, ACM Transactions on Storage, Vol. 5, No. 1, 2009.
 - [J44] M. Burmester, T. Van Le, B. de Medeiros and G. Tsudik, *Universally Composable RFID Identification and Authentication Protocols*, ACM Transactions on Information Systems Security, Vol. 12, No. 4, 2009.
 - [J45] R. Di Pietro, L. Mancini, C. Soriente, A. Spognardi and G. Tsudik, *Data Security in Unattended Wireless Sensor Networks*, IEEE Transactions on Computers, Special Issue on Autonomic Network Computing, Vol. 7, No. 8, 2009.
 - [J46] D. Ma, C. Soriente and G. Tsudik, *New Adversary and New Threats in Unattended Sensors Networks*, IEEE Network, Vol. 23, No. 2, 2009.
 - [J47] R. Di Pietro, L. Mancini, C. Soriente, A. Spognardi and G. Tsudik, *Maximizing Data Survival in Unattended Wireless Sensor Networks against a Focused Mobile Adversary*, Ad Hoc Networks, Vol. 7, No. 8, 2009.
 - [J48] G. Ateniese, J. Camenisch, M. Joye and G. Tsudik, *Remarks on "Analysis of One Popular Group Signature Scheme" in Asiacrypt 2006*, Journal of Applied Cryptography, Vol. 1, No. 4, 2009.
 - [J49] A. Kumar, N. Saxena, G. Tsudik and E. Uzun, *A Comparative Study of Secure Device Pairing Methods*, Pervasive and Mobile Computing, Vol. 5, No. 6, December 2009.
 - [J50] S. Jarecki, J. Kim and G. Tsudik, *Flexible Robust Group Key Agreement*, IEEE Transactions on Parallel and Distributed Systems, Vol. 22, No 5, March 2011.
 - [J51] D. Ma and G. Tsudik, *Security and Privacy in Emerging Wireless Networks*, IEEE Wireless Communications, Vol.17, No. 5, October 2010.
 - [J52] K. ElDefrawy and G. Tsudik, *ALARM: Anonymous Location-Aided Routing in Suspicious MANETs*, IEEE Transactions on Mobile Computing, Vol. 10, No. 9, September 2011. Selected as the **Spotlight Paper**.
 - [J53] R. Di Pietro, C. Soriente, A. Spognardi and G. Tsudik, *Intrusion-Resilient Integrity in Data-Centric Unattended WSNs*, Pervasive and Mobile Computing, Vol. 7, No. 4, August 2011.
 - [J54] R. Nithyanand, G. Tsudik and E. Uzun, *User-aided Reader Revocation in PKI-Based RFID Systems*, Journal of Computer Security, Vol. 19, No. 6, December 2011.
 - [J55] K. ElDefrawy and G. Tsudik, *PRISM: Privacy-friendly Routing In Suspicious MANETs*, IEEE JSAC, Special Issue on Advances in Military Communications and Networking, Vol. 29, No. 10, December 2011.
 - [J56] Y. Lu and G. Tsudik, *Privacy-Preserving Cloud Database Querying*, Journal of Internet Services and Information Security, Vol. 1 No. 4, November 2011.
 - [J57] M. Conti, S. Das, C. Bisdikian, M. Kumar, L. Ni, A. Passarella, G. Roussos, G. Tröster, G. Tsudik and F. Zambonelli, *Looking Ahead in Pervasive Computing: Challenges and Opportunities in the Era of Cyber-Physical Convergence*, Pervasive and Mobile Computing, Vol. 8, No. 12, February 2012.
 - [J58] R. Di Pietro, D. Ma, C. Soriente and G. Tsudik, *Self-Healing in Unattended Wireless Sensor Networks*, ACM Transactions on Sensor Networks, Vol. 9, No. 1, November 2012.
 - [J59] J. Cheon, C. Lim and G. Tsudik, *Reducing RFID Reader Load with the Meet-in-the-Middle Strategy*, Journal of

- [J60] R. Di Pietro, G. Oligieri, C. Soriente and G. Tsudik, *United We Stand: Intrusion-Resilience in Mobile Unattended WSNs*, IEEE Transactions on Mobile Computing, Vol. 12, No. 7, July 2013.
- [J61] M. Al Mishari, E. De Cristofaro, K. ElDefrawy and G. Tsudik, *Harvesting SSL Certificate Data to Mitigate Web-Fraud*, International Journal of Network Security, Vol. 14, No. 6, November 2012.
- [J62] A. Kobsa, R. Nithyanand, G. Tsudik and E. Uzun, *Can Jannie Verify: Usability of Display-Equipped RFID Tags for Security Purposes*, Journal of Computer Security, Vol. 21, No. 3, July 2013.
- [J63] E. De Cristofaro, C. Soriente, G. Tsudik and A. Williams, *Tweeting with Hummingbird: Privacy in Large-Scale Micro-Blogging OSNs*, IEEE Data Engineering Bulletin, Special Issue on Cloud Security and Privacy, Vol. 35, No. 4, December 2012.
- [J64] M. Almishari, P. Gasti, N. Nathan and G. Tsudik, *Optimizing Bi-Directional Low-Latency Communication in Named Data Networking*, ACM Computer Communications Review, Vol. 44, No. 1, January 2014.
- [J65] C. Ghali, G. Tsudik and E. Uzun, *Network-Layer Trust in Named-Data Networking*, ACM Computer Communications Review, Vol. 44, No. 5, October 2014.
- [J66] S. Ali, V. Sivaraman, D. Ostry, G. Tsudik and S. Jha, *Securing First-Hop Data Provenance for Bodyworn Devices using Wireless Link Fingerprints*, IEEE Transactions on Information Forensics and Security, Vol. 9, No. 12, September 2014.
- [J67] E. Ayday, E. De Cristofaro, JP Hubaux and G. Tsudik, *Whole Genome Sequencing: Revolutionary Medicine or Privacy Nightmare?*, IEEE Computer, Vol 48, No. 2, February 2015.
- [J68] E. Ngai, B. Ohlman, G. Tsudik, E. Uzun, M. Whlisch and C. Wood, *Can We Make a Cake and Eat it Too? A Discussion of ICN Security and Privacy*, ACM Computer Communication Review, Vol. 47, No. 1, January 2017.
- [J69] I. Martinovic, K. Rasmussen, M. Roeschlin and G. Tsudik, *Authentication Using Pulse-Response Biometrics*, Communications of the ACM (CACM): Research Highlights, Vol. 60, No. 2, February 2017.
- [J70] G. Ács, M. Conti, C. Ghali, P. Gasti, G. Tsudik and C. Wood, *Privacy-Aware Caching in Information-Centric Networking*, IEEE Transactions on Dependable and Secure Computing, Vol. 16, No. 2, March 2019.
- [J71] I. Martinovic, K. Rasmussen, M. Roeschlin and G. Tsudik, *Pulse-Response: Exploring Human Body Impedance for Biometric Recognition*, ACM Transactions on Privacy and Security (TOPS), Vol. 20, No. 2, May 2017.
- [J72] G. Tsudik, *Authorship Integrity and Attacks*, IEEE Security & Privacy, Vol. 15, No. 4, July/August 2017.
- [J73] T. Bradley, X. Ding and G. Tsudik, *Genomic Security (Lest We Forget)*, IEEE Security & Privacy, Vol. 15, No. 5, September/October 2017.
- [J74] B. Berg, T. Kaczmarek, A. Kobsa and G. Tsudik, *An Exploration of the Effects of Sensory Stimuli on the Completion of Security Tasks*, IEEE Security & Privacy, Vol. 15, No. 6, November/December 2017.
- [J75] M. Ambrosin, A. Compagno, M. Conti, C. Ghali and G. Tsudik, *Security and Privacy Analysis of Prominent Future Internet Architectures*, IEEE Communications Surveys and Tutorials, Vol. 20, No.2, January 2018.
- [J76] X. Ding and G. Tsudik, *Initializing Trust in Smart Devices via Presence Attestation*, Computer Communications: 40th Anniversary Edition, Vol. 131, No. 1, October 2018.
- [J77] X. Carpent, N. Rattanavipanont and G. Tsudik, *Remote Attestation via Self-Measurement*, ACM Transactions on Design Automation of Electronic Systems (TODAES), Vol. 24, No. 1, January 2019. **NOTE:** greatly expanded version of C184.
- [J78] N. Asokan, T. Nyman, N. Rattanavipanont, A. Sadeghi and G. Tsudik, *ASSURED: Architecture for Secure Software Update of Realistic Embedded Devices*, ACM Transactions on Computer-Aided Design (TCAD), Vol. 37, No. 11, November 2018. **NOTE:** same as C193.
- [J79] S. Jha, H. Tan and G. Tsudik, *MTRA: Multi-Tier Randomized Remote Attestation in IoT Networks*, Computers & Security, Vol. 81, No. 1, March 2019. **NOTE:** substantially extended version of C177.
- [J80] K. Balagani, M. Cardaioli, M. Conti, P. Gasti, M. Georgiev, T. Gurtler, D. Lain, C. Miller, K. Molas, N. Samarin, E. Saraci, G. Tsudik and L. Wu, *PILOT: Password and PIN Information Leakage from Obfuscated Typing Videos*, Journal of Computer Security, Vol. 27, No. 4, June 2019. **NOTE:** substantially extended version of C190.
- [J81] S. Ceconello, A. Compagno, M. Conti, D. Lain and G. Tsudik, *Skype & Type: Keyboard Eavesdropping in Voice-over-IP*, ACM Transactions on Privacy and Security (TOPS), Vol. 22, No. 4, December 2019. **NOTE:** substantially extended version of C170.
- [J82] C. Ghali, G. Tsudik and E. Uzun, *In Content We Trust: Network-Layer Trust in Content-Centric Networking*, IEEE/ACM Transactions on Networking (ToN), Vol. 27, No. 5., October 2019. **NOTE:** substantially extended version of J65.
- [J83] Y. Nakatsuka, A. Pavard and G. Tsudik, *PDoT: Private DNS-over-TLS with TEE Support*, ACM Digital Threats: Research and Practice (DTRAP), Vol. 2, No. 1, February 2021. **NOTE:** substantially extended version of C203.

Book Chapters and Edited Volumes (oldest → newest)

- [B1] D. Chess, B. Grosz, C. Harrison, D. Levine, C. Parris and G. Tsudik, *Chapter: Itinerant Programs for Mobile Com-*

- puting, (Reprint from J9), in: *in Readings in Agents*, M. Huhns and M. Singh, editors. Morgan-Kaufman, 1997 ISBN: 1558604952.
- [B2] G. Tsudik, editor, *Proceedings of the 6th ACM Conference on Computer and Communications Security*, ACM Press, 1999, ISBN: 1581131488.
- [B3] S. Kent and G. Tsudik, editors. *Proceedings of the 1999 Network and Distributed System Security Symposium*, The Internet Society, 1999, ISBN: 1891562045 and 1891562053.
- [B4] A. Rubin and G. Tsudik, editors. *Proceedings of the 2000 Network and Distributed System Security Symposium*, The Internet Society, 2000, ISBN: 189156207X and 1891562088.
- [B5] G. Tsudik, editor. *Special Issue: Selected Papers from ACM CCS 1999*, ACM Transactions on Information and System Security, Vol. 3, No. 2, August 2000. ISSN: 1094-9224.
- [B6] G. Tsudik, editor. *Special Issue: Selected Papers from NDSS 2000*, ACM Transactions on Information and System Security, Vol. 4, No. 2, May 2001. ISSN: 1094-9224.
- [B7] N. Saxena, G. Tsudik and J. Yi, *Chapter: Experimenting with Admission Control in P2P Networks* in: *Computer Security in the 21st Century*, Springer, 2005, ISBN: 0387240055.
- [B8] G. Tsudik, D. Gollmann and L. Gong, editors. *Proceedings of the 2005 IEEE Conference on Security and Privacy in Communication Networks*, IEEE Computer Society Press, 2005, ISBN: 0769523692.
- [B9] R. Molva, G. Tsudik and D. Westhoff, editors. *Proceedings of the 2nd European Workshop on Security and Privacy in Ad-hoc and Sensor Networks*, Springer, 2006, ISBN: 3540309128.
- [B10] G. Ateniese and G. Tsudik, *Chapter: Group Signatures*, in: *Cryptographic Protocols: Techniques for Secure Protocol Design*, M. Jakobsson and W. Mao, editors. Prentice-Hall, 2006, ISBN: 0131484710.
- [B11] G. Tsudik and S. Kent, editors. *Proceedings of the 2006 IEEE Conference on Security and Privacy in Communication Networks*, IEEE Computer Society Press, 2006, ISBN: 1424404231.
- [B12] G. Tsudik, S. Xu and M. Yung, editors. *Proceedings of ACM Workshop on Scalable Trusted Computing*, ACM Press, 2006, ISBN: 1595935487.
- [B13] G. Tsudik and C. Nita-Rotaru, editors. *Proceedings of IEEE Workshop on Secure Network Protocols*, IEEE Computer Society Press, 2006, ISBN: 1424407737.
- [B14] G. Tsudik, editor. *Proceedings of 2008 International Conference on Financial Cryptography and Data Security (FC'08)*, Springer Verlag, LNCS 5143, 2008, ISBN: 3540852298.
- [B15] G. Tsudik, editor. *Special Issue: Selected Papers from IEEE Securecomm'06*, ACM Transactions on Information and Systems Security, April 2008.
- [B16] S. Jarecki and G. Tsudik, editors. *Proceedings of 2009 12th International Conference on Practice and Theory in Public Key Cryptography*, Springer Verlag, LNCS 5443, 2009.
- [B17] M. Burmester, G. Tsudik, S. Magliveras and I. Ilic, editors. *Proceedings of 2010 13th Information Security Conference*, Springer Verlag, LNCS 6531, 2011, ISBN 9783642181771.
- [B18] S. Stolfo and G. Tsudik, editors. *Special Issue: Privacy-Preserving Sharing of Sensitive Information*, IEEE Security and Privacy, Vol. 8, No. 4, July 2010.
- [B19] J. Lopez and G. Tsudik, editors. *Proceedings of 9th International Conference on Applied Cryptography and Network Security, (ACNS'11)*, Springer Verlag, LNCS 6715, 2011, ISBN 9783642215537.
- [B20] N. Asokan and G. Tsudik, editors. *Proceedings of 4th ACM Conference on Wireless Network Security (WiSEC)*, ACM Press, 2011, ISBN: 9781450306928.
- [B21] D. Lin, G. Tsudik and X. Wang, editors. *Proceedings of 10th International Conference on Cryptology and Network Security (CANS)* Springer Verlag, LNCS 7092, ISBN: 9783642255120
- [B22] E. Fleischmann, H. Teufel and G. Tsudik, editors. US National Cyber Leap Year Summit 2009, Co-Chairs Report: Chapter 5, Digital Provenance, The Federal Networking and Information Technology Research and Development (NITRD) Program, September 2009. Available from: <http://www.nitrd.gov/NCLYSummit.aspx>

Conference & Workshop Publications (oldest → newest)

- [C1] D. Estrin and G. Tsudik, *VISA Scheme for Inter-Organizational Network Security*, IEEE Symposium on Research in Security and Privacy (S&P), 1987.
- [C2] D. Estrin and G. Tsudik, *Security Issues in Policy Routing*, IEEE Symposium on Research in Security and Privacy (S&P), 1989.
- [C3] D. Estrin and G. Tsudik, *Policy Enforcement in Interconnected Autonomous Networks*, DIMACS Workshop on Connections Between Distributed Computing and Cryptography, 1989.
- [C4] G. Tsudik, *Message Authentication with One-Way Hash Functions*, IEEE INFOCOM, 1992.
- [C5] G. Tsudik, *Policy Enforcement in Stub Autonomous Domains*, European Symposium on Research in Computer Security (ESORICS), 1992.
- [C6] R. Molva, G. Tsudik, E. Van Herreweghen and S. Zatti, *KryptoKnight Authentication and Key Distribution System*, European Symposium on Research in Computer Security (ESORICS), 1992.

