

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

FORTINET INC.,
Petitioner,

v.

BT AMERICAS INC.,
Patent Owner.

Case IPR2019-01324
Patent 7,895,641

DECLARATION OF WENKE LEE IN SUPPORT OF BT AMERICAS INC.'S
PRELIMINARY RESPONSE UNDER 35 U.S.C. §313 AND 37 C.F.R. §42.107(a)

BT EXH. 2001
FORTINET INC. V. BT AMERICAS
IPR2019-01324
PAGE 1

PAN EX1048
Palo Alto Networks v. BT Americas
IPR2023-00889

TABLE OF CONTENTS

I.	INTRODUCTION	3
II.	PERSON OF ORDINARY SKILL IN THE ART	7
III.	SHORT SUMMARY OF CONCLUSIONS	7
IV.	MY QUALIFICATIONS, COMPENSATION, AND EXPERIENCE.....	24
V.	THE INVENTIONS DESCRIBED AND CLAIMED IN THE ‘641 PATENT	26
VI.	THE FILE WRAPPER HISTORY FOR THE ‘641 PATENT	45
VII.	THE PRIOR ART.....	50
	A. General Observations on the State of the Art at the Time of Invention, Including the Jansen Reference.....	50
	B. The Hill Reference	52
	C. The Warshaw Reference	59
VIII.	THE REFERENCES CITED BY PETITIONER DO NOT TEACH THE LIMITATIONS DIRECTED TO FILTERING RESIDUE, AS THE EXAMINER APPRECIATED IN RELATION TO HILL, AND ALSO DO NOT TEACH A DYNAMIC UPDATE THROUGH FEEDBACK TO THE PROBE.....	65
	A. Neither Hill Nor Warshaw Suggest an Analysis of Residue to Flag Potential Events so That Information About Those Events Can Be Sent on for Analysis	65
	B. The Claim Limitations Directed to Updating the Probe are Also Missing from Hill, By Itself or as Combined with Warshaw	68
	C. Hill and Warshaw Both Fail to Discuss Customer Information as Required by Independent Claim 18.....	75
IX.	IT IS NOT REASONABLE TO MODIFY HILL WITH WARSHAW TO INCORPORATE A DIFFERENT FILTERING SUBSYSTEM OR FEEDBACK TO THE PROBE.	77
X.	CONCLUSION.....	79

I. INTRODUCTION

1. My name is Wenke Lee and BT Americas Inc. (“BT”) has retained me as a technical expert witness in connection with this proceeding.

2. A curriculum vitae (“CV”), which I have recently reviewed and believe to be accurate, is attached as Exhibit A to this Declaration. My particular area of expertise is network security and attendant technologies, with an emphasis on the detection of malware and network intrusions, as well as remedial measures to be employed following detection.

3. I have been asked to provide insight into the state of the art during the relevant time frame. In this regard, the stated Field of Invention of the ‘641 patent (Exh. 1001, col. 1 ll. 18-20) matches my own area of expertise and work, starting with my PhD thesis, which was completed in 1999. My thesis is directed to a data mining framework for constructing features and models for network intrusion detection systems. It describes algorithms and procedures for processing network traffic data and extracting useful attributes and patterns (or signatures) for the purpose of detecting attacks to enterprise networks.

4. I will also address the technology described and inventions claimed in the ‘641 patent, the contents of the file wrapper history for the ‘641 patent, the

references cited by Petitioner, and the differences between the inventions claimed in the '641 patent and the disclosures of the cited references. I will also address related technical points which may prove helpful to the Board in assessing factual assertions and opinions set forth in Petitioner's papers and exhibits, all of which I have read and will address as appropriate to the present context, except for the Declaration of Sylvia D. Hall-Ellis, Exhibit 1007, which I skimmed but did not read carefully.

5. I am not an attorney, but I am a named inventor in six patents, five of which are directed to detecting intrusions, malware, or responding to network attacks (as well as detecting them). I have also served as an expert and have provided testimony in a number of prior patent litigations. Consequently, I have a general familiarity with principles of patent law respecting validity and obviousness.

6. I understand that a patent is obvious if the differences between the claimed subject matter and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art. BT's counsel has informed me that obviousness is a question of law based on underlying questions of fact.

7. I understand that the underlying factual inquiries in an obviousness analysis include: (1) determining the scope and content of the prior art; (2) resolving the level of ordinary skill in the prior art; (3) ascertaining the differences between the claimed invention and the prior art; and (4) considering objective evidence of nonobviousness.

8. I also understand that the Petitioner, Fortinet Inc., has the burden to show that a person of ordinary skill in the relevant field had a reason to combine the elements in the manner claimed when asserting obviousness in view of a combination of references. BT's counsel has informed me that I should value "common sense" over "rigid preventative rules" in determining whether a motivation to combine references exists. I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed. However, I also understand that if a person of ordinary skill in the art would have had reason to attempt to make the composition or device, or carry out the claimed process, Petitioner must also demonstrate that such a person would have had a reasonable expectation of success in doing so.

9. I understand that the express, implicit, and inherent disclosures of a prior art reference may be relied upon in the rejection of claims for anticipation and obviousness.

10. Further, I understand that hindsight may not be used to determine obviousness. In other words, it is wrong to use the patent as a guide through the prior art references, to combine the right references in the right way so as to achieve the result of the claims of the challenged patent.

11. I further understand that a reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the patent applicant. I also understand that a reference may teach away from a use when that use would render the result inoperable.

12. I understand that there is no suggestion or motivation to make a modification to a prior art reference if the proposed modification would render the prior art invention unsatisfactory for its intended purpose. I also understand that an obviousness allegation cannot be supported by a combination of references that would require a substantial reconstruction and redesign of the elements shown in

the primary reference as well as a change in the basic principle under which the primary reference was designed to operate.

II. PERSON OF ORDINARY SKILL IN THE ART

13. For the purposes of this declaration, I assume that a person of ordinary skill in the art (a “POSITA”) pertaining to the ‘641 patent at the time it was invented has the education and/or experience that Dr. Reddy has set out in his declaration. *See, e.g.*, Exh. 1003 ¶¶24-25.

III. SHORT SUMMARY OF CONCLUSIONS

14. I understand that the immediate question is whether there is a reasonable likelihood that Petitioner would prevail in showing any of the claimed inventions are invalid for obviousness. I do not believe there is any such likelihood and that the inventions claimed in the ‘641 patent constitute a substantial advance in the art and were not obvious in any way as I understand that term to be employed in the law of patents. There are at least two separate limitations present in every single claim that would remain missing from the cited references, even if the references were to be combined as presented by Petitioner, and no reason to combine those references in the manner Petitioner presents exists. My principal reasons for reaching these conclusions follow.

15. To begin, everything the Patent Examiner stated in her Reasons for Allowance in the parent application that resulted in issuance of U.S. Patent No. 7,159,237, including her comments about the Hill et al. patent, attached as Exhibit 1004 (“Hill”), which were reiterated verbatim in the ‘641 patent file history, as well as the further examination provided in the file history of the application for the ‘641 patent, is entirely correct. In relation to the subject matter of independent claim 1 of the ‘641 patent, the Examiner also found and applied the closest work known to me to have been in existence at the time (i.e., the works of Esbensen and Porras et al.) and found it distinguished by the limitation directed to an analysis of post-filtering residue. Consequently, the Examiner suggested adding that limitation to the claim of the application that resulted in independent claim 18 of the ‘641 patent. Exh. 1002, p. 63. For the following reasons, I agree with her conclusions.

16. Everything Petitioner identified as disclosed in Hill that is supposed to make it similar to the ‘641 patent (Petition at 24-34) was identified in the Esbensen prior art by the Examiner during prosecution of the parent application. The ‘641 patent that issued from the parent patent application gave rise to a double patenting rejection during the prosecution of the claim that issued as claim 1 of the ‘641

patent. Although the double patenting rejection was overcome by filing a terminal disclaimer, it shows the relevance of Esbsensen to the '641 patent. Hill does not actually show all that Petitioner claims and is, in my opinion, less relevant than the Esbsensen prior art the Examiner chose to apply directly to the language of the claims. The Warshaw et al. reference, attached at Exhibit 1005 ("Warshaw"), and cited by Petitioner is less relevant than the Porras prior art as a secondary reference because Warshaw does not filter status data, as required by the claims, whereas the Examiner concluded the Porras prior art does. Presuming that the Examiner applied the correct legal standard to her technical conclusions, there is no basis whatsoever to second-guess her conclusions.

17. One particularly novel feature of the inventions claimed in the '641 patent is the use of a filtering subsystem, which looks at status data to positively filter out what is bad in the traffic, negatively filter what is good, and retain the residue. The residue (generally thought at the time of the invention to be of little interest) is then analyzed to identify potentially security-related events. Once a potential event has been flagged from an analysis of the residue, information about that particular potential event is sent to an analyst system at a secure operations center ("SOC") for deeper analysis, with a dynamic feedback mechanism from the

SOC back to the probe. In the preferred embodiment, the SOC is directed to monitoring the operations of different customers and approaches its analysis on a customer-specific basis, incorporating customer-specific information into the analysis and resolution of the potential security event. None of this is actually disclosed in Hill and Warshaw.