- [C7] R. Molva and G. Tsudik, *Authentication Method with Impersonal Token Cards*, IEEE Symposium on Research in Security and Privacy (S&P), 1993.
- [C8] G. Tsudik and E. Van Herreweghen, *Some Remarks on Protecting Weak Secrets and Poorly-Chosen Keys from Guessing Attacks*, IEEE Symposium on Reliable Distributed Systems (SRDS), 1993.
- [C9] G. Tsudik and E. Van Herreweghen, *On Simple and Secure Key Distribution*, ACM Conference on Computer and Communications Security (CCS), 1993.
- [C10] R. Hauser, P. Janson, R. Molva, G. Tsudik and E. Van Herreweghen, *Robust and Secure Password/Key Change Method*, European Symposium on Research in Computer Security (ESORICS), 1994.
- [C11] R. Molva, D. Samfat and G. Tsudik, *An Authentication Protocol for Mobile Users*, IEE Colloquium on Security and Cryptography Applications to Radio Systems, 1994.
- [C12] A. Herzberg, H. Krawczyk and G. Tsudik, *On Traveling Incognito*, IEEE Mobile Computing Systems and Applications, 1994.
- [C13] M. Bellare, R. Hauser, A. Herzberg, J. Garay, H. Krawczyk, M. Steiner, G. Tsudik and M. Waidner, *iKP – A Family of Secure Electronic Payment Protocols*, USENIX Workshop on Electronic Commerce, 1995.
- [C14] R. Hauser and G. Tsudik, *On Shopping Incognito*, USENIX Workshop on Electronic Commerce, 1996.
- [C15] M. Steiner, G. Tsudik and M. Waidner, *Diffie-Hellman Key Distribution Extended to Groups*, ACM Conference on Computer and Communications Security (CCS), 1996.
- [C16] C. Gülcü and G. Tsudik, *Mixing E-mail with Babel*, ISOC Symposium on Network and Distributed System Security (NDSS), 1996.
- [C17] R. Hauser, T. Przygienda and G. Tsudik, *Low-cost Security in Link State Routing*, Security in Communication Networks (SCN), 1996.
- [C18] N. Asokan, G. Tsudik and M. Waidner, *Server-Supported Signatures*, European Symposium on Research in Computer Security (ESORICS), 1996.
- [C19] R. Hauser, T. Przygienda and G. Tsudik, *Reducing the Cost of Security in Link State Routing*, ISOC Symposium on Network and Distributed System Security (NDSS), 1997.
- [C20] P. Janson, G. Tsudik and M. Yung, *Scalability and Flexibility in Authentication Services*, IEEE INFOCOM, 1997.
- [C21] M. Steiner, G. Tsudik and M. Waidner, *CLIQUEs: A New Approach to Group Key Agreement*, IEEE International Conference on Distributed Computing Systems (ICDCS), 1998.
- [C22] M. Franklin and G. Tsudik, *Secure Group Barter*, Financial Cryptography and Data Security Conference (FC), 1998.
- [C23] K. Obraczka and G. Tsudik, *Multicast Routing Issues in Ad Hoc Networks*, IEEE International Conference on Universal Personal Communication (ICUPC), 1998.
- [C24] I. Foster, C. Kesselman, G. Tsudik and S. Tuecke, *Security Architecture for Large-Scale Distributed Computations*, ACM Conference on Computer and Communications Security (CCS), 1998.
- [C25] G. Ateniese, M. Steiner and G. Tsudik, *Authenticated Group Key Agreement and Friends*, ACM Conference on Computer and Communications Security (CCS), 1998.
- [C26] G. Ateniese and G. Tsudik, *Group Signatures a' la Carte (Short Paper)*, ACM/SIAM Symposium on Discrete Algorithms (SODA), 1999.
- [C27] C. Ho, K. Obraczka, G. Tsudik and K. Viswanath, *Flooding for Reliable Multicast in Multi-Hop Ad Hoc Networks*, ACM/NSF Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, 1999.
- [C28] C. Landwehr, M. Reed, P. Syverson and G. Tsudik, *Towards Analysis of Onion Routing Security*, Workshop on Design Issues in Anonymity and Unobservability, 2000.
- [C29] G. Ateniese and G. Tsudik, *Some Open Issues and New Directions in Group Signatures*, Financial Cryptography and Data Security Conference (FC), 2000.
- [C30] G. Ateniese, M. Joye and G. Tsudik, *On the Difficulty of Coalition Resistance in Group Signature Schemes*, Security in Communication Networks (SCN), 1999.
- [C31] Y. Amir, G. Ateniese, D. Hasse, Y. Kim, C. Nita-Rotaru, T. Schlossnagle, J. Schultz, J. Stanton and G. Tsudik, *Secure Group Communication in Asynchronous Networks with Failures: Integration and Experiments*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2000.
- [C32] G. Ateniese, O. Chevassut, D. Hasse, Y. Kim and G. Tsudik, *The Design of a Group Key Management API*, DARPA Information Security Conference and Exposition (DISCEX), 2000.
- [C33] G. Ateniese, J. Camenisch, M. Joye and G. Tsudik, *A Practical and Provably Secure Coalition-Resistant Group Signature Scheme*, IACR CRYPTO, 2000.
- [C34] K. Bacakci, B. Tung and G. Tsudik, *On Constructing Optimal One-Time Signatures*, International Symposium on Computer and Information Sciences (ISCIS), 2000.
- [C35] Y. Kim, A. Perrig and G. Tsudik, *Simple and Fault-Tolerant Key Agreement for Dynamic Collaborative Groups*, ACM Computer and Communication Security Conference (CCS), 2000.
- [C36] D. Boneh, X. Ding, G. Tsudik and B. Wong, *Fast Revocation of Security Capabilities*, Usenix Security Symposium, 2001.

- [C37] D. Agarwal, O. Chevassut, M. Thompson and G. Tsudik, *An Integrated Solution for Secure Group Communication in Wide-Area Networks*, IEEE Symposium on Computers and Communications (ISCC), 2001.
- [C38] Y. Kim, A. Perrig and G. Tsudik *Communication-Efficient Group Key Agreement*, IFIP-SEC, 2001.
- [C39] Y. Amir, Y. Kim, C. Nita-Rotaru, J. Schultz, J. Stanton and G. Tsudik, *Exploring Robustness in Group Key Agreement*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2001.
- [C40] K. Obraczka, G. Tsudik and K. Viswanath (Short Paper), *Pushing the Limit of Multicast in Ad Hoc Networks*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2001.
- [C41] D. Boneh, X. Ding and G. Tsudik, *Simple Identity-Based Mediated RSA*, Workshop on Information Security Applications (WISA), 2002.
- [C42] Y. Park, K. Rhee and G. Tsudik, *Group Key Management in Hierarchical Ad-hoc Wireless Networks*, Workshop on Information Security Applications (WISA), 2002.
- [C43] Y. Kim, F. Maino, M. Narasimha and G. Tsudik, *Secure Group Services for Storage-Area Networks*, IEEE International Workshop on Storage Security, 2002
- [C44] M. El Zarki, S. Mehrotra, G. Tsudik and N. Venkatasubramanian, *Security Issues in a Future Vehicular Network*, European Wireless Symposium, 2002.
- [C45] X. Ding, D. Mazzocchi and G. Tsudik, *Experimenting with Server-Aided Signatures*, Network and Distributed Systems Security Symposium (NDSS), 2002.
- [C46] G. Ateniese, D. Song and G. Tsudik, *Quasi-Efficient Revocation in Group Signatures*, Financial Cryptography and Data Security Conference (FC), 2002.
- [C47] Y. Amir, Y. Kim, C. Nita-Rotaru and G. Tsudik, *On the Performance of Group Key Agreement Protocols (Short Paper)*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2002.
- [C48] G. Tsudik, *Weak Forward Security in Mediated RSA*, Security in Communication Networks (SCN), 2002.
- [C49] X. Ding and G. Tsudik, *Identity-Based Cryptography with Mediated RSA*, RSA Conference: Crypto Track (CT-RSA), 2003.
- [C50] Y. Amir, C. Nita-Rotaru, J. Stanton and G. Tsudik, *Scaling Secure Group Communications: Beyond Peer-to-Peer*, DARPA Information Security Conference and Exposition (DISCEX), 2003.
- [C51] Y. Kim, D. Mazzocchi and G. Tsudik, *Admission Control in Collaborative Groups*, IEEE Symposium on Network Computing and Applications (NCA), 2003.
- [C52] M. Narasimha, G. Tsudik and J. Yi, *On the Utility of Distributed Cryptography in P2P Settings and MANETs*, IEEE International Conference on Network Protocols (ICNP), 2003.
- [C53] G. Tsudik and S. Xu, *Accumulating Composites and Improved Group Signing*, IACR ASIACRYPT, 2003.
- [C54] G. Tsudik, *Admission Control in Peer Groups*, NSF/NASA/DOE Workshop on New Directions on Scalable Cyber-Security, 2003.
- [C55] N. Saxena, G. Tsudik and J. Yi, *Admission Control in Peer-to-Peer: Design and Performance Evaluation*, ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), 2003.
- [C56] N. Saxena, G. Tsudik and J. Yi, *Experimenting with Peer Group Admission Control*, International Workshop on Advanced Developments in Software and Systems Security (WADIS), 2003.
- [C57] N. Saxena, G. Tsudik and J. Yi, *Access Control Mechanisms for Ad Hoc Networks*, IEEE Workshop on Hot Topics in Peer-to-Peer (HOT-P2P), 2004.
- [C58] E. Mykletun, M. Narasimha and G. Tsudik, *Authentication and Integrity in Outsourced Databases*, ISOC Symposium on Network and Distributed Systems Security (NDSS), 2004.
- [C59] X. Ding, G. Tsudik and S. Xu, *Leak-Free Mediated Group Signatures*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2004.
- [C60] S. Mehrotra, E. Mykletun, G. Tsudik and S. Wu, *A Framework for Efficient Storage Security in RDBMS*, International Conference on Extending Database Technology (EDBT), 2004.
- [C61] B. Hore, S. Mehrotra, and G. Tsudik, *A Privacy-Preserving Index for Range Queries*, International Conference on Very Large Databases (VLDB), 2004.
- [C62] C. Castelluccia, S. Jarecki and G. Tsudik, *Secret Handshakes from CA-oblivious Encryption (Short Paper)*, ACM Conference on Principles of Distributed Computing (PODC), 2004.
- [C63] E. Mykletun, M. Narasimha and G. Tsudik, *Signature Bouquets: Immutability of Aggregated Signatures*, European Symposium on Research in Computer Security (ESORICS), 2004.
- [C64] C. Castelluccia, S. Jarecki and G. Tsudik, *Secret Handshakes from CA-oblivious Encryption*, IACR ASIACRYPT, 2004.
- [C65] C. Castelluccia, S. Jarecki and G. Tsudik, *Verifiable and Secure Acknowledgement Aggregation*, Security in Communication Networks (SCN), 2004.
- [C66] N. Saxena, G. Tsudik and J. Yi, *Identity-based Access Control for Ad Hoc Groups*, International Conference on Information Security and Cryptology (ICISC), 2004.
- [C67] C. Castelluccia, E. Mykletun and G. Tsudik, *Efficient Aggregation of Encrypted Data in Wireless Sensor Networks*,

IEEE Mobiquitous, 2005.

- [C68] J. Kim and G. Tsudik, *Securing Route Discovery in DSR*, IEEE Mobiquitous, 2005.
- [C69] G. Tsudik and S. Xu, *A Flexible Framework for Secret Handshakes (Short Paper)*, ACM Conference on Principles of Distributed Computing (PODC), 2005.
- [C70] G. Ateniese, D. Chou, B. de Medeiros and G. Tsudik, *Sanitizable Signatures*, European Symposium on Research in Computer Security (ESORICS), 2005.
- [C71] N. Saxena, G. Tsudik and J. Yi, *Efficient Node Admission for Short-lived Mobile Ad Hoc Networks*, IEEE International Conference on Network Protocols (ICNP), 2005.
- [C72] M. Narasimha and G. Tsudik, *DSAC: Integrity of Outsourced Databases with Signature Aggregation and Chaining (Short Paper)*, ACM Conference on Information and Knowledge Management (CIKM), 2005.
- [C73] E. Mykletun and G. Tsudik, *On using Secure Hardware in Outsourced Databases*, International Workshop on Innovative Architecture for Future Generation High-Performance Processors and Systems (IWIA), 2005.
- [C74-A] S. Nasserian and G. Tsudik, *Revisiting Oblivious Signature-Based Envelopes*, Financial Cryptography (FC), 2006.
- [C74] J. Solis and G. Tsudik, *Simple and Flexible Private Revocation Checking*, Privacy-Enhancing Technologies Symposium (PETS), 2006.
- [C75] G. Tsudik and S. Xu, *A Flexible Framework for Secret Handshakes*, Privacy-Enhancing Technologies Symposium (PETS), 2006.
- [C76] X. Yang, X. Li and G. Tsudik, *A Technical Approach to Net Neutrality*, ACM Workshop on Hot Topics in Networks (Hotnets-V), 2006.
- [C77] G. Tsudik, *YA-TRAP: Yet Another Trivial RFID Authentication Protocol (WiP Paper)*, IEEE Conference on Pervasive Computing (PERCOM), 2006.
- [C78] M. Narasimha and G. Tsudik, *Authentication of Outsourced Databases using Signature Aggregation and Chaining*, International Conference on Database Systems for Advanced Applications (DASFAA), 2006.
- [C79] C. Castelluccia, E. Mykletun and G. Tsudik, *Improving Secure Server Performance by Re-balancing SSL/TLS Handshakes*, ACM Symposium on Information, Computer and Communications Security (ASIACCS), 2006.
- [C80] S. Jarecki, J. Kim and G. Tsudik, *Authentication for Paranoids: Multi-Party Secret Handshakes*, Applied Cryptography and Network Security (ACNS), 2006.
- [C81] E. Mykletun and G. Tsudik *Aggregation Queries in the Database-As-a-Service Model*, IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec), 2006.
- [C82] M. Goodrich, M. Sirivianos, J. Solis, G. Tsudik and E. Uzun, *Loud And Clear: Human Verifiable Authentication Based on Audio*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2006.
- [C83] K. Eldefrawy, M. El Zarki and G. Tsudik, *Incentive-Based Cooperative and Secure Inter-Personal Networking*, ACM Workshop on Mobile Opportunistic Networking (MobiOpp), 2007.
- [C84] M. Narasimha and G. Tsudik, *Privacy-Preserving Revocation Checking*, European PKI Workshop: Theory and Practice (EuroPKI), 2007.
- [C85] C. Soriente, G. Tsudik and E. Uzun, *BEDA: Button-Enabled Device Authentication*, International Workshop on Security for Spontaneous Interaction (IWSSI), 2007.
- [C86] S. Jarecki, J. Kim and G. Tsudik, *Group Secret Handshakes or Affiliation-Hiding Authenticated Group Key Agreement*, RSA Conference, Cryptographers' Track (CT-RSA), 2007.
- [C87] D. Ma and G. Tsudik, *Forward-Secure Aggregation*, IEEE Symposium on Research in Security and Privacy (S&P), 2007.
- [C88] K. Eldefrawy and G. Tsudik, *ALARM: Anonymous Location-Aided Routing in MANETs*, IEEE International Conference on Network Protocols (ICNP), 2007.
- [C89] S. Jarecki, J. Kim and G. Tsudik, *Robust Group Key Agreement using Short Broadcasts*, ACM Computer and Communication Security Conference (CCS), 2007.
- [C90] G. Tsudik, *A Family of Dunces: Trivial RFID Identification and Authentication Protocols*, Privacy Enhancing Technologies Symposium (PETS), 2007.
- [C91] R. di Pietro, L. Mancini, C. Soriente, A. Spognardi and G. Tsudik, *Catch Me (If You Can): Data Survival in Unattended Sensor Networks*, IEEE Conference on Pervasive Computing (PERCOM), 2008.
- [C92] S. Jarecki, J. Kim and G. Tsudik, *Beyond Secret Handshakes: Affiliation-Hiding Authenticated Key Agreement*, RSA Conference, Cryptographers' Track (CT-RSA), 2008.
- [C93] D. Ma and G. Tsudik, *A New Approach to Secure Logging*, IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec), 2008.
- [C94] C. Soriente, G. Tsudik and E. Uzun, *HAPADEP: Human-Assisted Pure Audio Device Pairing*, Information Security Conference (ISC), 2008.
- [C95] G. Ateniese, R. Di Pietro, L. Mancini and G. Tsudik *Scalable and Efficient Provable Data Possession*, Conference on Security and Privacy in Communication Networks, (Securecomm), 2008. **Winner of the 2020 IFIP 10.4 Jean-Claude Laprie Award.**

- [C96] R. Di Pietro, D. Ma, C. Soriente and G. Tsudik *POSH: Proactive co-Operative Self-Healing in Unattended Wireless Sensor Networks*, 2008 IEEE Symposium on Reliable Distributed Systems (SRDS), 2008.
- [C97] K. Eldefrawy and G. Tsudik, *PRISM: Private Routing In Suspicious MANETs*, IEEE International Conference on Network Protocols (ICNP), 2008.
- [C98] D. Ma and G. Tsudik, *DISH: Distributed Self-Healing (in Unattended Sensor Networks)*, International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS), 2008.
- [C99] J. Kim and G. Tsudik, *Survival in the Wild: Robust Group Key Agreement in Wide-Area Networks*, International Conference on Information Security and Cryptology, (ICISC), December 2008.
- [C100] F. Massacci, G. Tsudik and A. Yautsiukhin, *Extended Abstract: Logging Key Assurance Indicators in an Enterprise*, ACM Symposium on Information, Computer & Communication Security (ASIACCS), March 2009.
- [C101] A. Kumar, N. Saxena, G. Tsudik and E. Uzun, *Caveat Emptor: A Comparative Study of Secure Device Pairing Methods*, IEEE Conference on Pervasive Computing (PERCOM), March 2009.
- [C102] R. Di Pietro, C. Soriente, A. Spognardi and G. Tsudik, *Intrusion-Resilience via Collaborative Authentication in Unattended WSNs*, ACM Conference on Wireless Network Security (WiSec), March 2009.
- [C103] K. ElDefrawy, J. Solis and G. Tsudik, *Leveraging Social Contacts for Message Confidentiality in Delay Tolerant Networks*, IEEE Computer Software and Applications Conference (COMPSAC), 2009.
- [C104] E. De Cristofaro, S. Jarecki, J. Kim and G. Tsudik, *Privacy-Preserving Policy-based Information Transfer*, Privacy Enhancing Technologies Symposium (PETS), 2009.
- [C105] A. Kobsa, R. Sonawalla, G. Tsudik, E. Uzun and Y. Wang, *Serial Hook-Ups: A Comparative Usability Study of Secure Device Pairing Methods*, Symposium On Usable Privacy and Security (SOUPS), 2009.
- [C106] E. De Cristofaro, X. Ding and G. Tsudik, *Privacy-Preserving Querying in Sensor Networks*, IEEE Conference on Computer Communications and Networks (ICCCN), 2009.
- [C107] E. De Cristofaro and G. Tsudik, *Practical Private Set Intersection Protocols*, 2008 Financial Cryptography and Data Security Conference (FC), 2010.
- [C108] R. Di Pietro, G. Oligieri, C. Soriente and G. Tsudik, *Intrusion-Resilience in Mobile Unattended WSNs*, IEEE INFOCOM, 2010.
- [C109] C. Catellucia, K. ElDefrawy and G. Tsudik, *Link-Layer Encryption Effect on the Capacity of Network Coding in Wireless Networks*, IEEE INFOCOM WiP Session, 2010.
- [C110] D. Ma and G. Tsudik, *IRRES: Intrusion Resilient Remote Email Storage*, ICDCS Workshop on Security and Privacy in Cloud Computing (SPCC), 2010.
- [C111] M. Manulis, B. Poettering and G. Tsudik, *Affiliation-Hiding Key Exchange with Untrusted Group Authorities*, Applied Cryptography and Network Security (ACNS), 2010.
- [C112] M. Manulis, B. Poettering and G. Tsudik, *Taming Big Brother Ambitions: More Privacy for Secret Handshakes*, Privacy Enhancing Technologies Symposium (PETS), 2010.
- [C113] R. Nithyanand, N. Saxena, G. Tsudik and E. Uzun, *Groupthink: On the Usability of Secure Group Association of Wireless Devices*, ACM International Conference on Ubiquitous Computing (UBICOMP), 2010.
- [C114] R. Nithyanand, G. Tsudik and E. Uzun, *Readers Behaving Badly: Reader Revocation in PKI-Based RFID Systems*, European Symposium on Research in Computer Security (ESORICS), 2010.
- [C115] D. Perito and G. Tsudik, *Secure Code Update for Embedded Devices via Proofs of Secure Erasure*, European Symposium on Research in Computer Security (ESORICS), 2010.
- [C116] Y. Lu and G. Tsudik, *Towards Plugging Privacy Leaks in Domain Name System*, IEEE International Conference on Peer-to-Peer Computing (P2P), 2010.
- [C117] R. Di Pietro, G. Oligieri, C. Soriente and G. Tsudik, *Securing Mobile Unattended WSNs against a Mobile Adversary*, IEEE Symposium on Reliable Distributed Systems (SRDS), 2010.
- [C118] E. De Cristofaro, J. Kim and G. Tsudik, *Linear-Complexity Private Set Intersection Protocols Secure in Malicious Model*, IACR ASIACRYPT, 2010.
- [C119] C. Badea, M. Cesarano, A. Nicolau, G. Tsudik and A. Veidenbaum, *Leveraging Virtualization Towards Improving Cloud Security for Compute-Intensive Applications*, International Conference on Cloud Computing (CloudComp), 2010.
- [C120] G. Ateniese, E. De Cristofaro and G. Tsudik, *(If) Size Matters: Size-Hiding Private Set Intersection*, IACR Public Key Cryptography (PKC), 2011.
- [C121] Y. Lu and G. Tsudik, *Enhancing Data Privacy in the Cloud*, IFIP International Conference on Trust Management (IFIPTM), 2011.
- [C122] K. ElDefrawy, S. Capkun and G. Tsudik, *Group Distance Bounding Protocols*, International Conference on Trust and Trustworthy Computing (TRUST), 2011.
- [C123] E. De Cristofaro, Y. Lu and G. Tsudik, *Efficient Techniques for Privacy-Preserving Sharing of Sensitive Information*, International Conference on Trust and Trustworthy Computing (TRUST), 2011.
- [C124] A. Kobsa, R. Nithyanand, G. Tsudik and E. Uzun, *Usability of Display-Equipped RFID Tags for Security Purposes*, European Symposium on Research in Computer Security (ESORICS), 2011.

- [C125] P. Baldi, R. Baronio, E. De Cristofaro, P. Gasti and G. Tsudik, *Countering GATTACA: Efficient and Secure Testing of Fully-Sequenced Human Genomes*, ACM Computer and Communication Security Conference (CCS), 2011.
- [C126] S. DiBenedetto, P. Gasti, G. Tsudik and Ersin Uzun, *ANDaNA: Anonymous Named Data Networking Application*, ISOC Symposium on Network and Distributed System Security (NDSS), 2012.
- [C127] K. Eldefrawy, A. Francillon, D. Perito and G. Tsudik, *SMART: Secure and Minimal Architecture for (Establishing Dynamic) Root of Trust*, ISOC Symposium on Network and Distributed System Security (NDSS), 2012.
- [C128] E. De Cristofaro, C. Soriente, G. Tsudik and A. Williams, *Hummingbird: Privacy at the time of Twitter*, IEEE Symposium on Research in Security and Privacy (S&P), 2012.
- [C129] E. De Cristofaro and G. Tsudik, *Experimenting with Fast Private Set Intersection*, International Conference on Trust and Trustworthy Computing (TRUST), 2012.
- [C130] Z. Erkin and G. Tsudik, *Private Computation of Spatial and Temporal Power Consumption with Smart Meters*, Applied Cryptography and Network Security (ACNS), 2012.
- [C131] M. Almishari and G. Tsudik, *Exploring Linkability of Community Reviewing*, European Symposium on Research in Computer Security (ESORICS), 2012.
- [C132] E. De Cristofaro, S. Faber, P. Gasti and G. Tsudik, *GenoDroid: Are Privacy-Preserving Genomic Tests Ready for Prime Time?*, ACM Workshop on Privacy in the Electronic Society (WPES), 2012.
- [C133] E. De Cristofaro, P. Gasti and G. Tsudik, *Fast and Private Computation of Cardinality of Set Intersection and Union*, Conference on Cryptology and Network Security (CANS), 2012.
- [C134] J. Burke, P. Gasti, N. Nathan and G. Tsudik, *Securing Instrumented Environments over Content-Centric Networking*, IEEE Workshop on Emerging Design Choices in Name-Oriented Networking (NOMEN), 2013.
- [C135] G. Ács, M. Conti, C. Ghali, P. Gasti and G. Tsudik, *Cache Privacy in Name-Data Networking*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2013.
- [C136] P. Gasti, G. Tsudik, E. Uzun and L. Zhang, *DoS & DDoS in Named Data Networking*, IEEE Conference on Computer Communications and Networks (ICCCN), 2013.
- [C137] M. Almishari, P. Gasti, E. Oguz and G. Tsudik, *Privacy-Preserving Matching of Community-Contributed Content*, European Symposium on Research in Computer Security (ESORICS), 2013.
- [C138] A. Compagno, M. Conti, P. Gasti and G. Tsudik, *Poseidon: Mitigating Interest Flooding DDoS Attacks in Named Data Networking*, IEEE Local Computer Networks (LCN), 2013.
- [C139] E. Ayday, E. De Cristofaro, JP Hubaux and G. Tsudik, *Privacy Considerations of Genome Sequencing*, USENIX Summit on Hot Topics in Security (HotSec), 2013.
- [C140] E. De Cristofaro, S. Faber, and G. Tsudik, *Secure Genomic Testing via Size- and Position-Hiding Private Substring Matching*, ACM Workshop on Privacy in the Electronic Society (WPES), 2013.
- [C141] I. Martinovic, K. Rasmussen, M. Roeschlin and G. Tsudik, *Authentication Using Pulse-Response Biometrics*, ISOC Symposium on Network and Distributed System Security (NDSS), 2014. Winner of **Distinguished Paper Award**.
- [C142] A. Francillon, Q. Nguyen, K. Rasmussen and G. Tsudik, *A Minimalist Approach to Remote Attestation*, IEEE/ACM Design, Automation, and Test in Europe (DATE), 2014.
- [C143] C. Ghali, G. Tsudik and E. Uzun, *Needle in a Haystack: Mitigating Content Poisoning in Named-Data Networking*, NDSS Workshop on Security of Emerging Networking Technologies (SENT), 2014.
- [C144] M. Ambrosin, M. Conti, P. Gasti and G. Tsudik, *Covert Ephemeral Communication in Named Data Networking*, ACM Symposium on Information, Computer and Communications Security (ASIACCS), 2014.
- [C145] J. Burke, P. Gasti, N. Nathan and G. Tsudik, *Secure Sensing over Named Data Networking*, IEEE Symposium on Network Computing and Applications (NCA), 2014.
- [C146] M. Almishari, E. Oguz and G. Tsudik, *Fighting Authorship Linkability with Crowdsourcing*, ACM Conference on Online Social Networks (COSN), 2014.
- [C147] M. Almishari, E. Oguz, D. Kaafar and G. Tsudik, *Stylometric Linkability of Tweets*, ACM Workshop on Privacy in the Electronic Society (WPES), 2014.
- [C148] T. Kaczmarek, A. Kobsa, R. Sy and G. Tsudik, *An Unattended Study of Users Performing Security Critical Tasks Under Adversarial Noise*, NDSS Workshop on Usable Security (USEC), 2015.
- [C149] A. Compagno, M. Conti, P. Gasti, L. Mancini and G. Tsudik, *Violating Consumer Anonymity: Geo-locating Nodes in Named Data Networking*, Applied Cryptography and Network Security (ACNS), 2015. **Best Student Paper Award**.
- [C150] K. ElDefrawy, G. Holland and G. Tsudik, *Remote Attestation of Heterogeneous Cyber-Physical Systems: The Automotive Use Case*, Conference on Embedded Security in Cars (ESCAR), 2015.
- [C151] A. Compagno, M. Conti, C. Ghali, G. Tsudik, *To NACK or not to NACK? Negative Acknowledgments in Information-Centric Networking*, IEEE Conference on Computer Communications and Networks (ICCCN), 2015.
- [C152] S. Faber, R. Petrlic and G. Tsudik, *UnLinked: Private Proximity-based Off-line OSN Interaction*, ACM Workshop on Privacy in the Electronic Society (WPES), 2015.
- [C153] C. Ghali, M. Schlosberg, G. Tsudik and C. Wood, *Interest-Based Access Control for Content Centric Networks*, ACM Conference on Information-Centric Networking (ICN), 2015.