18. There are two independent claims in the '641 patent: claims 1 and 18. The relevant claim language present in each claim which relates to the first analytical step of filtering out residue status data and analyzing that residue to find potential events can be found in element (b) of claim 1 as follows:

analyze status data to identify potentially security-related events represented in the status data, wherein the analysis includes filtering followed by an analysis of post-filtering residue, wherein the post-filtering residue is data neither discarded nor selected by filtering;

Exh. 1001, col. 33 at ll. 31-35 (emphasis added). It can also be found within the first step of independent method claim 18 as follows:

creating an event record for information received about an identified potentially security-related event occurring on the network, wherein the potentially security-related event is identified by filtering followed by an analysis of post-

filtering residue, wherein the post-filtering residue is
neither discarded nor selected by the filtering;

Id. at col. 34 at ll. 25-30 (emphasis added).

19. Whereas claim 18 is directed to a “method of operating a secure operations center” (per the preamble) and speaks to information received, claim 1 is directed to the operation of a probe as part of a security monitoring system and contains a separate limitation respecting transmission of information about a potential event to the analyst system resident at the secure operations center. The relevant language in this regard appears as element c of claim 1:

transmit information about the identified events to an
analyst system associated with the security monitoring
system;

Id. at col. 33 at ll. 36-38.

20. The analysis of post-filtering residue to identify potential security events so that further analysis of the potential event might be performed by an analyst system at a SOC was specifically identified by the Examiner as a novel and distinguishing feature of the invention in her stated Reasons for Allowance. These features are neither disclosed nor suggested by the references cited by Petitioner.

21. It is a major advantage to identify potentially security-related events from residue, before anything is sent to the analyst system of the SOC. There is so much traffic (and attendant status data) in most networks that it is not realistic (in terms of cost, speed, and other operational parameters affected by high volumes of traffic) to send everything indeterminate on to the SOC for analysis. By conducting an analysis of residue to flag potentially security-related events before transmission to the SOC, the architecture described in the claims of the '641 patent manages to discover previously unknown intrusions into the customer network and manages the enormous amount of data filtered more efficiently. Furthermore, by conducting two levels of analysis, the first at the probe prior to transmission across pipes 3000 and the second at the SOC after transmission, false positives are reduced.

22. Hill is not, as Petitioner claims, virtually identical to the claimed invention. Hill does not discuss status data, does not discuss filtering status data, and does not contain the remotest suggestion of filtering status data into three buckets (positive, negative, and residue) so that residue can be analyzed to flag potential events. In short, Hill does not employ filtering to flag potential events from residue, first, and only then transmit information about a potential event to an

analyst system at a secure operations center for further analysis to determine whether a security event actually occurred.

23. Furthermore, Hill finds events by comparison to known signatures and variants of those known signatures created by artificial intelligence training exercises. Hill does not find anything until it is trained to find certain signatures and is not capable of detecting intrusions that are completely unknown, i.e., that are not variants of known signatures to which the processor can attempt to make a match.

24. The inventions of the '641 patent offer a solution that is capable of finding a far broader range of intrusions, including those previously unknown, by flagging any kind of potentially security-related event in residue status data. Rather than employing artificial intelligence to match to training signatures, the '641 patent has a SOC where a trained security analyst who has access to sophisticated tools and extensive databases can determine whether a potential event flagged from residue is an actual intrusion and/or address how best to resolve an event by invoking all the resources available to a security analyst at the SOC.

25. Hill also lacks other features of the claimed inventions, as well. Addressing independent claim 1, Hill does not describe a dynamic feedback

mechanism to update the analysis capability of the probe through the receipt of feedback at the probe, as claimed in elements (d) and (e) of claim 1 (and its dependents), which reads as follows:

- d) a receiver for receiving feedback at the probe based on empirically-derived information reflecting operation of the security monitoring system; and
- e) a modification control system for dynamically modifying an analysis capability of the probe during operation thereof based on the received feedback.

Id. at col. 33. ll. 39-44.

26. The “probe” of the ‘641 patent is described in the Summary of the Invention at col. 3 ll. 11-26 and shown as element 2000 in Figures 1 and 2. The probe collects and filters status data to identify potentially security-related events from residue status data, i.e., status data not positively or negatively filtered, and creates sentry messages about the potential events for transmission to the SOC (shown to the right of Figure 1) for more detailed analysis.

27. On the other hand, the components in Hill that most closely resemble the claimed probe are security agent 36, operating in conjunction with area server 30. As developed above, security agent 36 and area server 30 do not perform the same filtering functions claimed in the ‘641 patent in that they do not filter out

residue status data or analyze residue data to identify potential events. Security agent 36 and area server 30 do, however, look for security events at the nodal level of the architecture of Hill as described at col. 4 ll. 31-41 of the Hill reference. Data about security events collected by security agents 36 is then sent on to processor 40. Exh. 1004, col. 4 ll. 53-55.

28. The components of Hill that analyze incoming traffic for events and send data about those events are consequently the components that would correspond to the “probe” if they in fact performed the claimed functions, which they do not. These components find events and send data about them on for analysis at processor 40. The probe in the ‘641 patent finds potential events and sends information on to the SOC.

29. In one part of its papers, Petitioner considers processor 40 and display 42 of Hill as corresponding to the analyst system resident at the SOC of the ‘641 patent, since they receive transmissions about events and analyze them. *See, e.g.*, Exh. 1003 ¶¶115, 124, 130, 140. In another part of its papers, Petitioner says that processor 40 is also part of the “probe” 2000 of the ‘641 patent. *See, e.g., id.* at ¶¶102, 130. These mappings make absolutely no sense since the probe and the SOC are two completely different portions of the ‘641 patent architecture. *See,*

e.g., Figure 1 of the '641 (showing, for example, the probe to the left in the customer network and the SOC to the right within the SOC firewall, with secure transmission pipes 3000 between them). A person of ordinary skill in the art would understand in a network security monitoring system that a probe and a SOC server are different components serving different purposes.

30. The reason Petitioner takes this facially absurd position is that the security agents 36 and area servers 30 of Hill are not dynamically updated based on feedback received at the probe. As Petitioner's own papers reflect, the only component that is updated in the Hill reference is processor 40. Consequently, to even begin to attempt to find elements d and e of claim 1 of the '641 patent in Hill, and the purportedly invalidating combination of Hill over Warshaw, Petitioner is forced to claim that processor 40 is also part of the probe.

31. Processor 40 of Hill cannot be part of the claimed probe. This is because processor 40 does not scan for events at the nodal level. Processor 40 plays no part in the initial scan of traffic resulting in the transmission of data about events or potential events, but rather receives data about events after its transmission. The only components in Hill that can be roughly analogized to the probe, security agents 36 and area servers 30, never receive dynamic feedback

from which they are updated. As a result, elements (d) and (e) of claim 1 and dependent claims 2-17 of the '641 patent are lacking in the references cited by Petitioner.

32. Additionally, the Warshaw reference, assuming it qualifies as prior art, does not flag potential events from an analysis of residue so that information about potential events so identified can be transmitted to a SOC. Since Warshaw is missing the entire step of conducting an analysis of residue to flag potential events, it does not fill any gaps in the architecture described in Hill so as to allow Warshaw and Hill to be combined in a way that even arguably approaches the architecture of the '641 patent. Furthermore, Warshaw does not filter status data at all, while the claims of the '641 patent all require the identification of a potentially security-related event from residue (which is the status data that was neither selected nor discarded).

33. In that last regard, to the extent Warshaw addresses in paragraph 6.2 (discussed below) what I would consider to be status data, the reference indicates that the system of Warshaw was not capable of using status data at all. Since Warshaw does not employ status data or discuss how to employ status data, the very thing upon which the '641 patent relies to flag potential security events, there

would appear to be no reason to modify the architecture of Hill in that direction as a result of Warsaw.

34. Like Hill, Warsaw simply misses the entire point of the filtering subsystem, as claimed and called out by the Examiner in her Reasons for Allowance. This is because Warsaw transmits not only everything that was bad (“suspicious”), but also transmits everything that was indeterminate (“uncertain”), to the analysis layer. *See, e.g.*, Exh. 1005, p. 3. Therefore, Warsaw does not flag potential events from any form of residue (much less residue status data) that was neither selected nor discarded as the claims require and focus the subsequent transmission of information on the potential events so identified. In Warsaw, everything not negatively filtered (as uninteresting) is selected for transmission onwards for further analysis. *See, e.g., id.* at 3 (“A probe sniffs, in real time, all network traffic by discarding obvious non-threatening traffic . . .”). That is, there is no specific analysis of post-filtering residue as claimed in the ‘641 patent.

35. To attempt to address the additional volume of information that has to be transmitted under its approach, Warsaw spaces out the transmission into separate intervals. This is a slower and more cumbersome procedure directed to managing a volume of traffic that can be expected to be several orders of

magnitude higher than that typically achieved by the '641 patent, which only transmits to the SOC information about potential events flagged at the probe. In this last regard, potential events identified from the residue status data are comprised of anomalies outside of known attacks and usually involve a very small percentage of traffic flow.

36. To summarize, there is absolutely nothing wrong with the Examiner's analysis and conclusions respecting the limitations that relate to an analysis of "residue" to flag potential events from that residue so that information about the events can be transmitted for deeper analysis. Residue would have been considered unimportant by most persons skilled in the art at the time, but the inventors recognized that it was in the residue that clues about previously unknown intrusions might be found. Thus, they created an architecture that allowed for discovery of previously unknown intrusions and dynamic updates of the probe to recognize them, while managing large volumes of traffic and reducing false positives. The claim limitations that relate to this, which the Examiner concluded were not in the art of record, are also not in the references cited by Petitioner.

37. Furthermore, the inventors dynamically update the probe with feedback from the SOC so that the probe can filter more efficiently in the future.

Hill does not provide dynamic feedback to the probe in order to update its capabilities, as required by elements (d) and (e) of claim 1 and dependent claims 2-17.