- [C154] C. Ghali, A. Narayanan, D. Oran, G. Tsudik and C. Wood, *Secure Fragmentation for Content-Centric Networks*, IEEE Symposium on Network Computing and Applications (NCA), 2015. **Best Paper Award.**
- [C155] N. Asokan, F. Brasser, A. Ibrahim, A. Sadeghi, M. Schunter, G. Tsudik and C. Wachsmann, *SEDA: Scalable Embedded Device Attestation*, ACM Conference on Computer and Communications Security (CCS), 2015.
- [C156] G. Tsudik, E. Uzun, and C. Wood, *AC3N: Anonymous Communication in Content-Centric Networking*, IEEE Consumer Communications and Networking Conference (CCNC), 2016.
- [C157] G. Tsudik, C. Ghali, C. Wood and E. Yeh, *Practical Accounting in Content-Centric Networking*, IEEE/IFIP Network Operations and Management Symposium (NOMS), 2016.
- [C158] F. Brasser, K. Rasmussen, A. Sadeghi and G. Tsudik, *Remote Attestation for Low-End Embedded Devices: the Prover's Perspective*, ACM Design Automation Conference (DAC), 2016.
- [C159] T. Abera, N. Asokan, L. Davi, F. Koushanfar, A. Paverd, A. Sadeghi and G. Tsudik, *INVITED: Things, Trouble, Trust: On Building Trust in IoT Systems*, ACM Design Automation Conference (DAC), 2016.
- [C160] C. Ghali, G. Tsudik and C. Wood. *BEAD: Best Effort Autonomous Deletion in Content-Centric Networking*, IFIP Networking Conference (NETWORKING), 2016.
- [C161] A. Ibrahim, A. Sadeghi, G. Tsudik and S. Zeitouni, *DARPA: Device Attestation Resilient against Physical Attacks*, ACM Conference on Wireless Network Security (WiSEC), 2016.
- [C162] T. Bradley, S. Faber and G. Tsudik, *Bounded Size-Hiding Private Set Intersection*, Security and Cryptography for Networks (SCN), 2016.
- [C163] G. Tsudik (aka Trotta Gnam), *Zero Knowledge Made Easy, So It Won't Make You Dizzy*, Security and Cryptography for Networks (SCN), 2016.
- [C164] C. Ghali, G. Tsudik and C. Wood, *Network Names in Content-Centric Networking*, ACM Conference on Information-Centric Networking (ICN), 2016.
- [C165] T. Abera, N. Asokan, L. Davi, J. Ekberg T. Nyman, A. Paverd, A. Sadeghi and G. Tsudik, *C-FLAT: Control-Flow ATtestation for Embedded Systems Software*, ACM Conference on Computer and Communications Security (CCS), 2016.
- [C166] M. Almishari, E. Oguz and G. Tsudik, *Trilateral Large-Scale OSN Account Linkability Study*, AAAI Fall Symposium on Privacy and Language Technologies (PLT), 2016.
- [C167] M. Roeschlin, I. Sluganovic, I. Martinovic, G. Tsudik and K. Rasmussen, *Generating Secret Keys from Biometric Body Impedance Measurement*, ACM Workshop on Privacy in the Electronic Society (WPES), 2016.
- [C168] C. Ghali, G. Tsudik and C. Wood, *(The Futility of) Data Privacy in Content-Centric Networking*, ACM Workshop on Privacy in the Electronic Society (WPES), 2016.
- [C169] X. Carpent, K. ElDefrawy, N. Rattanavipanont and G. Tsudik, *Lightweight Swarm Attestation: A Tale of Two LISAs*, ACM Symposium on Information, Computer and Communications Security (ASIACCS), 2017.
- [C170] A. Compagno, M. Conti, D. Lain and G. Tsudik, *Dont Skype & Type! Acoustic Eavesdropping in Voice-Over-IP*, ACM Symposium on Information, Computer and Communications Security (ASIACCS), 2017.
- [C171] M. Conti, G. Lovisotto, I. Martinovic and G. Tsudik, *FADEWICH: Fast Deauthentication over the Wireless Channel*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2017.
- [C172] K. ElDefrawy, N. Rattanavipanont and G. Tsudik, *Fusing Hybrid Remote Attestation with a Formally Verified Microkernel: Lessons Learned*, IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Industry Track, 2017.
- [C173] C. Ghali, G. Tsudik, E. Uzun and C. Wood, *Closing the Floodgates with Stateless Content-Centric Networking*, IEEE Conference on Computer Communications and Networks (ICCCN), 2017. **Best Paper Award.**
- [C174] K. ElDefrawy, N. Rattanavipanont and G. Tsudik, *HYDRA: HYbrid Design for Remote Attestation (Using a Formally Verified Microkernel)*, ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSEC), 2017.
- [C175] B. Berg, T. Kaczmarek, A. Kobsa and G. Tsudik, *Lights, Camera, Action! Exploring Effects of Visual Distractions on Completion of Security Tasks*, Applied Cryptography and Network Security (ACNS), 2017.
- [C176] A. Compagno, M. Conti, D. Lain and G. Tsudik, *Skype & Type: Keystroke Leakage over VoIP*, Black Hat USA, 2017.
- [C177] H. Tan, G. Tsudik and S. Jha, *MTRA: Multiple-Tier Remote Attestations in the Internet of Things*, IEEE Conference on Communications and Network Security (CNS), 2017.
- [C178] I. Nunes-Oliveira, G. Tsudik and C. Wood, *Namespace Tunnels in Content-Centric Networks*, IEEE Conference on Local Computer Networks (LCN), 2017.
- [C179] C. Ghali, G. Tsudik and C. Wood, *Mitigating On-Path Adversaries in Content-Centric Networks*, IEEE Conference on Local Computer Networks (LCN), 2017.
- [C180] S. Mirzamohammadi, J. Chen, A. Amiri Sani, S. Mehrotra, and G. Tsudik, *Ditio: Trustworthy Auditing of Sensor Activities in Mobile & IoT Devices*, ACM Conference on Embedded Networked Sensor Systems (SenSys), 2017.
- [C181] C. Ghali, G. Tsudik and C. Wood, *When Encryption is Not Enough: Privacy Attacks in Content-Centric Networking*, ACM Conference on Information-Centric Networking (ICN), 2017.
- [C182] Z. Zhang, X. Ding, G. Tsudik, J. Cui and Z. Li, *Presence Attestation: The Missing Link In Dynamic Trust Boot-*

- strapping*, ACM Conference on Computer and Communications Security (CCS), 2017.
- [C183] X. Carpent, S. Faber, T. Sander and G. Tsudik, *Private Set Projections & Variants*, ACM Workshop on Privacy in the Electronic Society (WPES), 2017.
- [C184] X. Carpent, N. Rattanavipanon and G. Tsudik, *ERASMUS: Efficient Remote Attestation via Self-Measurements for Unattended Settings*, IEEE/ACM Design, Automation, and Test in Europe (DATE), 2018. **NOTE:** Much shorter, earlier version of J77.
- [C185] X. Carpent, N. Rattanavipanon and G. Tsudik, *Remote Attestation of IoT Devices via SMARM: Shuffled Measurements Against Roving Malware*, IEEE Hardware Oriented Security and Trust (HOST), 2018.
- [C186] K. Eldefrawy, X. Carpent, N. Rattanavipanon and G. Tsudik, *Temporal Consistency of Integrity-Ensuring Computations and Applications to Embedded Systems Security*, ACM Symposium on Information, Computer and Communications Security (ASIACCS), 2018.
- [C187] T. Kaczmarek, E. Ozturk and G. Tsudik, *Assentication: User Deauthentication and Lunchtime Attack Mitigation with Seated Posture Biometric*, Applied Cryptography and Network Security (ACNS), 2018.
- [C188] I. Nunes-Oliveira and G. Tsudik, *KRB-CCN: Lightweight Authentication & Access Control for Private Content-Centric Networks*, Applied Cryptography and Network Security (ACNS), 2018.
- [C189] X. Carpent, K. Eldefrawy, N. Rattanavipanon, A. Sadeghi and G. Tsudik, *INVITED: Reconciling Remote Attestation and Safety-Critical Operation of Simple IoT Devices*, ACM Design Automation Conference (DAC), 2018.
- [C190] K. Balagani, M. Conti, P. Gasti, M. Georgiev, T. Gurtler, D. Lain, C. Miller, K. Molas, N. Samarin, E. Saraci, G. Tsudik and L. Wu, *SILK-TV: Secret Information Leakage from Keystroke Timing Videos*, European Symposium on Research in Computer Security (ESORICS), 2018.
- [C191] P. Gasti and G. Tsudik, *Content-Centric and Named-Data Networking Security: The Good, The Bad and The Rest*, IEEE Symposium on Local and Metropolitan Area Networks (LANMAN), 2018.
- [C192] A. Ibrahim, A. Sadeghi and G. Tsudik, *US-AID: Unattended Scalable Attestation of IoT Devices*, IEEE Symposium on Reliable Distributed Systems (SRDS), 2018.
- [C193] N. Asokan, T. Nyman, N. Rattanavipanon, A. Sadeghi and G. Tsudik, *ASSURED: Architecture for Secure Software Update of Realistic Embedded Devices*, International Conference on Embedded Software (EMSOFT), 2018. **NOTE:** same as J78.
- [C194] T. Kaczmarek, E. Ozturk and G. Tsudik, *Thermanator and Thermal Residue Attacks*, Black Hat Europe, 2018.
- [C195] A. Ibrahim, A. Sadeghi and G. Tsudik, *HEALED: HEaling & Attestation for Low-end Embedded Devices*, Financial Cryptography and Data Security (FC), 2019.
- [C196] T. Kaczmarek, E. Ozturk, P. Tricomi and G. Tsudik, *AcuTherm: A Hybrid Attack on Password Entry Based on Both Acoustic and Thermal Side-Channels*, Black Hat Asia, 2019.
- [C197] K. Eldefrawy and G. Tsudik, *Advancing Secure Remote Attestation via Automated Formal Verification of Designs and Synthesis of Executables*, ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSEC), 2019.
- [C198] I. De Oliveira Nunes, G. Dessouky, A. Ibrahim, N. Rattanavipanon, A. Sadeghi and G. Tsudik, *Towards Systematic Design of Collective Remote Attestation Protocols*, IEEE International Conference on Distributed Computing Systems (ICDCS), 2019.
- [C199] T. Kaczmarek, E. Ozturk and G. Tsudik, **Thermanator**: *Thermal Residue-Based Post Factum Attacks On Keyboard Data Entry*, ACM Symposium on Information, Computer and Communications Security (ASIACCS), 2019.
- [C200] I. De Oliveira Nunes, K. Eldefrawy, N. Rattanavipanon, M. Steiner and G. Tsudik, *VRASED: A Verified Hardware/Software Co-Design for Remote Attestation*, Usenix Security Symposium (Usenix'Sec), 2019.
- [C201] I. De Oliveira Nunes, K. Eldefrawy, N. Rattanavipanon and G. Tsudik, *PURE: Using Verified Remote Attestation to Obtain Proofs of Update, Reset and Erasure in Low-End Embedded Systems*, International Conference On Computer Aided Design (ICCAD), 2019.
- [C202] X. Ding, E. Ozturk and G. Tsudik, *Balancing Security and Privacy in Genomic Range Queries*, Workshop on Privacy in Electronic Society (WPES), 2019.
- [C203] Y. Nakatsuka, A. Paverd and G. Tsudik, *PDOT: Private DNS-over-TLS with TEE Support*, Annual Computer Security Applications Conference (ACSAC), 2019.
- [C204] M. Ammar, B. Crispo and G. Tsudik, *SIMPLE: A Remote Attestation Approach for Resource-Constrained IoT Devices*, International Conference on Cyber-Physical Systems (ICCPS/CPSWeek), 2020.
- [C205] B. Berg, T. Kaczmarek, A. Kobsa and G. Tsudik, *Exploring Effects of Auditory Stimuli on CAPTCHA Performance*, Asia Workshop on Usable Security (AsiaUSEC), 2020.
- [C206] A. Compagno, M. Conti, E. Losiouk, G. Tsudik and S. Valle, *A Proactive Cache Privacy Attack on CCN*, IEEE/IFIP Network Operations and Management Symposium (NOMS) Mini-Conference, 2020.
- [C207] I. De Oliveira Nunes, K. Eldefrawy, N. Rattanavipanon and G. Tsudik, *APEX: A Verified Architecture for Proofs of Execution on Remote Embedded Devices*. Usenix Security Symposium (Usenix'Sec), 2020.
- [C208] M. Conti, P. Tricomi and G. Tsudik, *De-auth of the Blue! Transparent De-authentication using Bluetooth Low Energy*

- Beacon*, European Symposium on Research in Computer Security (ESORICS), 2020.
- [C209] F. Armknecht, P. Walther, G. Tsudik, M. Beck and T. Strufe, *ProMACs: Progressive and Resynchronizing MACs for Continuous Efficient Authentication of Message Streams*, ACM Conference on Computer and Communications Security (CCS), 2020.
- [C210] I. De Oliveira Nunes, S. Jakkamsetti and G. Tsudik, *TINY-CFA: A Minimalistic Approach for Control Flow Attestation Using Verified Proofs of Execution*, IEEE/ACM Design, Automation, and Test in Europe (DATE), 2021.
- [C211] Y. Nakatsuka, E. Ozturk, A. Pavard and G. Tsudik, *CACTI: CAPTCHA Avoidance via Client-side TEE Integration*, Usenix Security Symposium (Usenix'Sec), 2021.
- [C212] X. Ding, I. De Oliveira Nunes and G. Tsudik, *On the Root of Trust Identification Problem*, ACM Conference on Information Processing in Sensor Networks (IPSN), 2021.
- [C213] I. De Oliveira Nunes, S. Jakkamsetti and G. Tsudik, *DIALED: Data Integrity Attestation for Low-end Embedded Devices*, ACM Design Automation Conference (DAC), 2021.
- [C214] I. De Oliveira Nunes, S. Jakkamsetti, N. Rattanavipanon and G. Tsudik, *On the TOCTOU Problem in Remote Attestation*, ACM Conference on Computer and Communications Security (CCS), 2020.
- [C215] M. Ammar, B. Crispo, I. De Oliveira Nunes and G. Tsudik, *Delegated Attestation: Scalable Remote Attestation of Commodity CPS by Blending Proofs of Execution with Software Attestation*, ACM Conference on Wireless Network Security (WiSec), 2021.
- [C216] S. Capkun, E. Ozturk, G. Tsudik and K. Wuest, *ROSEN: RObust and SElective Non-repudiation (for TLS)*, ACM Cloud Computing Security Workshop (CCSW), 2021.
- [C217] M. Cardaioli, M. Conti, P. Tricomi and G. Tsudik, *Privacy-Friendly De-authentication with BLUFADE: Blurred Face Detection* IEEE Conference on Pervasive Computing (PERCOM), 2022.
- [C218] E. Aliaj, I. De Oliveira Nunes and G. Tsudik, *GAROTA: Generalized Active Root-Of-Trust Architecture (for Tiny Embedded Devices)*, Usenix Security Symposium (Usenix'Sec), 2022.
- [C219] L. Petzi, A. Ben Yahya, A. Dmitrienko, T. Prantl, G. Tsudik and S. Kounev, *SCRAPS: Scalable Collective Remote Attestation for Pub-Sub IoT Networks with Untrusted Proxy Verifier*, Usenix Security Symposium (Usenix'Sec), 2022.
- [C220] D. Koisser, P. Jauernig, G. Tsudik and A. Sadeghi, *V'CER: Efficient Certificate Validation in Constrained Networks*, Usenix Security Symposium (Usenix'Sec), 2022.
- [C221] I. De Oliveira Nunes, S. Hwang, S. Jakkamsetti, and G. Tsudik, *Privacy-from-Birth: Protecting Sensed Data from Malicious Sensors with VERSA*, IEEE Symposium on Research in Security and Privacy (S&P), 2022.
- [C222] Y. Liu, Y. Nakatsuka, A. Amiri Sani, S. Agarwal, and G. Tsudik, *VideoProv: Verifiable Provenance for Videos from Mobile Devices*, ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), 2022.
- [C223] K. Balagani, M. Cardaioli, S. Ceconello, M. Conti, and G. Tsudik, *We Can Hear Your PIN Drop: A New Acoustic Side-Channel Attack to Profile PIN Pads Keys*, European Symposium on Research in Computer Security (ESORICS), 2022.

Patents

- [P1] *Authentication Method and System with a Smartcard*, United States Patent 5347580, Issued September 13, 1994.
- [P2] *Personal Key Archive*, United States Patent 5495533, Issued February 27, 1996.
- [P3] *Encryption of Low Data Content ATM Cells*, United States Patent 5642421, Issued June 24, 1997.
- [P4] *Method and System for Providing Secure Authenticated Key Distribution in a Communication System*, European Patent 0711480, Issued June 11, 1997. United States Patent 5729608, Issued March 17, 1998.
- [P5] *System and Method for Changing the Key or Password in a Secure Distributed Communication Network*, European Patent 0720796, Issued July 16, 1997. United States Patent 5778065, Issued July 7, 1998.
- [P6] *Secure Anonymous Information Exchange in a Network*, European Patent 0876722, Issued July 17, 1997. United States Patent 6061789, Issued May 9, 2000.
- [P7] *Method and Apparatus for Secure Identification of a Mobile User in a Communication Network*, European Patent 0788688, Issued May 9, 1996. United States Patent 6072875, Issued June 6, 2000.
- [P8] *RFID Reader Revocation Checking using Low Power Attached Displays*, United States Patent 8710952, Issued April 29, 2014.
- [P9] *Transaction Verification on RFID Enabled Payment and Transaction Instruments*, United States Patent 9,443,240, Issued September 13, 2016.

Miscellaneous Publications

- [M1] *Integrity Protection for the Kerberos Error Messages*, Internet Draft, March 1997.
- [M2] *Public Key Cryptography for Cross-Realm Authentication in Kerberos*, Internet Draft, November 2001.
- [M3] *Internet Keyed Payments Protocol (iKP)*, Internet Draft, July 1995.

- [M4] *Zurich iKP prototype: Protocol Specification Document*. Research Report RZ 2792, IBM Research, February 1996.
- [M5] *Privacy-Oriented Cryptography (Dagstuhl Seminar 12381)*, Dagstuhl Reports 2 (9), (2012)
- [M6] *Genomic Privacy (Dagstuhl Seminar 13412)*, Dagstuhl Reports 3 (10), (2013)
- [M7] *Information-Centric Networking and Security (Dagstuhl Seminar 16251)*, Dagstuhl Reports 6 (6), 2016.
- [M8] *Genomic Privacy II (Dagstuhl Seminar 15431)*, Dagstuhl Reports 5 (10), 2016.

Invited Panelist

- *Networking The Intelligent Edge*, IEEE International Symposium on Local and Metropolitan Area Networks (LAN-MAN), June 2018.
- *Cyber-Crime and Privacy Legislations: Well Together or Growing Apart?*, NICTA/NSW Panel, Sydney (Australia), October 2013.
- *Privacy in the Age of Big Data*, IEEE International Conference on Computer Communications and Networks, August 2012.
- *Evolution of Wireless Security Research*, ACM Conference on Security and Privacy in Wireless and Mobile Networks, April 2012.
- *RFID Security/Privacy: Long-Term Research or Short-Term Tinkering?*, (Chair), ACM Conference on Wireless Network Security, April 2008.
- *Usable Security: Manifest Destiny or Oxymoron?*, 2008 Financial Cryptography and Data Security Conference, January 2008.
- *Securing Critical Software and Hardware*, DoE Workshop on Critical Infrastructure Protection, July 2007.
- *Ethics in Privacy Research*, 2007 Symposium on Privacy-Enhancing Technologies.
- *Security & Privacy in Dynamic Wireless Networks*, (Chair), US-EU Summit on Cybertrust, Dublin, November 2006.
- *Recent Developments and Trends in Trust, Security and Privacy in Ubiquitous Computing*, 2005 IEEE International workshop on Trust, Security and Privacy in Ubiquitous Computing.
- *Security in Ad Hoc and Sensor Networks*, 2005 IEEE Symposium on Research in Security and Privacy.
- *Security: Best Practices for Protecting Your Digital Assets*, 2004 Business 4Site Conference, June 18, 2004.
- *Grand Challenges and Crucial Long-term Research Directions in Security*, 2004 IEEE Symposium on Research in Security and Privacy. May 2004.
- *Research Issues in Resilient Networks*, 2004 UCLA Computer Science Department Research Review, May 2004.
- *Security in Residential Broadband Networks*, 2003 NSF Workshop on Research Priorities in Residential Broadband Networks, October 2003.
- *Trust and Economics*, 2003 NSF Trusted Computing PI meeting, August 2003.
- *Issues in Digital Privacy*, 2001 National Conference for Digital Government Research, May 2001.
- *The Meaning of Survivability*, Workshop on Issues in Theory of Security (WITS'02), January 2002.
- *Security Impact of Distributed Computing Technologies*, IEEE CFSW'98, Rockport, MA, June 1998.
- *Web Privacy and Anonymity*, (Chair) ISOC NDSS'97, San Diego, CA, February 1997.
- *Persistence and Longevity of Digital Signatures*, (Chair) ACM CCS'97, Zürich, Switzerland, April 1997.
- *Limitations of Firewalls as a Network Security Solution*, ESORICS'96, Rome, Italy, September 1996.
- *Emergent OS Security Topics*, DARPA OS Security Workshop, December 1997.

Keynote & Plenary Talks

- *Mitigating TOCTOU in Hybrid Remote Attestation*, 20th International Conference on Cryptology and Network Security (CANS), Vienna, Austria, December 2021.
- *Secure Code Execution on Untrusted Remote Devices*, EUROCRYPT Workshop on Secure Design Strategies For Intelligent CPUs, Zagreb, Croatia, October 2021.
- *Secure Code Execution on Untrusted Remote Devices*, IEEE 6th International Conference on Computer and Communications (ICCC2020), Chengdu, China, December 2020.
- *Thermanator & AcuTherm: Thermal Residue Attacks on Keyboard Password Entry and Hybrid Attacks Combining Acoustic and Thermal Side-Channels*, Great Lakes Security Day 2020, November 2020.
- *Proofs or Remote Execution and Mitigation of TOCTOU Attacks*. ACM Asia Conference on Computer and Communications Security (AsiaCCS), Taipei, Taiwan, October 2020.
- *Reconciling Security and Safety in Simple IoT Devices*, International Conference on Advanced Information Technologies (ICAIT), Yangon, Myanmar, November 2019.
- *VRASED: Verifiable Remote Attestation for Simple Embedded Devices*, International Conference on Science of Cyber Security (SciSec), Nanjing, China, August 2019.
- *Thermanator: Thermal Residue-Based Post Factum Password Attacks*, Data and System Security Workshop, Zhejiang University, Hangzhou, China, August 2019.
- *Reconciling Security and Safety in Simple IoT/CPS Devices*, ACM CCS Workshop on Cyber-Physical System Security (CPS-SPC), Toronto, October 2018.

- *Security in Personal Genomics*, International Workshop on Genome Privacy and Security (GenoPri 2018), Basel, Switzerland, October 2018.
- *Mitigating Tensions between Security and Safety in Low-End Embedded Devices*, European Symposium on Research in Computer Security (ESORICS), Barcelona, Spain, September 2018.
- *Stateless and PIT-Less Content-Centric Networking*, Symposium on Future Internet: Security Issues and Opportunities, Padova, Italy, May 2018.
- *Reconciling Remote Attestation and Safety-Critical Operation of Simple IoT Devices*, ACM DAC, Special Session on Security for IoT Worlds, San Francisco, June 2018.
- *Presence Attestation and Dynamic Trust Bootstrapping*, Workshop on Trusted Computing and its Applications, University of Surrey, UK, January 2018.
- *Security in Personal Genomics*, Tyrrhenian Workshop on Digital Communications, Palermo, Italy, September 2017.
- *Remote Attestation of Low-End Embedded Devices*, PhD School of Advanced Science on Smart Cities, University of So Paulo, Brazil, July 2017.
- *Genomic Security: Lest We Forget*, ACM Symposium on Information, Computer & Communication Security (ASIACCS'17), NYUAD, Abu Dhabi, April 2017.
- *Smart Everything: Dr Jekyll or Mr Hyde?*, Symposium on Mobile Security (MSEC), Seoul, Korea, December 2015.
- *Secure and Private Proximity-Based Discovery of Common Factors in Social Networks*, International Telecommunication Networks and Applications Conference (ITNAC), Sydney, Australia, November 2015.
- *Secure and Private Proximity-Based Discovery of Common Factors in Social Networks*, IEEE International Conference on Cyber Security and Cloud Computing (CSCloud) and Conference on Network and System Security (NSS), NYC, November 2015.
- *Off-line Proximity-Based Privacy-Preserving Social Networking*, International Summer School on Information Security, Bilbao, Spain, July 2015.
- *Challenges in Remote Attestation of Low-End Embedded Devices*, ACM CCS Workshop on Trusted Embedded Devices (TrustED), Tucson, November 2014.
- *Systematic Treatment of Remote Attestation*, ACM DAC, Special Session on Embedded Security, San Francisco, June 2014.
- *Elements of Trust in Content-Centric Networking*, NDSS Workshop on Security of Emerging Networking Technologies (SENT), San Diego, February 2014.
- *Security and Privacy in Named-Data Networking*, 12th International Conference on Cryptology and Network Security (CANS), Paraty, Brasil, November 2013,
- *Privacy in Router-Side Caching: the Case of Named-Data Networking*, IEEE LCN Privacy and Anonymity Workshop (PADE), Sydney, Australia, October 2013.
- *Fragmentation in Next-Generation Internet*, IEEE Conference on Communications and Network Security (CNS), Washington DC, October 2013.
- *Secure and Minimal Architecture for Remote Attestation of Embedded Devices*, IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Melbourne, Australia, July 2013.
- *Security and Privacy Issues in Named-Data Networking*, International Conference on Security of Internet of Things (SecurIT), Kochi, India, August 2012.
- *Security and Privacy Issues in Named-Data Networking*, IEEE Conference on Computer Communications and Networks (ICCCN), Munich, Germany, July 2012.
- *Usable Security: the Case of User-Aided Pairing of Wireless Devices*, Australasian Computer Science Week 2011 (ACSW), Perth, Australia, January 2011.
- *Some Security Topics with Possible Applications for Pairing-Based Cryptography*, International Conference on Pairing-Based Cryptography (PAIRING), Yamanaka, Japan, December 2010.
- *Privacy in Location-Based Mobile (and Vehicular) Ad-Hoc Networks*, Data Privacy Management (DPM) and Autonomous Spontaneous Security (SETOPS) Workshops, Athens, Greece, September 2010.
- *(RFID) Readers Behaving Badly*, Symantec Cutting Edge 2010 Conference, Culver City, February 2010.
- *Readers Behaving Badly: Reader Revocation in PKI-Based RFID Systems*, 4th IEEE International Workshop on Personalized Networks (PerNets 2010), Las Vegas, January 2010.
- *Usable Security: the Case of Device Pairing*, 2009 Conference on Information Security: Practice and Experience (ISPEC'09), Xian, China, April 2009.
- *Security and Privacy in Unattended Sensor Networks*, 2008 Asian Internet Engineering Conference (AINTEC'08), Bangkok, Thailand, November 2008.
- *Privacy Preserving Information Transfer*, 2008 Brazilian Symposium on Information and Computer System Security (SBSeg'08) Gramado, Brasil, September 2008.
- *Sensor Self-Defense: How to Withstand Mobile Adversary in Unattended Sensor Networks*, 2008 Workshop on Information Security: Theory and Practice (WISTP'08), Sevilla, Spain, May 2008.

- *Confronting a Mobile Adversary in Unattended Sensor Networks*, ACM Symposium on Information, Computer & Communication Security (ASIACCS'08), Tokyo, Japan, March 2008.
- *On Privacy in Critical Internet Services*, Second Italian Workshop on Privacy and Security (PRISE'07), Rome, Italy, June 2007.
- *Forward-Secure Aggregate Authentication*, Second Conference on Security in Network Architecture and Information Systems (SAR-SSI'07), Annecy, France, June 2007.
- *Privacy-Preserving Authentication*, Fourth European PKI Workshop: Theory and Practice (EUROPKI'07), Palma de Majorca, Spain, June 2007.
- *A Brief Tour of RFID Security and Privacy*, Reunión Española sobre Criptología y Seguridad de la Información (RECSI'06), Barcelona, Spain. September 2006.
- *Group Key Management: History and Directions*, International Workshop on Secure Multiparty Protocols (SMP'04), Amsterdam, Netherlands, October 2004.
- *Membership Control in P2P Systems*, International Workshop on Hot Issues in Peer-to-Peer Networks (HOT-P2P'04), Volendam, Netherlands, October 2004.
- *Experimenting with Admission Control*, International Workshop on Advanced Developments in Software and Systems Security (WADIS'03), Taipei, Taiwan, December 2003.
- *Security in Outsourced Databases*, Workshop on Information Security Applications (WISA'04), Cheju, Korea, August 2004.
- *Survivability Using Controlled Security Services*, Workshop on Information Security Applications (WISA'01), Seoul, Korea, September 2001.

Invited Talks & Lectures

- *Secure Electronic Commerce*, Bank Of International Settlement, Basel, Switzerland, January 1996.
- *Is There Privacy on the Internet?*, KCET Life and Times Program, Los Angeles, CA, Feb. 11, 1998.
- *Important Issues in Critical Infrastructure Protection*, NSF CIP Workshop, July 1998.
- *Alternative Digital Signatures*, Pacific Region AISS Forum, October 1997.
- *Research Issues in Reliable and Secure Wireless/Mobile Networks*, NSF Invitational Wireless and Mobile Communications Workshop, Airlie, VA, March 1997.
- *Security for Collaborative Groups*, Xerox PARC, November 1998.
- *CLIQUEs: Collaborative Security*, Lucent Bell Labs, August 1999.
- *Collaborative Security and Group Signatures*, UCLA Graduate Colloquium, December 1999.
- *Survivable Security*, Joint US/EU Information Survivability Workshop, Malvern, UK, June 2000.
- *Computer (In)Security in Everyday Life*, UCI Chancellor's Club featured Speaker, May 2000.
- *Controlling Access to Security Services*, Harvey Mudd College, January 2001.
- *Fast Revocation of Credentials*, Johns Hopkins University, October 2001.
- *SUCSES: Survivability Using Controlled Security Services*, University of Maryland, College Park, October 2001.
- *Security in Dynamic Peer Groups*, Swiss Polytechnic Lausanne (EPFL), March 2002.
- *Group Key Agreement*, Turin Polytechnic, February 2002.
- *Fast Credentials Revocation*, Electronics and Telecom Research Institute (ETRI) Korea, September 2001.
- *New Directions in Group Signatures*, Korean Information Security Agency (KISA), September 2001.
- *Secure Group Communication with Spread*, DARPA OPX Pacific Command Meeting, February 2002.
- *Identity-Based Cryptography*, PKNU Busan, Korea, August 2002.
- *Mediated Cryptography*, University of Minnesota, February 2003.
- *Database Privacy in the DAS Model*, Turin Polytechnic, March 2003.
- *Privacy in Outsourced Databases*, IBM Zürich Research Laboratory, September 2003.
- *Authentication and Integrity in Outsourced Databases*, Georgia Tech, November 2003.
- *Admission Control in Dynamic Groups*, Academia Sinica, Taiwan, December 2003.
- *Security in Outsourced Databases*, Purdue University, CERIAS Distinguished Lecture, April 2004.
- *Hacking, Viruses and Related Topics*, UCI University Club Forum, May 2004.
- *Identity Theft on the Internet: Trends and Countermeasures*, UCI Annual CEO Roundtable Retreat, May 2004.
- *Authenticity and Integrity in Outsourced Databases*, Vrije Universiteit, Amsterdam, Netherlands, October 2004.
- *Securing Dynamic Source Routing*, University of Rome "La Sapienza", Rome, Italy, June 2005.
- *Privacy-Preserving Authentication*, Web-Minds Consortium Yearly Meeting, Salerno, Italy, June 2005.
- *Oblivious Information Transfer and Human Assisted Authentication*, Nokia Research Seminar, Helsinki, Finland, October 2005.
- *Secret Handshakes and Private Authentication*, Research Seminar, Helsinki University of Technology, Helsinki, Finland, October 2005.
- *A Flexible Framework for Secret Handshakes*, Dagstuhl Seminar: *Anonymous Communication and its Applications*,