38. In relation to claim 18 and its dependents, the language within the first limitation, which indicates that the event in question is one identified through an analysis of post-filtering residue neither selected nor discarded during filtering, is not the only limitation missing from Petitioner's cited references. Since claim 18 is directed to the operation of the SOC, rather than the probe, it does not contain language directed to updating the probe. It does, however, contain language directed to customer-specific methodologies employed at the SOC which are nowhere to be found in the references cited by Petitioner.

39. In this last regard, the remaining limitations of claim 18, which follow the "creating" step, read as follows:

correlating the event record with customer information
and a symptom record;
using the correlated symptom record to link the event
record to problem resolution information;
consolidating the event record, correlated customer
information and symptom record, and linked
problem resolution assistance information into a

problem ticket; and
providing the problem ticket to a security analyst console
for analysis.

Exh. 1001, col. 34 ll. 31-39.

40. As reflected in the preamble to claim 18 (reading: “A method of operating a secure operations center as part of a security monitoring system for a customer computer network, comprising”), the steps relate to a systems architecture and method of operation that allow a remote SOC to monitor the operations of multiple customers. In fact, the “problem ticket” recited in the claim “represents a discrete security-related event or incident of possible intrusive activity happening on a customer’s network.” Exh. 1001, col. 3 ll. 47-49. As shown in Figure 1, there is a SOC on the right that may communicate with multiple customer sites, only one of which is depicted, as an example, in Figure 1.

41. Hill is not intended to be used as a monitoring service for customers. It makes no mention of customers at all, something Petitioner infers as a possibility from a passing statement in Hill that the nodes monitored by security agents may be geographically remote. It is not stated anywhere in Hill that the equipment at those nodes is operated by different customers or that there is a SOC designed to monitor the operations of disparate customers. Hill is directed to a single

integrated network containing a processor 40 and display 42, which Petitioner calls a SOC, but those components are not the SOC disclosed in the ‘641 patent.

42. The SOC contemplated in the ‘641 patent is described under its own heading in the Summary of the Invention at col. 2 l. 65–col. 3 l. 3:

C. Secure Operations Centers (“SOCs”): MSM service security analysts would preferably work in secure operations centers. To ensure the security of their customer networks, the SOC should be connected to these networks preferably through cryptographically secured tunnels (also called “pipes” or “pathways”).

43. The MSM service referenced in this quote is a managed security monitoring service for monitoring a customers’ network activity. *Id.* at col. 1 l. 59–col. 2 l. 5. Trained security analysts working at the SOC would have extensive databases to draw upon (*id.* at col. 2 ll. 18-24), and access to “sensitive customer information” (*id.* at col. 2 ll. 46-50). The flowchart of Figure 5, directed to the handling of a problem ticket (*id.* at col. 10 ll. 45–col. 11 l. 3) contemplates an interaction with the customer to obtain as much information as is advisable to resolve the problem. Consequently, the limitations of the method of claim 18 expressly contemplate customer-specific action.

44. For example, the first element of the claim calls for creating an “event record” about the identified potentially security-related event. The next element calls for “correlating the event record with customer information and a symptom record.”

45. Addressing the event record itself, the ‘641 patent states that these are first created from gateway messages, and then may be linked with other event records and information from a variety of databases “including customer information from client information database 6022 and problem resolution information from problem/event resolution database 6023” to form “problem tickets.” *Id.* at col. 10 ll. 15-27. In short, the potentially security-related event is analyzed in the context of its having arisen at the site of a particular customer. In this fashion, the operations and history of a particular customer can be taken into consideration in analyzing and resolving the security issue raised by the event.

46. Neither Hill nor Warshaw contemplate anything like this. Neither reference correlates any “event record” with “customer information” as required by the second step of independent claim 18.

47. Neither is there any discussion of a “problem ticket” in either Hill or Warshaw, much less one that consolidates “the event record, correlated customer

information and symptom record, and linked problem resolution assistance information” as required by the last limitation of claim 18.

48. In short, claim 1 and its dependents are missing the limitations directed to an analysis of post-filtering residue and the limitations directed to updating the analytical capabilities of the probe dynamically through feedback. Claim 18 and its dependents are missing the limitations directed to an analysis of post-filtering residue and the limitations directed to a customer-specific analysis and resolution of potentially security-related events at the SOC.

49. Two different claim limitations of every claim of the ‘641 patent are missing from Petitioner’s cited references, however forced in combination. There is also no reason at all to combine Petitioner’s references since Warshaw does not even use status data, or filter out any form of residue from which to flag potential events to allow subsequent transmission of information about them for further analysis.

IV. MY QUALIFICATIONS, COMPENSATION, AND EXPERIENCE

50. I am currently employed as a Professor at Georgia Institute of Technology where I hold the John P. Imlay Chair in the College of Computing. I am receiving compensation from BT for my services in this proceeding at the rate

of \$550/hr. My compensation is not dependent upon the outcome of this proceeding and has not affected the statements I am making in this Declaration in any way.

51. My main area of interest and the focus of my current research is malware analysis and detection. A fuller description of my current interests and areas of research can be found at the bottom of page 2 and top of page 3 of my CV.

52. As reflected in the succeeding pages of my CV (pages 4-6), I have taught courses directed to Network and Information Security essentially continuously since 1999.

53. I have published more than a hundred academic papers in top journals and conference proceedings, published half a dozen books or parts of books, and participated in well over a hundred Conference Presentations and over a dozen Workshop Presentations, the vast preponderance of which relate to Network Security and most of which relate to Intrusion Detection in particular.

54. I have also performed and supervised a substantial amount of research in the field of Network Security. A list of the 46 Research Proposals and Grants in which I have participated can be found at pages 26-31 of my CV.

55. Research Honors and Awards I have received are listed on page 32 of my CV and include an award received in 2018 for co-authorship on a paper deemed to have had the greatest impact on security research and practice of the past decade. I am a Fellow of the Association for Computing Machinery (ACM), which is the world's largest scientific and educational computing society; only the top 1 percent of ACM members are recognized as Fellows.

56. I am also active in Conference Committee Activities and Seminars, all as reflected towards the back end of my CV.

57. I am named as an inventor in six patents (listed on page 38 of my CV), most of which relate to the Field of Invention at hand, which is described in the next paragraph.

V. THE INVENTIONS DESCRIBED AND CLAIMED IN THE '641 PATENT

58. The Field of Invention to which the '641 patent relates is "network security and, more specifically, to methods and systems for dynamic network intrusion monitoring, detection and response." Exh. 1001, col. 1 ll. 17-20.

59. The inventions described in the '641 patent "offer[] methods and systems for dynamic network intrusion monitoring, detection and response." *Id.* at col. 1 ll. 57-59. In one implementation, discussed in the Summary of Invention,

the '641 patent is directed to a managed security monitoring system for monitoring a customer's network. *Id.* at col. 1 l. 59–col. 2 ll. 2; col. 2 ll. 3-5, 28-29.

60. Managed security monitoring was a business in its infancy in 2000 and the architecture disclosed in the '641 patent, while not limited to monitoring customer sites, is certainly well suited to it as reflected in Figure 1. The dominant commercial practice at the time was to install pattern-matching anti-virus software to analyze incoming traffic at the edge of the network where traffic comes in from the internet and update that anti-virus software periodically with software upgrades. Remote monitoring services were known, but not common, and there was no well-considered architecture to run them to help discover unknown intrusions.

61. The '641 patent describes an entirely novel architecture for remotely monitoring a network for intrusions. An overview of a preferred embodiment is provided in Figure 1 copied below:

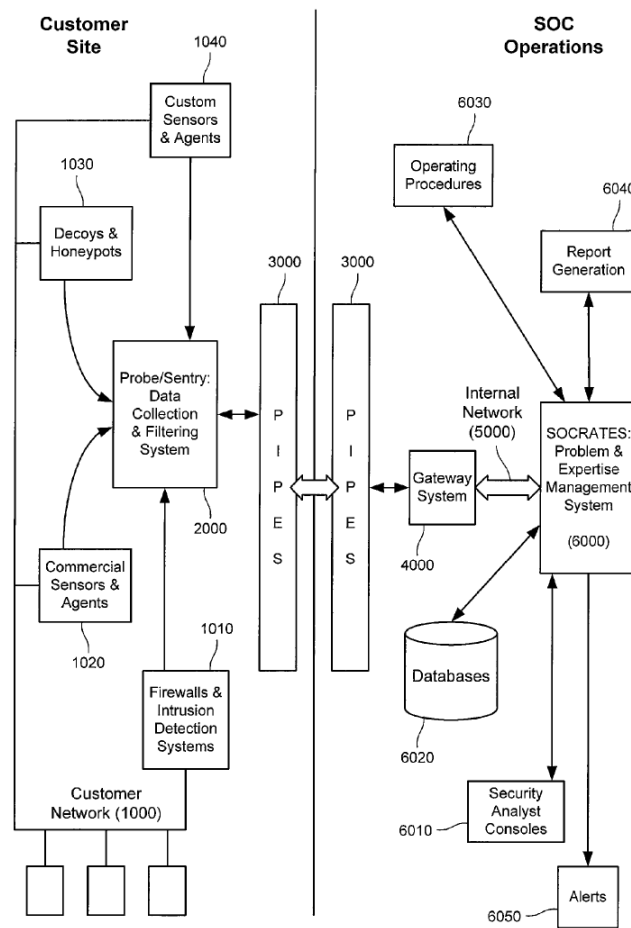


FIG. 1

62. In Figure 1, there is a customer site, which is depicted to the left of the figure, and a Secure Operations Center (the SOC) which is depicted to the right. *Id.* at col. 4 ll. 44-49. A single SOC “can monitor and service multiple customer sites....” *Id.* at col. 4 ll. 49-50. Pipes 3000 provide a communications pathway, preferably encrypted and secure, between the customer site and the SOC. *Id.* at col. 5 ll. 50-54.