- Dagstuhl, Germany, October 2005.
- *Secret Handshakes, or Privacy-Preserving Interactive Authentication*, TU Hamburg-Harburg Security Seminar, Hamburg, Germany, October 2005.
 - *Emerging Threats and Challenges in Security Research*, INRIA Seminar of the Scientific and Technical Orientation Council, Paris, France, March 2006.
 - *Trivial RFID Authentication and Identification Protocols*, INRIA Guest Seminar, Grenoble, France, April 2006.
 - *Some Thoughts on the Future of Internet Security*, National Academies Meeting, Irvine, April 2006.
 - *Secure Device Pairing*, International Chair in Information Security Distinguished Lecture, Sophia Antipolis, France, December 2006.
 - *Secure First Connect: Ubiquitous Device Pairing*, UCLA Center for Embedded Networked Sensing (CENS) Seminar, January 2007.
 - *Anonymous Mobility in Suspicious MANETs*, EPFL (Swiss Federal Polytechnic Institute Lausanne) Summer Research Institute, July 2007.
 - *Security Issues in Unattended Sensor Networks*, Zürich Information Security Workshop, September 2007.
 - *Privacy in Suspicious MANETs*, Fulbright Lecture, University of Trento, August 2007.
 - *Forward-Secure Aggregate Authentication*, Fulbright Lecture, University of Trento, August 2007.
 - *Challenges in RFID Security and Privacy*, Fulbright Lecture, University of Malaga, September 2007.
 - *Forward-Secure Aggregate Authentication*, Fulbright Lecture, University of Malaga, September 2007.
 - *Robust Group Key Agreement with Short Broadcasts*, New York Polytechnic, November 2007.
 - *Data Survival in Unattended WSNs*, IBM TJ Watson Research Center, November 2007.
 - *Anonymous Mobility in Suspicious MANETs*, Stony Brook University (SUNYSB), November 2007.
 - *Coping with a Mobile Adversary in Unattended Sensor Networks*, Dagstuhl Seminar: *Security Hardware in Theory and Practice*, Dagstuhl, Germany, June 2008.
 - *Anonymous Mobility in Suspicious MANETs*, Universitat Rovira i Virgili (Tarragona, Spain), June 2008.
 - *Anonymous Mobility in Suspicious MANETs*, University of Calgary, (Canada), October 2008. June 2008.
 - *Usable Security: the Case of Device Pairing*, CRYISIS Seminar, Budapest University of Technology and Economics. Budapest, Hungary, July 2009.
 - *Overview of Security & Privacy Research in SPROUT*, Symantec Research Labs, Culver City, CA, June 2009 .
 - *Privacy-Preserving Sharing of Sensitive Information*, JHU Applied Physics Laboratory, September 2009.
 - *Overview of Modern Cryptography*, Universidad de La Laguna, Tenerife, Spain. January 2010.
 - *Sharing Sensitive Information with Privacy*, Johns Hopkins University, April 2010.
 - *Secure Code Update and Attestation for Embedded Devices*, Katholieke Universiteit Leuven, Belgium, May 2010.
 - *(If) Size Matters: Size-Hiding Private Set Intersection*, Tel Aviv University, Israel, May 2010.
 - *Privacy-Preserving Querying in Wireless Sensor Networks*, Delft University of Technology, Netherlands, May 2010.
 - *Private Set Intersection and Friends*, Mathematics Dept., UCI, October 2010.
 - *Secure Manipulation of Embedded Devices*, University of Michigan, Dearborn, November 2010.
 - *Selected Research Topics in Wireless Security*, Nokia Research Center - Helsinki, Finland, December 2010.
 - *Efficient Private Set Operation*, University of Maryland - College Park, January 2011.
 - *Sharing Sensitive Information with Privacy*, Macquarie University, Sydney, Australia, January 2011.
 - *Usable Security: The Case of Pairing Wireless Personal Devices*, COR WIT Inaugural Lecture, Technical University of Delft, The Netherlands, February 2011.
 - *Readers Behaving Badly (or when Malice meets Fob)*, Security Seminar, Radboud University Nijmegen, The Netherlands, February 2011.
 - *Size-Hiding Private Set Intersection*, Delft University of Technology, Netherlands, September 2011.
 - *SMART: Secure and Minimal Architecture for Establishing Dynamic Root of Trust*, Notre Dame University, April 2012.
 - *Hummingbird: Privacy at the Time of Twitter*, Amrita University, India-US Workshop on Big Data, August 2012.
 - *Limits of Privacy in On-line Micro-Blogging*, University of New South Wales (UNSW), November 2012.
 - *Hummingbird: Privacy at the Time of Twitter*, CISCO Corporate Seminar, January 2013.
 - *Privacy in Micro-Blogging OSNs*, Arizona State University, April 2013.
 - *Security and Privacy in Named-Data Networking*, Astor Fellowship Lecture, Oxford University, May 2013.
 - *Privacy in Router-Side Caching: the Case of Named-Data Networking*, Berkeley Inria Stanford (BIS) Workshop, May 2013.
 - *Overview of NDN Security and Privacy Landscape*, Zurich Information Security and Privacy Center (ZISC) Workshop on "Securing Future Communication Networks against Emerging Threats," October 2013.
 - *Elements of Trust in Content-Centric Networking*, UCI Computer Science Seminar Series, April 2014.
 - *User Authentication using Pulse-Response Biometric*, CSE Department Seminar, Aalto University, May 2014.
 - *Trust, Security and Privacy in Content-Centric Networks*, Swedish Royal Institute (KTH), June 2014.
 - *Elements of Trust in Named-Data Networking*, University College London, CS Department Seminar, July 2014.

- *Systematic Design of Minimalist Remote Attestation for Low-End Embedded Systems*, EPFL Summer Research Institute, June 2014.
- *Remote Attestation for Low-End Embedded Devices*, Hughes Research Labs (HRL), August 2014.
- *Usability and Trust Challenges in Cyber-Physical and Embedded Systems*, National Academies, Board on Human-System Interaction Meeting, September 2014.
- *Challenges in Remote Attestation of Embedded Devices*, Distinguished Lecture Series at Technical University Darmstadt, December 2014.
- *Overview of Research Activities in SPROUT*, UCI Tech Open House, February 2015.
- *Elements of Trust in Content-Centric and Named-Data Networking*, Distinguished Lecture Series at University of Calgary, March 2015.
- *Discrete and Continuous Authentication via Pulse-Response Biometric*, CS Department Seminar, Saarland University, April 2015.
- *Private Off-Line Social Networking*, Aalto University, April 2015.
- *Private Off-Line Social Network Interactions*, Technical University Darmstadt (CROSSING), June 2015.
- *Security Issues in Low-End Embedded Devices*, University of Padova, June 2015.
- *Fighting Authorship Linkability with Crowdsourcing*, University of Padova, June 2015.
- *Scalable Embedded Device Attestation*, National ICT Australia (NICTA), November 2015.
- *Authentication with Pulse-Response Biometric*, Korea Advanced Institute of Science and Technology (KAIST), December 2015.
- *Security and Privacy in Content-Centric (and Information-Centric) Networking*, National Security Research (NSR), Korea, December 2015.
- *Off-line Private Social Network Interactions*, Kookmin University, Korea, December 2015.
- *Remote Attestation of Embedded Devices*, Samsung Advanced Training Seminar, Korea, December 2015.
- *Attestation of IoT Devices*, Singapore Management University, August 2016.
- *Security and Privacy in Content-Centric Networking*, USC Informations Sciences Institute, January 2017.
- *Security in Personal Genomics*, Vanderbilt University, Computer Science Seminar, April 2017.
- *Presence Attestation*, IoT Security Worskhop, Aalto University, Finland, September 2017.
- *Genomic Security*, Computer Science Seminar, University of New South Wales, Australia, October 2017.
- *Presence Attestation of IoT Devices*, Computer Science Seminar, Macquarie University, Australia, October 2017.
- *Genomic Security and Privacy*, UCI LPS/Philos 144: Topics in Philosophy of Social Science: Privacy, December 2017.
- *Security & Privacy in Content-Centric Networking*, Distinguished Lecture, Technical University of Dresden, Germany, December 2017.
- *Security Challenges in Personal Genomics*, UCLA Institute for Pure and Applied Mathematics (IPAM), January 2018.
- *ICN/CCN Security & Privacy Landscape*, UC Davis Computer Science Department Seminar, February 2018.
- *Establishing Initial Trust in Personal Devices*, Security Seminar, Technical University of Dresden, Germany, March 2017.
- *Security in Content-Centric and Named-Data Networking : The Good, The Bad and The Rest* CSE Seminar, University at Buffalo, October 2018.
- *Coping with Security / Safety Tensions in Low-End Embedded Devices*, UCSD HDSI Mini-Symposium on Security and Privacy at the Edge, February 2019.
- *IoT Security Challenges*, UCSD CYSMICS Picks Workshop, February 2019.
- *VRASED: Verifiable Remote Attestation for Simple Embedded Devices*, Distinguished Lecture Series at Technical University Darmstadt, March 2019.
- *Security Issues for Simple Embedded/IoT Devices*, UC Riverside ECE Department Distinguished Colloquium, April 2019.
- *Reconciling Security and Real-Time Constraints for Simple IoT Devices*, University College London (UCL), InfoSec Seminar, March 2020.
- *Secure Code Execution on Untrusted Remote Devices*, IEEE COMSOC SecIG Special Interest Group, April 2021.
- *CACTI: Captcha Avoidance via Client-side TEE Integration*, UNSW Institute for Cyber Security (IFCYBER) Seminar, August 2021.
- *CACTI: Captcha Avoidance via Client-side TEE Integration*, ETH Zurich Computer Science Distinguished Colloquium, October 2021.
- *Secure Code Execution on Untrusted Remote Devices*, Zurich Information Security Group (ZISC) Seminar, December 2021.
- *CACTI: Captcha Avoidance via Client-side TEE Integration*, University of Colorado (Colorado Springs), Computer Science Distinguished Talk, March 2022.

Scientific & External Advisory Board Membership

- *CloSe Project*, Academy of Finland, 2015-2017.
- *ReCRED Project*, European Union, H2020 Program, 2015-2017.
- *UCN@Sophia Laboratoire d'Excellence (LABEX) Project*, French Ministry of Research, 2016-2018.
- *GetPreCiSe: NIH Center for Genetic Privacy and Identity in Community Settings*, Vanderbilt University, 2016-2018.
- *CYBERCAT: Center for Cybersecurity Research of Catalonia*, Tarragona, Spain, 2018-present.
- *The Cyber Security Body of Knowledge (CYBOK)*, Official Chapter Reviewer, UK, 2019-2021.
- *INTEL Emerging Technology Project*, Invited Contributor, 2021.

Awards & Honors

- [A1] **UCI School of Information and Computer Sciences Dean's Research Award**, June 2020.
- [A2] **IFIP Jean-Claude Laprie Award in Dependable Computing**, May 2020.
- [A3] **IFIP Fellow**, "For Contributions to Internet Security and Privacy", November 2019.
- [A4] **Fulbright Specialist**, 2014-2019 Specialist Pool:
 - Singapore Management University, Singapore, July-August 2016, June-July 2017.
 - National Cheng Kung University, Taiwan, November 2016.
 - University of Information Technology, Yangon, Myanmar, October-November 2019.
- [A5] **Distinguished Professor**, UC Irvine, July 2019.
- [A6] **TÜV Süd Foundation Visiting Professorship**, TU Dresden (Germany), 2017-2018.
- [A7] **2017 ACM SIGSAC Outstanding Contributions Award**, ACM Special Interest Group on Security, Audit and Control (SIGSAC), 2017.
- [A8] **Best Paper Award**, IEEE International Conference on Computer Communications and Networks (ICCCN), 2017.
- [A9] **Technical Highlights Article**: "Authentication Using Pulse-Response Biometrics", Communications of the ACM (CACM), February 2017.
- [A10] **AAAS Fellow**, "For Contributions to Internet Security and Privacy", November 2016.
- [A11] **Distinguished Visiting Professor**, University of Padova, Italy, 2016.
- [A12] **Foreign Member of Academia Europea**, Informatics Section, September 2015.
- [A13] **Best Paper Award**, IEEE Network Computing and Applications (NCA), 2015.
- [A14] **Best Student Paper Award**, Applied Cryptography and Network Security (ACNS), 2015.
- [A15] **PoC and PI**, UCIs designation as a National Center of Academic Excellence in Cyber Defense Research (CAE-R) 2015-2019, by the National Security Agency (NSA) and the Department of Homeland Security (DHS).
- [A16] **ACM Fellow**, "For Contributions to Internet Security and Privacy", November 2014.
- [A17] **Distinguished Paper Award**, ISOC Symposium on Network and Distributed System Security (NDSS), 2014.
- [A18] **UCI School of Information and Computer Sciences Dean's Award for Graduate Student Mentoring**, January 2014.
- [A19] **Chancellor's Professor**, UC Irvine, July 2013 (re-appointed in 2018).
- [A20] **IEEE Fellow**, Communications Society, "For Contributions to Distributed Systems Security and Privacy", Class of 2013.
- [A21] **Astor Visiting Fellow**, Oxford University, UK, May 2013.
- [A22] **Steering Committee Chair**, WiSec: ACM Conference on Wireless Network Security, 2011-2015.
- [A23] **Associate Editor** for Security and Privacy, 2010-11 ACM New Computing Classification System (CCS) Project, November 2010.
- [A24] **Cor Wit Chair** at Delft University of Technology, The Netherlands, September 2009.
- [A25] **Editor-in-Chief**, ACM Transactions on Information and Systems Security (TISSEC), 2009-2015.
 - Initial appointment, December 2008.
 - Re-appointed for a second term, August 2011.
- [A26] **Spotlight Paper**, IEEE Transactions of Mobile Computing (TMC), "ALARM: Anonymous Location-Aided Routing in Suspicious MANETs," September 2011.
- [A27] **Fulbright Senior Lecturer/Researcher**, University of Rome, "La Sapienza", Italy, April – July 2007.
 - Fulbright Inter-Country Award, University of Malaga, Spain, September 2007.
 - Fulbright Inter-Country Award, University of Trento, Italy, August 2007.
- [A28] DARPA Program Award for Excellence in Academic Research, January 2002.
- [A29] IBM 2nd Plateau Invention Achievement Award, February 1996.
- [A30] IBM Outstanding Innovation Award, September 1994.
- [A31] IBM 1st Plateau Invention Achievement Award, June 1994.
- [A32] IBM Initial Patent Application Award, August 1992.

Funding History

- [G1] PI: *CLIQUEs: Security Services for Groups Communication*, DARPA ITO, 940K, 06/97-12/00.
- [G2] PI: *Instrumentation Support for Secure Group Communication*, IBM, 85K, 07/98, unrestricted.
- [G3] PI: *Server-Assisted Signatures*, NSA, 199K, 07/98-07/00.
- [G4] co-PI: *Integrated Multicast for Ad Hoc Networks*, NSF CCR, 720K, 11/98-11/01.
- [G5] co-PI: *Flexible One-Time Signatures*, NSA, 184K, 07/98-07/00.
- [G6] co-PI: *New Multicast Techniques in Ad Hoc Networks*, Hughes HRL, 35K, 11/97, unrestricted.

- [G7] PI: *Secure Internet Multicast*, Bay Networks/Nortel, 100K, 06/99, unrestricted.
- [G8] PI: *SUCSES: Survivability Using Controlled Security Services*, DARPA ITO, 1,289K, 07/99-06/02.
- [G9] PI: *TOCLAK: Reliable and Secure Communication for Collaborative Applications*, DoE, 560K, 08/99-07/01.
- [G10] PI: *Rainbow Technologies Faculty Initiation Award*, Rainbow Technologies, 100K, 01/00, unrestricted.
- [G11] co-PI: *NSF Computer & Communications Research PI Workshop*, NSF CCR, 200K, 08/00-08/01.
- [G12] PI: *High Performance Robust and Secure Group Communication for Dynamic Coalitions*, subcontract from Johns Hopkins University under DARPA ITO contract, 360K, 05/00-04/03.
- [G13] PI: *Mediated Security Services*, subcontract from USC/ISI under DARPA ITO, 528K, 06/00-06/02.
- [G14] PI: *Secure Group Communication for Storage-Area Networks*, CISCO, 60K, 12/01, unrestricted.
- [G15] co-PI: *Privacy in Database-as-a-Service Model*, NSF ITR, 430K, 09/02-08/05.
- [G16] PI: *IPv6 Ubiquitous Networking Security*, SUN Labs, 80K, 08/03, unrestricted.
- [G17] PI: *IPv6 Ubiquitous Networking Security*, University of California (UC) MICRO, 54K, 08/03-12/04.
- [G18] Senior Personnel: *RESCUE: Responding to Crises and Unexpected Events*, NSF Large ITR, 12.5M, 09/03-08/08.
- [G19] PI: *Membership Control in P2P and MANETs*, Army Research Office (ARO), 300K, 08/04-07/07.
- [G20] PI: *Key Management in Sensor Networks and MANETs*, INRIA (FRANCE) Collaborative Research Award, 40K EUR, 01/05-12/07.
- [G21] PI: *Secure Hardware in Outsourced Databases*, NSF ITR Supplement, 66K, 09/05-08/07.
- [G22] co-PI: *Towards Privacy-Preserving Internet Name Service*, UCI CORCL, 8K, 06/07-06/08.
- [G23] PI: *UCLA IPAM Cryptography Program Participation*, UCLA sub-contract, 15K, 09/06-12/06.
- [G24] PI: *Supplement: Privacy in Tactical MANETs*, Army Research Office (ARO), 100K, 08/07-12/08.
- [G25] PI: *Phd Fellowships in Cybertrust: Educating Security and Privacy Experts for the Nation*, Department of Education (DoEDU), approx. 500K, 08/07-08/10.
- [G26] PI: *Defending Electronic Frontiers: PhD fellowships in Information Assurance*, NSF Scholarship for Service (SFS), 1,134K, 08/07-08/14.
- [G27] PI: *PhD Fellowship in Electronic Privacy*, IBM Research, 43K, 08/07-08/08.
- [G28] PI: *ARO Cyber-Physical Security Workshop*, Army Research Office, 30K, 04/08-09/08.
- [G29] PI: *Policy-Based Privacy-Preserving Information Transfer*, IARPA, 430K, 01/09-07/10.
- [G30] PI: *User-Aided Secure Association of Wireless Devices*, NSF Cybertrust, 460K (250K – UCI), 09/08-09/11.
- [G31] PI: *Secure and Usable Group Association of Personal Wireless Devices*, Google Research Award, 50K, 12/09, unrestricted.
- [G32] co-PI: *EAGER: Usable Location Privacy in Geo-Social Networks*, NSF IIS, 300K, 09/09-08/11.
- [G33] PI: *COR WIT Professorship at TU Delft*, funded by KPN - Dutch Telecom, 50K EUR, 09/09-03/11.
- [G34] PI: *FIA: Named Data Networking (NDN)*, NSF CISE, 600K, 09/10-09/14.
- [G35] co-PI: *ESPADA: Efficient Security and Privacy Assurance for Database Access*, IARPA, 740K (subc. from IBM), 09/11-03/15.
- [G36] PI: *Denial-of-Service Attack Mitigation in Information-Centric Networks*, CISCO, 95K, 07/12, unrestricted.
- [G37] PI: *Effects of Distractions on Usable Security*, NSF Cybertrust Supplement, 50K, 09/12-08/13.
- [G38] co-PI: *CSR: Enabling Privacy-Utility Trade-Offs in Pervasive Computing Systems*, NSF CISE, 142K, 10/12-10/15.
- [G39] PI: *Privacy Techniques for Embedded Systems*, Robert Bosch, Inc., 50K, 11/12, unrestricted.
- [G40] PI: *Further Exploration of Security and Privacy in Named-Data Networking*, NSF FIA Supplement, 103K, 08/13-09/14.
- [G41] PI: *Privacy in Internet Domain Name Service*, Verisign, Inc., 50K, 04/14, unrestricted.
- [G42] PI: *Unattended Studies of Effects of Audio & Visual Distractions on Users Performing Security-Critical Tasks*, UCI CORCL, 6K, 02/15-12/15.
- [G43] co-PI: *EAGER: Unattended/Automated Studies of Effects of Auditory Distractions on Users Performing Security-Critical Tasks*, NSF IIS, 240K, 09/15-08/18.
- [G44] PI: *ERADS: Efficient Remote Attestation of Dynamic Swarms*, NSA, 286K, 09/15-12/16.
- [G45] PI: *Side-Channel Causal Analysis for Design of Cyber-Physical Security*, DHS (subc. from HRL), 300K, 10/15-09/18.
- [G46] PI: *Remote Attestation of Critical Infrastructure Components*, Army Research Office (ARO), 407K, 08/16-07/20.
- [G47] PI: *WiFiUS: Collaborative Research: SELIOT: Securing Lifecycle of Internet-of-Things*, NSF CNS, 300K (150K UCI), 04/17-03/19.
- [G48] PI: *TÜV Süd Visiting Professorship Grant at TU Dresden (Germany)*, EUR 25K, 2017-2018.
- [G49] PI: *Addressing Mobility/Caching and Security/Privacy Challenges in Wireless/Mobile Edge Content-Centric Networks*, CISCO, 180K (90K UCI), 01/18, unrestricted.
- [G50] PI: *Genomic Security Issues*, UCI CORCL, 3K, 02/18-12/18.
- [G51] co-PI: *Horizon: Secure Large-Scale Scientific Cloud Computing*, 1,000K, NSF CICI, 09/18-08/21.
- [G52] PI: *PRISM: PRivacy-preserving Intrusion-resilient Secure Multiparty Computation (for Anonymous Messaging)*, 840K, SRI International (DARPA subcontract from SRI International), 06/19-05/22.

- [G53] co-PI: *SP-DAMIT: Security, Privacy and Data Analytics for Mobile and IoT Devices*, 150K, UCI Office of Vice-Chancellor for Research, 03/19-02/20.
- [G54] PI: *Formally Verified Hardware/Software Co-Design for Remote Attestation of IoT Devices*, 270K, Semiconductor Research Corporation (SRC), 10/19-09/22.
- [G55] PI: *Separate add-on to [G46]*, Army Research Office (ARO), 50K, 04/20-10/20.
- [G56] PI: *Secure 5G Design and Deployment*, DARPA Subcontract from Perspecta Labs, 996K, 10/20-09/23.
- [G57] PI: *Manipulation Secure Image Sensing for Smartphones*, 300K, NSA, 09/20-08/22.
- [G58] co-PI: *Frontiers: ProperData: Protecting Personal Data Flow on the Internet*, 10M, NSF, 10/20-09/25.
- [G59] PI: *Future Security and Privacy Directions for Low-End IoT/Embedded Devices*, 5K, INTEL, 03/21.
- [G60] PI: *Proactive TCBs for Low-End IoT/Embedded Devices*, 10K, ICS@UCI, 04/21-07/22.

Professional Service

- **Steering Groups:**
 - ACM Conference on Wireless Network Security (WiSEC): **Chair:** 2011-2015
 - ISOC Network and Distributed Systems Security Symposium (NDSS): 1999-2003
 - IEEE Conference on Security & Privacy in Communication Networks (SECURECOMM): 2005-2007
 - Security and Cryptography for Networks (SCN): 2006-2021
 - ACM Symposium on Computer and Communications Security (CCS): 2007-2011
 - IEEE Network Protocols Security Workshop (NPsec): 2007-2013
 - ACM Workshop on Scalable Trusted Computing (STC): 2008-2012
 - ACM Workshop on Secure Cloud Computing (CCSW): 2009-2015, **Chair:** 2015-2017.
 - Workshop on Automotive and Autonomous Vehicle Security (AutoSec): 2021-now.
- **Editor-in-Chief:**
 - ACM Transactions on Information and Systems Security (TISSEC, renamed ACM TOPS in 2016), 2009-2015.
- **Special Issue Guest Editor:**
 - ACM Transactions on Information and Systems Security, August 2000.
 - ACM Transactions on Information and Systems Security, May 2001.
 - ACM Transactions on Information and Systems Security, April 2008.
 - IEEE Security & Privacy Magazine, April 2010.
- **Editorial Board Member / Associate Editor:**
 - ACM Transaction on Information and Systems Security (TISSEC), Nov. 2000 – Dec. 2008.
 - ACM/Kluwer Wireless Networks (WINET), Aug. 2005 – Sept. 2007.
 - Computer Communications (Elsevier), Nov. 2008 – Dec. 2013.
 - Pervasive and Mobile Computing (Elsevier), Dec. 2009 – Dec. 2013.
 - IEEE Security and Privacy Magazine, Nov. 2016 – May 2017.
 - PoPETs: Proceedings on Privacy Enhancing Technologies (De Gruyter), Feb 2017 – Jan 2018.
- **Associate Editor-in-Chief:** IEEE Security & Privacy, May 2017-April 2018.
- **Chair,** ACM SIGSAC Awards Committee, 2019.
- **Member,** ACM SIGSAC Awards Committee, 2018.
- **Member,** MacArthur Fellows Nominating Committee, 2015.
- **Member,** ACM SIGSAC Dissertation Award Committee, 2014.
- **Member,** Award for Outstanding Research in Privacy Enhancing Technologies Committee, (PET Award), 2011.
- **Technical Program Chair & Co-Chair:**
 - 1999 ACM Conference on Computer and Communications Security (CCS).
 - 1999 ISOC Network and Distributed Systems Security Symposium (NDSS).
 - 2000 ISOC Network and Distributed Systems Security Symposium (NDSS).
 - 2005 European Workshop on Security in Ad Hoc and Sensor Networks (ESAS).
 - 2005 IEEE Conference on Security & Privacy for Emerging Areas in Communication (SECURECOM).
 - 2006 IEEE Conference on Security & Privacy for Emerging Areas in Communication (SECURECOM).
 - 2006 IEEE Workshop on Secure Network Protocols (NPsec).
 - 2006 ACM Workshop on Scalable Trusted Computing (STC).
 - 2008 Financial Cryptography and Data Security Conference (FC).
 - 2008 ARO Workshop on Security & Privacy in Cyber-Physical Systems.
 - 2009 IACR Conference on Theory and Practice of Public Key Cryptography (PKC).
 - 2010 Information Security Conference (ISC).
 - 2010 IEEE TCCC Computer Communications Workshop (CCW).
 - 2011 Applied Cryptography and Network Security (ACNS).
 - 2011 ACM Conference on Wireless Network Security (WiSec).

- 2011 Workshop on Data Security and Privacy in wireless Networks (D-SPAN).
- 2011 Conference on Cryptology and Network Security (CANS).
- 2012 Dagstuhl Seminar: *Privacy-Oriented Cryptography*.
- 2013 Dagstuhl Seminar: *Genomic Privacy*.
- 2015 Dagstuhl Seminar: *Genomic Privacy, Part II*.
- 2015 NDSS Workshop on Security of Emerging Networking Technologies (SENT).
- 2016 Dagstuhl Seminar: *Information-centric Networking and Security*.
- 2023 ACM Asia Conference on Computer and Communications Security (AsiaCCS).
- General co-Chair: 2010 Workshop on Security and Social Networking (SESOC).
- Publicity Chair: 1998 ACM Conference on Computer and Communications Security (ACM-CCS).
- Track Co-Chair: Security, Privacy and Content Protection Track, 2016 IEEE Consumer Communications & Networking Conference (CCNC).
- Program Committee Member:
 - 1994 IEEE Symposium on Research in Security and Privacy.
 - 1995 IEEE Computer Security Foundations Workshop.
 - 1996 IEEE Computer Security Foundations Workshop.
 - 1997 IEEE International Conference on Distributed Computing Systems.
 - 1997 ACM Conference on Computer and Communications Security.
 - 1998 ACM Conference on Computer and Communications Security.
 - 1997 ISOC Network and Distributed Systems Security Symposium.
 - 1998 ISOC Network and Distributed Systems Security Symposium.
 - 1999 Security in Computer Networks Conference.
 - 2000 ACM Conference on Electronic Commerce.
 - 2001 IEEE International Conference on Network Protocols (ICNP'01)
 - 2002 IEEE International Conference on Network Protocols (ICNP'02)
 - 2002 European Symposium on Research in Computer Security (ESORICS'02)
 - 2002 ACM Conference on Computer and Communications Security (CCS'02)
 - 2002 Security in Computer Networks Conference (SCN'02)
 - 2003 RSA Conference, Crypto Track (RSA-CT)
 - 2003 Financial Cryptography Conference (FC'03)
 - 2003 ACM Conference on Computer and Communications Security (CCS'03).
 - 2003 IFIP Conference on Communications and Multimedia Security (CMS'03)
 - 2003 Applied Cryptography and Network Security (ACNS'03)
 - 2004 IEEE Symposium on Research in Security and Privacy (S&P'04)
 - 2004 ISOC Network and Distributed Systems Security Symposium (NDSS'04)
 - 2004 European Workshop on Security in Ad-Hoc and Sensor Networks (SASN'04)
 - 2004 ACM Conference on Computer and Communications Security (CCS'04)
 - 2004 Security in Computer Networks Conference (SCN'04)
 - 2004 ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'04)
 - 2005 Financial Cryptography Conference (FC'05)
 - 2005 IEEE International Conference on Distributed Computing Systems (ICDCS'05)
 - 2005 IEEE Workshop on Hot Topics in Peer-to-Peer Systems (HOT-P2P'05)
 - 2005 International Workshop on Trust, Security and Privacy for Ubiquitous Computing (TSPUC'05)
 - 2006 Financial Cryptography Conference (FC'06)
 - 2006 IEEE Workshop on Hot Topics in Peer-to-Peer Systems (HOT-P2P'06).
 - 2006 International Workshop on Trust, Security and Privacy for Ubiquitous Computing (TSPUC'06).
 - 2006 Security in Computer Networks Conference.
 - 2006 IEEE Conference on Parallel and Distributed Systems (ICPADS'06)
 - 2007 ACM Symposium on Information, Computer and Communications Security (ASIACCS'07).
 - 2007 ACM Mobisys Healthnets Workshop
 - 2008 IEEE Symposium on Research in Security and Privacy (S&P'08).
 - 2008 Symposium on Privacy-Enhancing Technologies (PET'08)
 - 2008 European PKI Workshop (EuroPKI'08)
 - 2008 IEEE Workshop on Mission-Critical Networking (MCN'08)
 - 2008 IEEE International Conference on Network Protocols (ICNP'08)
 - 2009 ACM Symposium on Information, Computer and Communications Security (AsiaCCS'09)
 - 2009 IFIP International Information Security Conference (IFIP-SEC'09)
 - 2009 IEEE International Computer Software and Applications Conference (COMPSAC'09)

2009 ACM Workshop on Secure Cloud Computing (CCSW'10)
 2009 Cryptology and Network Security Conference (CANS'10)
 2010 IEEE International Conference on Distributed Computing Systems (ICDCS10)
 2010 IEEE International Conference on Pervasive Computing and Communications (PERCOM'10)
 2011 IEEE Symposium on Reliable Distributed Systems (SRDS'11)
 2012 IEEE International Conference on Network Protocols (ICNP'12)
 2013 International Conference on Applied Cryptography and Network Security (ACNS'13)
 2013 ACM CCS Smart Energy Grid Security Workshop (SEGS'13)
 2013 ACM CCS Workshop on Security, Privacy and dependability for CyberVehicles (CyCAR'13)
 2013 Content-Centric Networking (CCNx) Community Meeting
 2014 Workshop on Genome Privacy and Security (GenoPri'14)
 2014 Privacy-Enhancing Technologies Symposium (PETS'14)
 2015 Workshop on Genomic Privacy and Security (GenoPri'15)
 2015 Workshop on Privacy-Aware Computational Genomics (PACG'15)
 2016 European IEEE Symposium on Security and Privacy (EuroS& P'16)
 2016 IEEE INFOCOM
 2017 ACM Symposium on Information, Computer and Communications Security (ASIACCS'17)
 2018 Privacy-Enhancing Technologies Symposium (PETS'18)
 2018 NDSS Workshop on Decentralized IoT Security and Standards (DISS'18)
 2019-2020 Usenix Security Conference (Usenix'Sec'20)
 2019-2020 Network and Distributed Systems Security Symposium (NDSS'20)
 2019 International Conference on Advanced Information Technologies (ICAIT'19)
 2021 IEEE Workshop on the Internet of Safe Things (SafeThings'21)
 2022 NDSS Workshop on Automotive Security (AutoSec'22)