63. For purposes of clarity, I will move through the system as would typically occur during its operation. Consequently, I will first address the operation of the probe at the customer site as shown to the left of Figure 1, and, while doing so, focus in particular on some of the novel features captured by the claims. After discussing transmission of information to the SOC across pipes 3000, I will discuss the operation of the SOC and components shown to the right of Figure 1. Then, I will address the transmission of feedback from the SOC to the probe so as to dynamically update the probe once the analysis at the SOC has been completed.

64. A probe 2000, located at the customer site (i.e., to the left in Figure 1), collects and filters status data extracted or derived from traffic, conceptually speaking, into three categories. The first is positively-filtered data, the second is negatively-filtered data, and the third is “residue,” which is itself analyzed to flag potentially security-related events.

65. The status data which is filtered includes audit log and other audit information per col. 4 ll. 64–col. 5 l. 3. Status data provides information about the status of the network 1000 and components and, in the preferred embodiment, is

exemplified by fields extracted from packets and the identification of components involved in a potential event.

66. The '641 patent also states that status data is collected into sentry messages sent to the SOC, where the sentry messages are transformed into gateway messages as shown, for example, in Table 6. As stated at col. 8 ll. 66–col. 9 l. 2: “Communications and resource coordinator 2060 creates sentry messages out of the interesting status data and forwards those messages on to gateway system 4000....” Consequently, Table 6, located at columns 21-22, exemplifies the types of information comprising status data that might be included in a sentry message and gateway message derived from it. *Id.* at col. 9 ll. 2-7.

67. In reviewing Petitioner’s discussion of “status data” at page 8 of the Petition (tracking Dr. Reddy’s discussion at paragraph 40 of his declaration), I concluded that it was largely, if not entirely, consistent with my own understanding except for one potential difference. At the end of the discussion, Petitioner states that a POSITA “understood that status data collected by sensors includes traffic data.” Petition at 8.

68. This statement, read literally, is accurate in that we each agree that field data extracted from packets, as shown for example in Table 6, comprises

status data. However, Petitioner appears to imply in later arguments that all traffic is status data, which is not accurate. *See, e.g., id.* at 41 (“Warshaw discloses that the analysis layer (*filtering subsystem*) analyzes traffic to identify and report intrusions (*security-related events represented in the status data*).”). By way of example, unstructured ASCII, discussed in Warshaw, which I address below, is not status data. This follows from the fact that unstructured ASCII data does not convey a meaning, which is informative as to the status of the operation of the network or its components.

69. Towards obtaining status data in the first instance, the probe collects status data from sensors in the form of firewalls and intrusion detection systems 1010, commercial or custom sensors or agents 1020 and 1040, and decoys and honeypots 1030. Exh. 1001 at col. 4 ll. 55-64. The probe can monitor and collect information from any component that can provide status data. *Id.* at col. 4 l. 64—col. 5 l. 3.

70. The features of the architecture of the filtering subsystem and associated components of the probe, which work together to create a sentry message for transmission to the SOC, are unique features of the inventions and are depicted, schematically, in Figure 2, copied below:

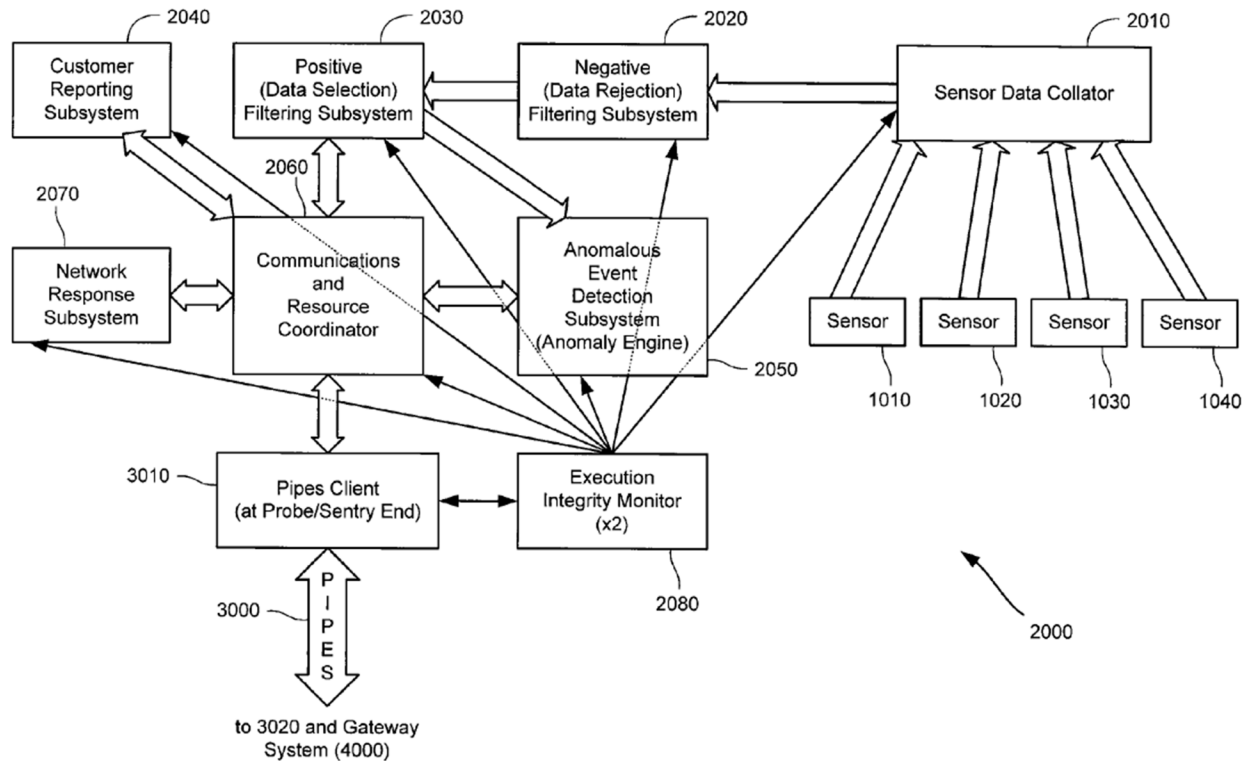


FIG. 2

71. In Figure 2, which depicts probe/sentry system 2000, a sensor data collator 2010 collates status data collected by the sensors. Then, as stated at col. 8

ll. 51-59:

Once collated, the data is first filtered by negative filtering subsystem 2020, which discards uninteresting information, and then by positive filtering subsystem 2030, which selects possibly interesting information and forwards it to communications and resource coordinator 2060. Data neither discarded by negative filtering subsystem 2020 nor selected out as interesting by positive

filtering subsystem 2030 form “residue,” which is sent to anomaly engine 2050 for further analysis.

72. Anomaly engine 2050 picks through the residue to determine what residue information is sufficiently interesting that it should be sent to the secure operations center (SOC) where trained analysts, having access to tools and databases allowing for a more sophisticated analysis, can make an informed determination as to whether there has been an intrusion. *Id.* at col. 8 ll. 59-63.

73. It is first noteworthy that the inventors had the foresight to pick through leftover residue status data to flag information within it sufficiently interesting that it was worth sending on to the SOC. Most persons skilled in the art at the time would have considered it uninteresting and discarded it. However, the residue is where the status data corresponding to unknown intrusions (i.e., those which the filtering system was unable to identify and filter out as bad) actually resides. In short, it is a significant advance for security monitoring systems to include analysis of post-filtering residue.

74. After anomaly engine 2050 identifies residue interesting enough to send to the SOC, it is sent to communications and resource coordinator 2060, which creates sentry messages sent through pipes 3000 to the SOC. *Id.* at col. 8 ll. 67–col. 9 l. 10.

75. More particularly, after filtering by the probe, status data of sufficient interest is incorporated into sentry messages, which are sent to the SOC through pipes 3000 and are received by gateway system 4000, where they are transformed into gateway messages. *Id.* at col. 5 ll. 25-29, 44-49.

76. In this fashion, a decision is made at the probe, after filtering, based on an analysis by anomaly engine 2050 of leftover status data (residue), as to whether there might have been a potentially security-related event worth investigating. The significant advantage of conducting the analysis of leftover residue at the probe is to flag potentially security-related events before transmission in order to control the amount of information subsequently transmitted to the SOC.

77. The generation of “residue” status data and the analysis of it to flag potential events worth investigating at the SOC is an entirely novel aspect of the ‘641 patent, and was called out by the Examiner in her stated Reasons for Allowance. Consequently, I digress briefly to point out the claim limitations of the independent claims directed to these particular novel features.

78. The generation and analysis of residue is captured by language found in element (b) of claim 1 (and its dependents) as follows:

analyze status data to identify potentially security-related events represented in the status data, wherein the analysis includes filtering followed by an analysis of post-filtering residue, wherein the post-filtering residue is data neither discarded nor selected by filtering;

See id. at col. 33 ll. 31-35 (claim 1) (emphasis added); *see also id.* at col. 34 ll. 25-30 (showing claim 18).

79. The next limitation of claim 1, which establishes that the information transmitted to the SOC is information about potentially security-related events already identified (through analysis of residue status data) is found in element (c) of claim 1 (and its dependents) as follows:

transmit information about the identified events to an analyst system associated with the security monitoring system;

See id. at col. 33 ll. 36-38 (emphasis added).

80. This first level of analysis, prior to transmission, takes place at the probe. Moving to the SOC shown to the right in Figure 1, sentry messages first pass through pipes 3000 into a gateway system 4000.

81. Gateway system 4000 is broken out schematically in Figure 3 and described at col. 9 ll. 42-53. Sentry update manager 4010 is responsible for

updates sent back to the probe and a communications and management coordinator 4020 converts sentry messages into gateway messages passed onto the SOC.

82. The “problem and expertise management system” that manages and addresses actual and potential problems at the customer site, e.g., as reported through sentry messages generated from an analysis of residue status data, is called SOCRATES (an acronym for Secure Operations Center Responsive Analyst Technical Expertise System per col. 3 ll. 61-62), which is identified as element 6000 in Figure 1.

83. As stated at col. 9 ll. 56-62 of the ‘641 patent:

The SOCRATES system is a consolidated system used to manage customers’ problems and the supporting data helpful in resolving such problems. It provides security analysts at a SOC a single, integrated system with which to track information concerning, for example, problems, companies, people, contacts, tools, and installed network components and known vulnerabilities.

84. The SOCRATES system and its associated databases and console is best shown in Figure 4, with the succeeding figures showing incident workflows and the handling of customer calls when the system is in use.

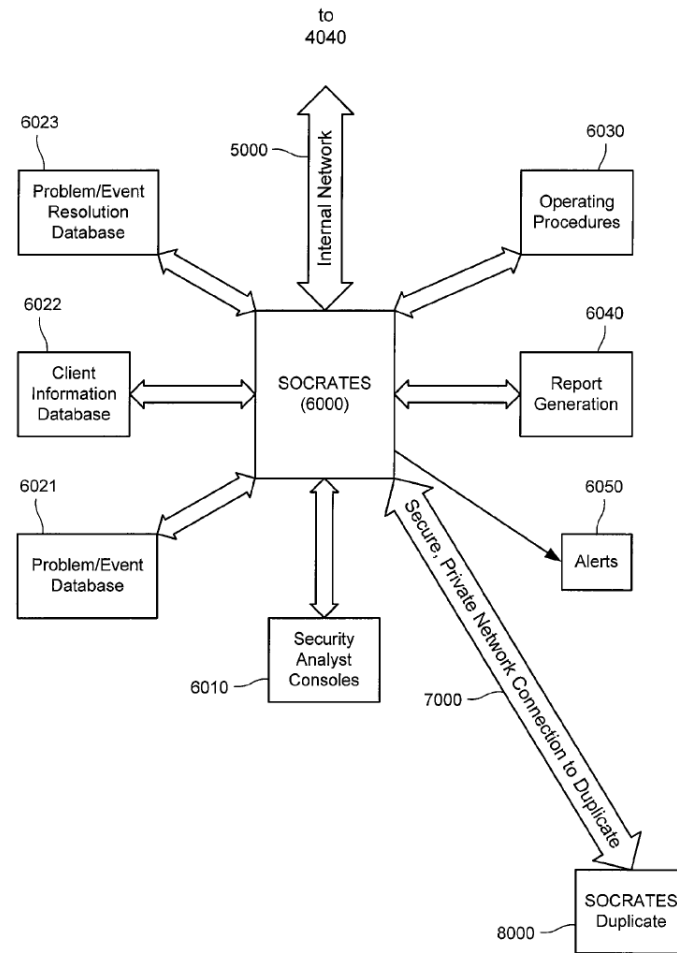


FIG. 4

85. As reflected in Figure 4, security analyst consoles 6010, which interface with SOCRATES, are available to security analysts, i.e., “personnel specializing in the analysis of network attacks.” *Id.* at col. 2 ll. 43-44. In an exemplary embodiment, security analysts at the SOC work in concert with SOCRATES to categorize, prioritize, investigate, and respond to customer incidents or problems. *Id.* at col. 3 ll. 42-45.

86. SOCRATES has access to multiple databases, shown as 6021, 6022, and 6023, which provide “security intelligence” in the form of characteristics and known vulnerabilities of certain products, characteristics of hacker tools, and the effectiveness of alternative responses. *Id.* at col. 2 ll. 55-64.

87. In keeping with the preferred implementation of providing a managed security monitoring system for customers (col. 1 l. 59–col. 2 l. 2), SOCRATES can generate customer-specific reports 6040 or send customer-specific alerts 6050 (col. 7 ll. 7-9, 32-33). The operating procedures 6030 for operating SOCRATES itself can be tailored to meet specific customer needs. *Id.* at col. 7 ll. 61-63.

88. In operation, a sentry message is sent from the probe, reformatted into a gateway message and transmitted across internal network 5000, to inform SOCRATES of the potential event. *Id.* at col. 10 ll. 10-15. SOCRATES then uses the gateway message to form an event record. The event record is linked with other event records, customer information, problem resolution information, and other information available from the databases associated with SOCRATES to form “problem tickets” displayed on console 6010 for a security analyst to handle. *Id.* at col. 10 ll. 15-32; col. 10 ll. 45 et seq.

89. Once the analysis of a potential event has been completed at the SOC, the security monitoring system provides feedback from the SOC to the probe which contains a communications and resource coordinator within 2060 to update software at the probe dynamically to provide the probe with an enhanced capability with which to perform filtering. *Id.* at col. 5 ll. 32-43.

90. Claim 18 of the ‘641 patent (and its dependents) is directed to a “method of operating a secure operations center as part of a security monitoring system for a customer computer network.” In this regard, the body of the claim, as well as the preamble, incorporates the customer-specific aspects of its operation.

91. The first limitation of claim 18 requires the creation of an event record. An event record is created from gateway messages generated within the SOC. *Id.* at col. 10 ll. 13-15. The event record is then linked with other information such as customer information and problem/resolution information resident in databases accessible at the SOC. *Id.* at col. 10 ll. 15-28.

92. Furthermore, the event record is “for information received about an identified potentially security-related event occurring on the network, wherein the potentially security-related event is identified by filtering followed by an analysis of post-filtering residue, wherein the post-filtering residue is neither discarded nor

selected by the filtering.” *Id.* at col. 34 ll. 25-30 (referring to claim 18).

Consequently, the SOC is generating an event record about the potentially security-related event flagged through an analysis of residue, and that remains a distinguishing feature over the prior art for claim 18 as well as claim 1.

93. The next step of claim 18 is “correlating the event record with customer information and a symptom record.” This, in itself, distinguishes the Hill and Warshaw references, neither of which create an event record that is correlated with customer information. Hill and Warshaw do not concern themselves with servicing customers and would have no reason to do so. Petitioner argues that the location identifiers 60 of Hill can be equated with customer information, but those location identifiers 60 only identify the location of the nodes under attack. They say nothing about who runs them. As described in Hill, the nodes are part of one integrated system presumably run by one outfit.

94. One obvious benefit of accessing customer information and linking it to the event record is to provide additional information specific to a given customer that may be helpful to the resolution of the problem, such as a customer-specific history or information about the customer’s operations. There is no discussion of

customers in Hill or Warshaw, much less creating an event record with a customer-specific link to help address and resolve potential problems.

95. Still further, claim 18 calls for “consolidating the event record, correlated customer information and a symptom record, and linked problem resolution assistance information into a problem ticket.” Problem tickets are described at col. 7 ll. 24-37 as representing potentially security-related happenings on the customer’s network. They are customer-specific and therefore incorporate customer information as well as security intelligence and problem resolution information. The analyst working at the SOC can contact the customer to obtain additional information that might be needed from the customer. *Id.* at col. 10 ll. 49-52; col. 13 ll. 27-61 (describing the flow of Figure 8 in responding to an inbound call from a customer and relating it to a problem ticket to provide a customer-specific solution).

96. I see absolutely nothing in Hill or Warshaw, or the analysis at pages 63-66 of the Petition that addresses the customer-specific limitations that I have just discussed that are incorporated into claim 18. Consequently, those limitations, as well as the analysis of residue to identify a potential event about which to

transmit information to a security analyst system, are all not present in the asserted references.

97. As reflected in the Summary of the Invention, the preferred embodiment of the '641 patent describes an overall managed security monitoring service "that monitors a customer's network activity using a probe or 'sentry' system, collects status data from monitored components, filters or otherwise analyzes the collected data for activity possibly implicating security concerns, alerts and transmits information about such activity to trained security analysts working at secure operations centers ('SOCs')...." *Id.* at col. 1 ll. 59-67.

98. As is also reflected in the Summary of the Invention, the component parts of that service might include a probe/sentry system located at a customer site (col. 3 ll. 11-26), a gateway system to link the probe to the SOC (col. 3 ll. 27-29), and a SOC (col. 2 l. 65–col. 3 l. 9) where security analysts work (col. 2 ll. 42-50, ll. 65-67) and can access the SOCRATES problem and expertise management system (col. 3 ll. 42-45) which has access to multiple databases that provide security intelligence (col. 2 ll. 55-64).

99. There are many benefits to this architecture, as implemented by the inventors in a manner, which was unheard of in 2000, and has now come to be

widely employed. I start with the fact that the identification of the potential security event occurs from an examination of post-filtering residue comprised of status data that was neither selected by positive filtering nor discarded by negative filtering.