- Invited Participant:

NSF Wireless and Mobile Communications Workshop, March 1997.
 NSF Critical Infrastructure Protection Workshop, July 1998.
 NASA/DoE Bridging the Gap NGI Workshop, August 1999.
 US/EU Information Survivability Workshop, June 2000.
 NSF/DoE/NASA Workshop on Challenges in Large-Scale Security, March 2003.
 NSF Research Challenges in Residential Networks, Broadband and Applications Workshop, October 2003.
 2nd Japan/US NSF Workshop on Critical Information Infrastructure Protection, June 2005.
 Dagstuhl Seminar: *Anonymous Communication and its Applications*, September 2005.
 ARO/CYLAB Workshop on Security Issues in Sensor Networks, May 2006.
 NSF GENIE Security Workshop, July 2006.
 ARDA/DTO Workshops on Privacy-Protecting Technologies, September, October and November 2006.
 US-EU Summit on Cybertrust, Dublin, November 2006.
 ARO Workshop on New Directions in Wireless and Sensor Security, February 2007.
 DoE Workshop on Critical Infrastructure Protection, July 2007.
 Zürich Information Security Workshop, September 2007.
 DHS Cyber Security Research Roadmap Workshop, March 2008.
 Dagstuhl Seminar: *Security Hardware in Theory and Practice*, June 2008.
 Dagstuhl Seminar: *Forgery-Resilient Cryptographic Hardware*, July 2009.
 US Government National Cyber Leap Year Summit, August 2009.
 NSF Future Internet Summit, October 2009.
 NSF Future Directions in Sensor Networks Workshop, November 2009.
 ARO Workshop on Cyber-Physical System Security, August 2009.
 Microsoft Research Workshop on Cloud Security, August 2010.
 CMU/NSF Trusted Computing in Embedded Systems Workshop, November 2010.
 Dagstuhl Seminar: *Privacy and Security in Smart Energy Grids*, December 2011.
 NSF Workshop on Security for Cloud Computing, March 2012.
 NSF China-US Software Workshop, March 2012.
 NSF US/Middle-East Workshop on *Trustworthiness in Emerging Distributed Systems and Networks*, June 2012.
 Intel Summer 2013 Security Curriculum Workshop, July 2013.
 Dagstuhl Seminar, *Secure Routing for Future Communication Networks*. March 2015.
 Dagstuhl Seminar, *Assuring Resilience, Security and Privacy for Flexible Networked Systems*. April 2015.
 NSF Workshop on *Wireless Security*, March 2015.
 NWO/NSF Privacy Workshop, October 2015.

ARO/UMD Workshop on *Cryptography and Hardware Security for the Internet of Things*, October 2015.
 Dagstuhl Seminar, *Modern Cryptography and Security: An Inter-Community Dialogue*, January 2016.
 ARO/ASU Workshop on *Trustworthy Human-Centric Social Networking*, May 2016.
 DARPA/DoC Workshop on *Dynamic Authentication*, February 2017.
 UCLA IPAM Workshop on Algorithmic Challenges in Protecting Privacy for Biomedical Data, January 2018.
 Dagstuhl Seminar, *Secure Routing for the Internet*, June 2018.

University Service

- System-wide & External Service:
 - External Reviewer, Computer Science Graduate Program, UC Davis, February 2018.
 - External Reviewer, School of EECS, KTH (Stockholm, Sweden), August 2021.
- Campus-wide Service:
 - Graduate Council, 2022-present.
 - Senate Committee on Committees, 2014-2017.
 - Director, Networked Systems Graduate Program, 2011-2013.
 - Senate Committee on Rules and Jurisdiction, 2009-2012.
 - Postdoc Advisory Committee, 2008-2009.
 - Responsible Conduct of Research (RCR) Task Force, 2008-2009.
 - Graduate Opportunity Fellowship Selection Committee, 2008.
 - Ombudsman Search Committee, 2007.
- School & Department Service:
 - Vice-Chair for Graduate Studies, 2016-2023.
 - Chair, Graduate Admissions Committee, 1999-2002, 2004-2007, 2009, 2017-2019.
 - Chair, Graduate Policy Committee, 2004-2007.
 - Faculty Chair of Information and Computer Science, 2001-2002
 - Associate Dean for Research and Graduate Studies, 2002-2007.
 - Vice-Chair for Graduate Studies, 2008-2010.
 - Vice-Chair, Systems Division, 2010-2011.
 - Search Committee (Security & Cryptography), 2001-2002, 2002-2003.
 - Search Committee (Mobile Computing), 2002-2003.
 - Search Committee (Computer Networks), 2003-2004.
 - Co-chair, Search Committee (General CS, 2 positions), 2006-2007.
 - Chair, Search Committee (Systems & Programming Languages), 2010-2011.
 - Search Committee (Computer Networks), 2012-2013.
 - Chair, Lecturer Search Committee, 2013-2014.
 - Chair, Search Committee (Systems), 2014-2015.
 - Search Committee (Open Mid-Career), 2016-2017.
 - Search Committee (Cybersecurity), 2017-2018.

Students

- Graduated PhD Students: advised or co-advised[†]:
 - Brenda Timmerman (deceased), 1998 (USC). Associate Professor, California State University, Northridge.
 - Giuseppe Ateniese, 1999 (University of Genova, Italy), Chaired Professor, George Mason University.
 - Yongdae Kim, 2001 (USC), Full Professor, Korea Advanced Institute of Science & Technology (Daejeon, Korea).
 - [†]Michael Steiner, 2002 (University of Saarland, Germany), Research Staff, INTEL Labs.
 - Xuhua Ding, 2003 (USC). Associate Professor, Singapore Management University (Singapore).
 - Jeong Yi, 2005 (UCI). Associate Professor, Soongsil University (Seoul, Korea).
 - Maithili Narasimha, 2006 (UCI), Sr. Research Staff, CISCO.
 - [†]Nitesh Saxena, 2006 (UCI), Full Professor, Texas A&M University.
 - Einar Mykletun, 2006 (UCI), Chief Security Architect, DELL/QUEST Software.
 - [†]Jihye Kim, 2008 (UCI), Associate Professor, Kookmin University (Seoul, Korea).
 - Di Ma, 2009 (UCI), Full Professor, University of Michigan, Dearborn.
 - Claudio Soriente, 2009 (UCI), Senior Researcher, NEC Research Europe.
 - Ersin Uzun, 2010 (UCI), Vice-President and Lab Director, Xerox PARC.
 - John Solis, 2010 (UCI), Research Staff, SANDIA National Labs.
 - Karim Eldefrawy, 2010 (UCI), Principal Computer Scientist, SRI International.
 - Emiliano de Cristofaro, 2011 (UCI), Full Professor, University College London (UK).
 - Yanbin Lu, 2011 (UCI), Software Engineer, Google.
 - Mishari Al-Mishari, 2012 (UCI), Assistant Professor, King Saud University (Riadh, Saudi Arabia).

Cesar Ghali, 2016 (UCI), Software Engineer, Google.
 Sky Faber, 2016 (UCI), Software Engineer, Google.
 Christopher Wood, 2017 (UCI), Research Lead, Cloudflare.
 Tyler Kaczmarek, 2018 (UCI), Research Staff, MIT Lincoln Labs.
 †Ahmad Ibrahim, 2019 (TU Darmstadt, Germany).
 Norrathep Rattanaivanon, 2019 (UCI), Assistant Professor, Prince of Songla University (Phuket, Thailand).
 †Ferdinand Brasser, 2019 (TU Darmstadt, Germany).
 Ivan de Oliveira Nunes, 2021 (UCI), Assistant Professor, Rochester Institute of Technology (RIT).
 Ercan Ozturk, 2021 (UCI), Research Staff, Facebook.

- Graduated PhD Students: dissertation committee member:

Anastasia Shuba, 2019 (UCI)
 Simon Eberz, 2018 (Oxford University, UK)
 Hubert Ritzdorf, 2018 (ETH Zürich, Switzerland)
 Travis Mayberry, 2015, (Northeastern University)
 Stephen Crane, 2015, (UCI)
 Lukasz Olejnik, 2015, (INRIA Rhône-Alpes, France)
 Philip Ginzboorg, 2014 (Aalto University, Finland)
 Abdelberi Chaabane, 2014 (INRIA Rhône-Alpes, France)
 Steffen Schulz, 2014 (Macquarie University, Australia)
 Lei Wei, 2013, (UNC Chapel Hill)
 Luis Angel Bathen, 2012 (UCI)
 Anh Le, 2012 (UCI)
 Todd Jackson, 2012 (UCI)
 Ahren Studer, 2011 (CMU)
 Ali Bagherzandi, 2011 (UCI)
 Kasper Bonne Rasmussen, 2011 (ETH, Switzerland) – Winner of 2011 ETH Medal
 Alessandro Colantonio, 2011 (Universita' di Roma Tre, Italy)
 Julien Freudiger, 2010 (EPFL, Switzerland)
 Dan Forsberg, 2010 (Aalto University, Finland)
 Yang Wang, 2010 (UCI)
 Xiaomin Liu, 2010 (UCI)
 Bryan Parno, 2010 (CMU) – Winner of 2010 ACM Doctoral Dissertation Award
 Gabriele Oligieri, 2010 (University of Pisa, Italy)
 Enriche Caiuch, 2009 (UCI)
 Gergely Ács, 2009 (BUTE, Hungary)
 Bijit Hore, 2007 (UCI)
 Walid Bagga, 2006, (ENST, France).
 Vivek Haldar, 2006, (UCI)
 Kumar Viswanath, 2005 (UC Santa Cruz)
 Pietro Michiardi, 2004, (ENST, France)
 Adrian Perrig, 2003, (CMU)
 Alain Pannetrat, 2002, (ENST, France)

- Former MS Students:

Kemal Bicakci, 1999 (USC)
 Damian Hasse, 2000 (USC)
 Alexei Khavaev, 2000 (USC)
 Paolo Montini, 2001 (Torino Polytechnic, Italy)
 Ashit Tandon, 2001 (USC)
 Ignacio Solis, 2001 (USC)
 Simon Skaria, 2002 (UCI)
 Elliot Jones, 2002 (UCI)
 Samad Nasserian, 2005 (RWTH Aachen, Germany)
 Stephen Asherson, 2008 (Univ. of Capetown, RSA)
 Rishab Nithyanand, 2010 (UCI)
 Audrey Hesse, 2011 (UCI)
 Andrew Williams, 2012 (UCI)
 Alberto Compagno, 2012 (Univ. of Padova, Italy)
 Marco Teoli, 2013 (Univ. of Padova, Italy)

Robert Sy, 2013 (UCI)
Quan Nguyen, 2013 (UCI)
Naveen Nathan, 2013 (UCI)
Norratthep Rattanaipanon, 2015 (UCI)
Ekin Oguz, 2015 (UCI)

- Current PhD Students:

Yoshimichi Nakasuka, from 2017 (advanced)
Sashidhar Jakkamsetti, from 2018 (advanced)
SeoYeon Hwang, from 2018 (advanced)
Andrew Searles, from 2019
Benjamin Turner, from 2020
Renaissance Tarafder Prapty, from 2020
Youngil Kim, from 2020
Elina van Kempen, from 2021

Post-Doctoral Associates and Visitors

- Daniele Mazzocchi, 2001-2002, Research Staff, ISMB, Turin, Italy.
- Shouhuai Xu, 2002-2003, Professor, University of Texas at San Antonio.
- Jung-Hee Cheon, Summer 2002 and 2006-2007, Professor, Seoul National University, Korea.
- Kyung Hyune Rhee, 2001-2002, Professor, Pukyong National University, Pusan, Korea.
- Claude Castelluccia, 2003-2005, Senior Research Staff, INRIA Rhône-Alpes, Grenoble, France.
- Daniele Perito, 2009, Postdoctoral Researcher, UC Berkeley.
- Gabriele Oligieri, 2009, Researcher, University of Trento, Italy.
- Shuichiro Ako, 2009-2010, Patent Examiner, Japan Patent Office, Tokyo, Japan.
- Ivan Martinovic, 2009-2010, Assistant Professor, Oxford University, UK.
- Paolo Gasti, 2010-2012, Assistant Professor, New York Institute of Technology.
- Kasper Rasmussen, 2011-2013, Postdoctoral Researcher, UCI.
- Juliano Kazienko, 2011-2012, Researcher, Universidade Federal Fluminense, Brazil.
- Mauro Conti, 2012 and 2013, Assistant Professor, University of Padova, Italy.
- Gergely Acs, 2012, Researcher, INRIA Rhône-Alpes, France.
- Filipe Beato, 2012, Researcher, KUL, Belgium.
- Jaroslav Šedenka, 2013, PhD student, Masaryk University, Brno, Czech Republic.
- Ronald Petrlic, 2013, PhD student, Paderborn University, Germany.
- Marc Roeschlin, 2013, PhD student, ETH (Swiss Polytechnic), Zurich, Switzerland.
- Srdjan Čapkun, 2013, Professor, ETH (Swiss Polytechnic), Zurich, Switzerland.
- Alberto Compagno, 2014, PhD student, University of Rome, “La Sapienza”, Italy.
- Luca Ferretti, 2015, PhD student, University of Modena, Italy.
- Xavier Carpent, 2015-2017, Postdoctoral Researcher, University of Louvain-le-Neuf, Belgium.
- Andrew Paverd, 2018, Fulbright Scholar, Aalto University, Helsinki, Finland.
- Pier-Paolo Tricomi, 2018, PhD student, University of Padova, Italy.

Appendix B

ExifTool Version Number : 12.52
File Name : notepad.exe
Directory : C:/Windows/system32
File Size : 166 kB
File Modification Date/Time : 2019:12:06 14:40:00-05:00
File Access Date/Time : 2023:01:26 20:09:46-05:00
File Creation Date/Time : 2019:12:07 04:51:36-05:00
File Permissions : -rw-rw-rw-
File Type : Win32 EXE
File Type Extension : exe
MIME Type : application/octet-stream
Machine Type : Intel 386 or later, and compatibles
Time Stamp : 1971:01:28 19:30:01-05:00
Image File Characteristics : Executable, 32-bit
PE Type : PE32
Linker Version : 14.20
Code Size : 140288
Initialized Data Size : 30208
Uninitialized Data Size : 0
Entry Point : 0x21860
OS Version : 10.0
Image Version : 10.0
Subsystem Version : 10.0
Subsystem : Windows GUI
File Version Number : 10.0.19041.1
Product Version Number : 10.0.19041.1
File Flags Mask : 0x003f
File Flags : (none)
File OS : Windows NT 32-bit
Object File Type : Executable application
File Subtype : 0

Language Code	: English (U.S.)
Character Set	: Unicode
Company Name	: Microsoft Corporation
File Description	: Notepad
File Version	: 10.0.19041.1 (WinBuild.160101.0800)
Internal Name	: Notepad
Legal Copyright	: © Microsoft Corporation. All rights reserved.
Original File Name	: NOTEPAD.EXE
Product Name	: Microsoft® Windows® Operating System
Product Version	: 10.0.19041.1
Warning	: Possibly corrupt Version resource