100. In this regard, there is an initial analysis at the probe in the customer network to determine whether there is a potentially security-related event. The initial analysis is performed, post-filtering, upon status data that is residue, and it is performed before any information is sent to the SOC. Considered to be of little interest by many, the inventors of the '641 patent recognized that it is in just such residue that clues about unknown intrusions are most likely to reside. This is the more effective and the right approach to identify a wider variety of previously undiscovered forms of malware and attack traffic than previously employed techniques such as pattern matching with variants of patterns of known malware and attacks.

101. The initial analysis at the probe cuts down considerably on the flow of traffic to the SOC by limiting that transmission to information about potential events flagged by the anomaly engine. Prior art approaches that transmit all traffic or all information extracted or derived from traffic would be comparatively slow

and expensive. Prior art approaches that leave residue discarded to reduce volume were unable to use that residue to uncover previously unknown intrusions.

102. Further, by conducting a second, confirmatory analysis at the SOC and providing a feedback mechanism to dynamically update the probe (see, e.g., col. 5 ll. 32-43) rather than relying on the probe to make the decision whether there has actually been a security event as traffic comes in, “false positives” are reduced. Still further, when a previously unknown attack is confirmed by the SOC, the probe can be updated with new information from the SOC to detect future occurrences of the same attack (now identified as a known attack).

103. Although analysis of post-filtering residue at the probe before transmission to the SOC makes the initial determination about whether there has been a potential event, a more detailed analysis of that event can be performed at the SOC, with all the tools available to the system and security analysts working there. *See id.* at col. 2 ll. 15-18 (“[P]articulars may then be examined and analyzed by trained security analysts continuously monitoring the customer’s network to further understand the incident and eliminate false positives.”). In this fashion, false positives, a significant problem at the time, are reduced.

VI. THE FILE WRAPPER HISTORY FOR THE ‘641 PATENT

104. I have reviewed the entire file wrapper history for the ‘641 patent as presented as Exhibit 1002, and the entire file wrapper history for the ‘237 patent, Exhibit 1008, of which the ‘641 patent is a continuation.

105. Ultimately, the same claim limitation that was added to the independent claims of the ‘237 patent and cited by the Examiner in her reasons for allowance was also added to all the independent claims of the ‘641 patent. Furthermore, in her stated reasons for allowing the ‘641 patent the Examiner reiterated what she had said about Hill in her stated reasons for allowing the ‘237 patent. All told, I see absolutely nothing wrong with the Examiner’s analysis, which suggests only that the Examiner had a far better understanding of the invention described and claimed by the inventors of the ‘641 patent than does Petitioner.

106. At pages 22-23 of the Petition, Petitioner claims that the Examiner’s analysis of Hill was neither complete nor accurate and focuses first on the following comment made by the Examiner as somehow manifesting some lack of appreciation of Hill by the Examiner:

Hill et al. systems is merely a system that simulates the attacks to determine how these attacks will effect or have the potential to effect the network.

Petition at 23 (quoting the Examiner's Reasons for Allowance).

107. This comment actually focuses one of the critical differences between the '641 patent and Hill. As the Examiner first notes in the same passage, "Hill does not disclose or suggest the claim limitation, wherein the post-filtering residue data is neither discarded not [sic] selected by filtering." *See id.* at 13. In this regard, the '641 patent filters status data and flags potential events through an analysis of status data so that information about those events can be sent to a SOC to perform a full analysis with the benefit of tools and databases at the analyst's disposal. There is a first level of analysis to flag potential security events before transmission to the SOC, where there is a second level of analysis to determine if the potential event is an actual event and/or address how to best resolve an event by invoking all the resources available to a security analyst at the SOC. *See, e.g.,* Exh. 1001, col. 10 l. 60–col. 11 l. 1 (discussing exemplary use of SOCRATES).

108. In Hill, there is a conventional inspection looking for intrusions at the wall (e.g., through a firewall), but the only subsequent analysis is one performed by processor 40, which compares patterns in traffic data to pre-existing signatures

created through training exercises without which the processor cannot identify anything. The identification of actual security events is performed only by comparison to signatures in memory generated through the simulated attacks, whereas the '641 invention has the capability to identify any unknown intrusion that can be flagged, initially, from status data as representing a potentially security-related event.

109. In short, when the Examiner states that Hill is merely a system that simulates attacks to determine how those attacks might affect the network, she is noting that the entire analysis of potential security events turns on a comparison to signatures created through simulated attacks. The '641 patent allows for the discovery of anything that might be understood from an analysis of status data, first by a probe, and then by a SOC at which databases, tools, and the judgment of an analyst are all available. This is a critical difference and the Examiner's having focused upon it hardly belies a lack of appreciation of other aspects of Hill.

110. Petitioner also claims that the architecture of Hill is virtually identical to that disclosed in the '641 patent and, continues on page 23 of the Petition with a claim that the Examiner failed to appreciate that Hill's methodology is a type of filtering in the eyes of a POSITA. In actuality, the Hill architecture is nothing like

that disclosed in the '641 patent and the Examiner's comments, as well as her decision to apply other references to the claims, only shows that she appreciated the differences.

111. There are at least three major differences in the architecture. The filtering subsystem described and claimed in the '641 patent, which examines residue and the post-filtering analysis, is completely missing from the Hill reference. The identification of potential events at the probe to reduce transmission volume to the SOC, where a second analysis takes place to reduce false positives is also not present in Hill. Third, the feedback mechanism to update the probe, as described and claimed in the '641 patent is also not present in Hill.

112. Seen for what it really is, Petitioner's claim that it would be evident to a POSITA to analyze residue status neither selected by positive filtering nor discarded by negative filtering is nothing more than a conclusory statement that a novel feature, neither disclosed nor suggested by any reference, would have been obvious to a POSITA. This bald conclusion lacks substance and lacks support. It also contradicts the findings of the Examiner as set forth in her stated reasons for allowance.

113. In short, in relation to Hill, the primary reference upon which Petitioner purports to rely, the Examiner was 100% correct when she stated that “Hill does not disclose or suggest the claim limitation, wherein the post-filtering residue is data neither discarded nor selected by filtering.” Exh. 1002, p. 19. I see absolutely nothing in Petitioner’s papers, including Dr. Reddy’s declaration, to suggest that the Examiner was in any way mistaken in this regard or any other aspect of her analysis which led to the issuance of the ‘641 patent.

114. I can confirm from my own familiarity with the state of the art as of the effective filing date of the ‘641 patent (March of 2000) that there had not been a lot of work on the architecture of an intrusion detection system suitable for use in a large enterprise. While that work does not feature a third category of unselected residue, as the Examiner correctly concluded, it does not surprise me that the Examiner considered the works of Esbensen and Porras et al. as the closest prior art (with respect to the parent patent application, which led to the ‘237 patent, over which the Examiner issued obviousness-type double patenting rejections on all claims except for the claims that eventually issued as claim 18 and its dependents). Exh. 1002, pp. 60-61, 105-06, 120-21. I can concur that Esbensen is at least as relevant as the Hill reference, if not more. In this last regard, I note that the

Examiner found every element of the claims that issued into the ‘237 patent to have been disclosed by Esbensen except for the post-filtering residue limitation and that the Applicant did not dispute this in its Remarks, choosing instead to focus on the post-filtering residue limitation. Exh. 1008, pp. 79-80, 125-26. In relation to that limitation, Porras does not disclose an analysis of residue, but looks at streamed alerts and was taken by the Examiner as disclosing positively and negatively filtering status data. Warshaw does not even filter status data.

VII. THE PRIOR ART

A. General Observations on the State of the Art at the Time of Invention, Including the Jansen Reference

115. I have reviewed the Jansen et al. article attached as Exhibit 1006 (“Jansen”) and cited by Petitioner as reflecting the state of the art in the relevant time-frame. Far from suggesting anything that might lead to the invention described in the ‘641 patent, Jansen characterizes the subject matter to which the ‘641 patent subsequently addresses itself as largely unexplored, futuristic technology that might be the subject of research down the road.

116. Specifically, Jansen notes that “[n]o major breakthroughs in intrusion detection have recently been made” (Exh. 1006, p. 1) and specifically states that

improvements in architectures for identifying intrusions are something that can be expected to be the subject of future research and one of several unexplored areas:

[F]uture intrusion detection research is expected to focus on relatively unexplored areas such as:

Architectures for highly distributed intrusion detection systems,

Id. at 2.

117. The bullet points Petitioner lists on pages 16-18 of its Petition are largely aspirational, in the nature of a wish-list much of which no one had actually been able to fulfill at that time.

118. For example, Jansen notes that current intrusion detection systems “are less than perfect” (Exh. 1006, p. 18), and specifically calls out the problem of false positives, stating that “[v]ery high ‘false positive’ alarm rates” have plagued conventional intrusion detection systems (*id.* at 22). Still further Jansen laments “existing response mechanisms are too weak to stop sophisticated attackers.” *Id.* at 25.

119. As developed above, the ‘641 patent addresses these very concerns such that, fairly read, Jansen can only be seen as evidence of invention in the ‘641 patent—not against it.

B. The Hill Reference

120. The Abstract of U.S. Patent No. 6,088,804, issued to Hill et al. attached as Exhibit 1004 (Hill), provides a succinct description of its network by referring to the elements of Figure 1, reproduced below, as follows:

A dynamic network security system (20) responds to a security attack (92) on a computer network (22) having a multiplicity of computer nodes (24). The security system (20) includes a plurality of security agents (36) that concurrently detect occurrences of security events (50) on associated computer nodes (24). A processor (40) processes the security events (50) that are received from the security agents (36) to form an attack signature (94) of the attack (92). A network status display (42) displays multi-dimensional attack status information representing the attack (92) in a two dimensional image to indicate the overall nature and severity of the attack (92). The network status display (42) also includes a list of recommended actions (112) for mitigating the attack. The security system (20) is adapted to respond to a subsequent attack

that has a subsequent signature most closely resembling the attack signature (94).

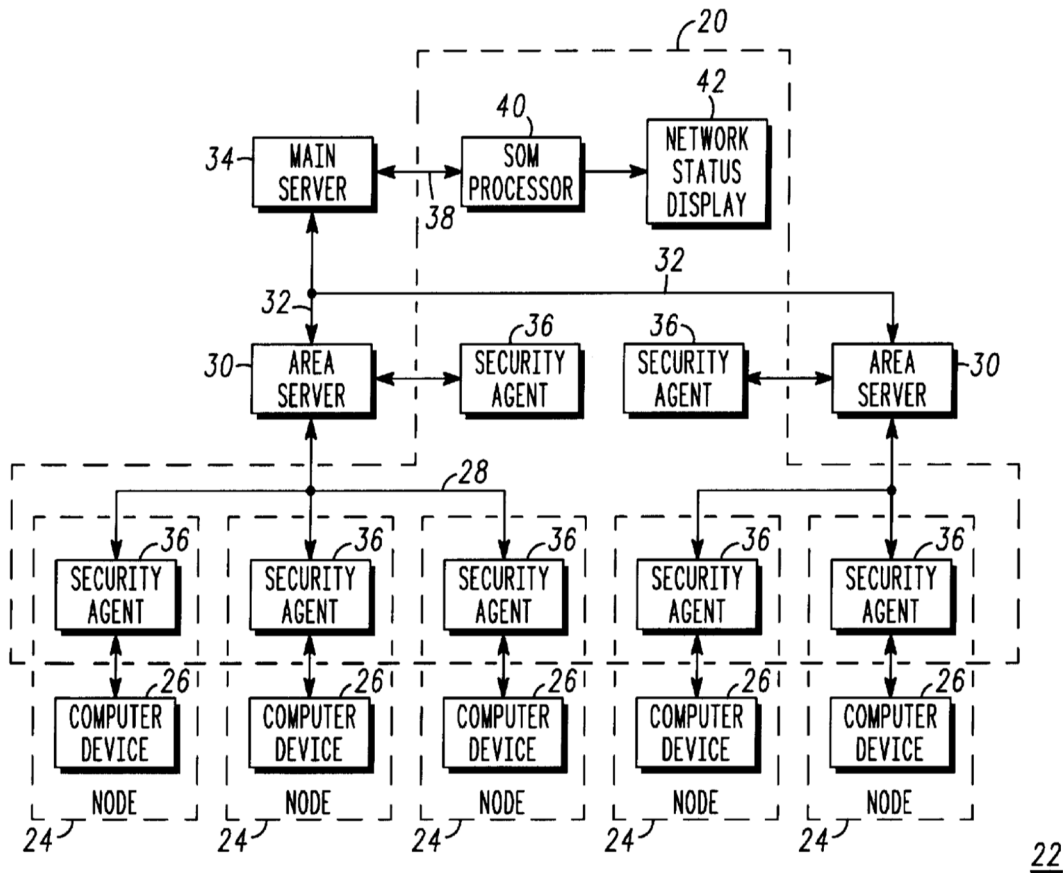


FIG. 1

121. Focusing on Figure 1, dashed line 20 defines the network security system. Security agents 36 detect occurrences on nodes 24, where a computer device 26 is located. Exh. 1004, col. 4 ll. 11-12, 31-38. Little is said about the manner in which security agents 36 operate. We are told that security agents are software programs that identify security events as they appear at the nodal level.

Id. at col. 4 ll. 35-41. There is no suggestion that security agents 36 operate in a fashion similar to that disclosed in the '641 patent. Instead, it is stated at col. 4 ll. 32-41 that:

Security agents 36 are configured to concurrently detect occurrences of security events (discussed below) on associated computer nodes 24. Security agents 36 are software programs located at nodes 24 and area servers 30 that identify security events as they appear at the nodal level. Security events may include port scans, malicious software, penetration attempts, and others that are identified through either a specific code signature or through actions or attempts at actions.

122. There is no discussion of status data and Petitioner presumes that Hill has something to do with it. However, all traffic data is not status data and there is insufficient information in Hill to determine whether Hill is matching unstructured data in some fashion or making use of status data.

123. The other descriptions are no more specific. For example, Hill states that the security system first addresses security events through security agents 36, which may be associated with conventional tools such as virus detectors and firewalls. *Id.* at col. 7 l. 64–col. 8 l. 5.

124. Whatever inferences one chooses to make about the undisclosed details of the methodology of Hill and what a person skilled in the art might infer from them, one thing is abundantly clear: the Hill reference neither discloses nor suggests a filtering subsystem that generates “residue” status data that has neither been “selected” nor “discarded.”

125. “Data about security events” is collected by the security agents 36 and transmitted to a processor 40, which, in a preferred embodiment, is a self-organizing map (“SOM”) processor which applies artificial neural network technology. Alternatively, the processor might be a linear vector quantization (“LVQ”) processor, which applies a category of artificial neural network technology. *Id.* at col. 4 ll. 53-61. In both instances, the processor employs artificial intelligence to match data to signatures to identify attacks.

126. The processor that determines whether there has been a security event in data collected by security agents 36 is element 40. That processor and display 42 may be incorporated into an existing network 22 (*id.* at col. 4 ll. 5-10), with the processor of the main server 34 handling the responsibilities of the SOM processor 40 (*id.* at col. 5 ll. 16-18).

127. Petitioner cannot find an analysis of “residue,” much less residue comprised of status data, in the operation of processor 40 for two reasons. First, it is not there. Second, the claims require an analysis of residue at the post-filtering stage, prior to transmission on for further analysis.

128. Pattern matching, as employed by processor 40, is a form of positive identification and if the pattern matching found near-hits as well as direct hits, both were selected through application of the pattern. Since near-hits represent data that has been selected, it cannot be considered “residue” (assuming that the pattern matching in fact involved status data).

129. Furthermore, analysis of residue takes place before transmission to the SOC according to the claim language in the ‘641 patent, which requires residue to be analyzed to identify potentially security-related events at the probe so that information related to those specific events can then be sent to processor 40. Since Petitioner maps the SOC of the ‘641 patent to processor 40 and display 42 of Hill, where information about an event is received, it cannot also map the same elements to the probe, where residue is first analyzed to decide what should be sent.

130. Processor 40 of Hill also requires training. *Id.* at col. 4 l. 62–col. 5 l. 6, ll. 23-25. Figure 2 of Hill shows a system training process, which involves generating simulated attacks against which the processor generates signatures indicative of an attack type that may occur on the network. *Id.* at col. 5 ll. 23-25, 39-45.

131. Types of attacks simulated to generate signatures against them include, e.g., viruses, worms, Trojan horses, FTP requests and network overload. *Id.* at col. 5 ll. 58-61. Types of attacks may also be rated according to their anticipated severity, e.g., by the number of nodes they are likely to impact. *Id.* at col. 6 ll. 9-22.

132. After the processor is primed to recognize attack types through the iterative generation of training signatures, the security system 20 is ready to respond. *Id.* at col. 7 ll. 55-57. In this same regard, Hill is only looking for data that matches signatures that have been created and stored to which the artificial intelligence in processor 40 is able to make a match.

133. A network status display 42 shows a map 66 that has types of attacks placed in proximity to one another and moves attacks that may have greater severity to the peripheral area of the map. *Id.* at col. 6 ll. 50-60. In this fashion,

the network manager can more easily see the type of attack and its severity by looking at the location and grouping of attacks on the display. *Id.* at col. 7 ll. 3-8. The network display 42 may ultimately provide a list of suggested actions calculated to mitigate the attacks. *Id.* at col. 8 ll. 59-61. The security console in Hill simply maps the conclusions at which processor 40 of Hill arrived by employing artificial intelligence.

134. After an attack, the processor is primed to predict a pattern of subsequent attacks. This process is set forth in Figure 5 of Hill and accomplished by utilizing the new attack signature as a training signature and repeating the training process. *Id.* at col. 9 ll. 34-40. Hill provides a mechanism for updating processor 40 with additional signatures using the process summarized by Petitioner at page 32 of the Petition, citing to Hill at col. 9 ll. 34-46. Simply stated, the attack signature becomes a new training signature, with which to update processor 40.

135. The update is confined to processor 40, and does not take place at any component which can be mapped to a probe as claimed in the '641 patent. There is no statement or suggestion that security agents 36 or area servers 30 are themselves updated to filter more efficiently as a result of the attack. The feedback comes in

the form of re-training the processor 40. There is no transmission to the components that perform the initial filtering at the nodal level.

136. Furthermore, the Hill architecture is limited to what a processor employing artificial intelligence can discern through comparison to pre-existing signatures. On the other hand, by flagging potential events from residue for further analysis at the SOC, the architecture of the '641 patent allows virtually any kind of unknown attack, including "zero day" attacks never before known to exist to be uncovered.

C. The Warsaw Reference

137. For the purposes of this declaration, I have been asked to assume that the Warsaw et al. reference, attached as Exhibit 1005 (Warsaw), qualifies as prior art to the '641 patent.

138. Warsaw is an article principally about VenusID. VenusID was developed "as an enhancement to the analysis layer of a two-layer distributed network intrusion detection system" with the two layers being identified as a network layer and an analysis layer. Exh. 1005, pp. 1, 2.

139. Filtering is performed by probes within the network layer, but the article barely describes what is done in this regard. Warsaw actually states that

the intent is to focus on the analysis layer and not to discuss the implementation of the network probes:

In this paper, we do not discuss the implementation of the network probes; we assume that the probes can be configured, automatically or manually, to recognize events needed by the analysis layer.

Id. at p. 3.

140. What discussion is provided of the network layer and the operation of the probe is very short. First, near the beginning of the introduction section, Warsaw states:

The two layers of this IDS consist of a network layer and an analysis layer. The network layer contains an array of protected sub-networks. A privileged computer on each sub-network, called a *probe*, sniffs all network traffic. Probes execute an initial set of filters that analyzes traffic, capturing suspicious packets while discarding obvious non-threatening connections. A synopsis of the results and the filtered subset of connections are forwarded to the analysis layer and logged to an Oracle database. At the analysis layer, an ad-hoc C program with embedded SQL periodically wakes and further analyzes the activities. This analysis searches for possible intrusion patterns and

raises alarms for the human network analysts if suspicious activity is detected.

Id. at p. 2 (emphasis in original).

141. No description is provided as to how this filtering is accomplished. Subsequent comments by Warshaw, however, indicate that the authors were not filtering status data. Specifically, in section 6.2.2, the authors raise data formatting issues and make the following observation:

Most intrusion detection systems log network traffic containing discrete measures (such as to and from IP addresses, the time of the connection, login destination, etc) and continuous measurements (such as CPU time, IO activity, etc) [5]. Unfortunately, encoding of the data transferred during a connection and the events recognized by probes are not nearly as informative. This information is often contained solely within ASCII. Thus, many of our rules must, at run-time, parse unstructured ASCII data to discover anomaly patterns. Consequently, this lack of structure severely limits the quality and extensibility of VenusIDS rules.

Our knowledge engineers are working on more effective strategies to structure event and connection information data.

Id. at p. 11.

142. This passage indicates that the authors were attempting to parse unstructured ASCII data because many of the rules applied during analysis must parse unstructured ASCII data. That form of ASCII, presented in an unstructured state that does not confer any meaning, is not “status data.” Consequently, the authors appear not to have filtered status data.

143. Warshaw does not espouse the use of status data. Instead, it speaks to the difficulty of parsing unstructured ASCII and has engineers looking for different, and potentially more effective strategies.

144. There is also one further discussion of the network layer and its interaction with the analysis layer in section 2, entitled “Application,” as it appears at page 3 of Warshaw.

145. After reiterating that there are two layers and depicting the architecture in Figure 1, the passage on page 3 continues as follows:

The network layer consists of an array of subnetworks each monitored by a probe. A probe sniffs, in real time, all network traffic on a subnetwork and filters the traffic by discarding obvious non-threatening traffic and classifying the remaining traffic as suspicious or uncertain. Suspicious traffic, also called *events*, is logged

in real-time to the analysis layer, while all non-discarded traffic is logged at predetermined intervals to the analysis layer.

Id. at 3 (emphasis in original).

146. Here, Warshaw confirms that there is no analysis of residue, or anything else, from which to flag potential security events after filtering and prior to transmission to the analysis layer. Instead of controlling the amount of information sent on for analysis by requiring potential events to be flagged before transmission and limiting transmission to those events, Warshaw sends everything that has not been discarded already. To deal with the volume of traffic created by sending everything, and to manage traffic flow, Warshaw sends the information at predetermined intervals rather than all at once.

147. In short, the passage of Warshaw copied above confirms that there is no analysis of the uncertain data (i.e., the putative “residue”), much less anything that could be called “status data” “to identify potentially security-related events” so that “information about the identified events” is transmitted to an analyst system.

148. Instead, in Warshaw, all uncertain data, as well as all suspicious data, is transmitted to the analysis layer. This is precisely what the ‘641 patent architecture avoids by requiring potential events to be identified from residue

before transmission for further analysis so that the information actually transmitted is information about “said identified events.” Since the anomaly engine 2050 of the ‘641 patent is only looking for anomalous events that have not already been positively selected, I would expect the amount of traffic flagged as involving potentially security-related events to be relatively small. Consequently, the methodology claimed in the ‘641 patent reduces the amount of traffic sent on for analysis by several orders of magnitude relative to Warsaw.

149. By sending all uncertain information on to the analysis layer and failing to analyze such data to identify potential events prior to sending, Warsaw may flood the communications lines and analysis layer with too much information. In fact, Warsaw notes that the first generation of the system was non-scalable since every anomaly discovered by a query was reported, so as to bombard the analyst with too much information. The solution in the second generation, however, was not to identify events from post-filtering residue and then send information about those events, rather than about everything. Instead, the solution in the second generation was to enhance the rule processing capabilities of the analysis layer to enable it to better handle everything considered suspicious or uncertain.

VIII. THE REFERENCES CITED BY PETITIONER DO NOT TEACH THE LIMITATIONS DIRECTED TO FILTERING RESIDUE, AS THE EXAMINER APPRECIATED IN RELATION TO HILL, AND ALSO DO NOT TEACH A DYNAMIC UPDATE THROUGH FEEDBACK TO THE PROBE

A. Neither Hill Nor Warshaw Suggest an Analysis of Residue to Flag Potential Events so That Information About Those Events Can Be Sent on for Analysis

150. The claims of the '641 patent require two levels of analysis, beginning with a first analysis performed upon post-filtering residue to identify potentially security-related events represented in that residue status data that is neither selected (by positive filtering) nor discarded (by negative filtering).

151. The claim language that requires this first level of analysis to take place through an examination of residue status data at the probe can be found in both independent claims—for example as element (b) of claim 1:

analyze status data to identify potentially security-related events represented in the status data, wherein the analysis includes filtering followed by an analysis of post-filtering residue, wherein the post-filtering residue is data neither discarded nor selected by filtering;

Exh. 1001 at col. 33 ll. 31-35. While independent claim 18 is directed to a method of operating a secure operations center (SOC) (per the preamble), the first level of

analysis is also described with respect to the creation of an “event record” identified through filtering and an analysis of post-filtering residue:

wherein the potentially security-related event is identified by filtering followed by an analysis of post-filtering residue, wherein the post-filtering residue is neither discarded nor selected by the filtering;

Id. at col. 34 ll. 25-30.

152. The limitation present in claim 1 relating to the subsequent transmission to the SOC (described in claim 18) of information concerning the potential event flagged from residue at the probe, can be found as element (c) of claim 1:

transmit information about the identified events to an analyst system associated with the security monitoring system;

Id. at col 33 ll. 36-38.

153. The second level of analysis occurs after information about potentially security-related events uncovered by the first level of analysis (“the identified events”) has been transmitted to the SOC for a more complete analysis. As depicted in Figure 1 of the ‘641 patent, information about the potential event identified earlier at the probe, is sent across communication pipes 3000, to the SOC

where sophisticated tools, databases, and a security analyst are all available to perform that second analysis.

154. Hill does not disclose the use of status data and does not filter anything into “residue” which is status data that has not been selected or discarded.

155. Warshaw also does not filter status data and, in fact, applies rules in its analysis that would not allow it to do so. Consequently, Warshaw can hardly be taken as a suggestion to filter status data at all.

156. Further still, Warshaw sends everything that is uncertain as well as suspicious onwards from the probe to the analysis layer. Whereas the ‘641 patent addresses the issue of traffic volume by limiting transmission of information for analysis to potential security events identified at the probe, Warshaw addresses the issue of traffic volume by transmitting everything, but transmitting it piecemeal, at separate intervals.

157. However generously interpreted by Petitioner to make its case, Hill does not disclose or suggest a first analysis of residue status data to identify potential events followed by transmission of information about those identified events to the SOC for a second level of analysis.

158. However generously read by Petitioner to make its case, Warshaw also does not disclose or suggest a first analysis of residue status data to identify potential events followed by transmission of information about those identified events to the SOC for a second level of analysis.

159. Consequently, Hill does not identify any potential events at a probe so that information about those identified potential events, as required by claims, can be sent off for further analysis. Instead, Hill sends everything to processor 40, identified by Petitioner (along with display 42) as corresponding to the SOC.

160. In short, Hill and Warshaw cannot be combined to fulfill each and every limitation of each independent claim relating to the filtering subsystem present in the '641 patent. Simply put, all claims of the '641 patent require the identification of potentially security-related events based on an analysis of residue followed by the transmission of information about "the identified events." The cited references cannot be combined to achieve this.

B. The Claim Limitations Directed to Updating the Probe are Also Missing from Hill, By Itself or as Combined with

Warshaw

161. As stated in the Abstract of the ‘641 patent, feedback from problem resolution efforts can be used “to update the filtering and analysis capabilities of the probe and other systems.” Exh. 1001, Abstract.

162. Claim 1 of the ‘641 patent (and its dependents) requires the receipt of feedback at the probe based on empirically derived information reflecting operation of the security monitoring system (e.g., the analysis performed at the SOC). The claims also require the dynamic modification of an analysis capability of the probe, during operation, based on the received feedback. In this regard, I start with the claim language present in the ‘641 patent.

163. Elements (d) and (e) of claim 1 read as follows:

- d) a receiver for receiving feedback at the probe based on empirically-derived information reflecting operation of the security monitoring system; and
- e) a modification control system for dynamically modifying an analysis capability of the probe during operation thereof based on the received feedback.

Id. at col. 33 ll. 39-45.

164. The meaning of “probe” in the ‘641 patent is spelled out in the Summary of Invention at col. 3 ll. 11-26. The probe collects data from software

and/or hardware products. *Id.* at col. 3 ll. 11-14. The probe then filters status data, analyzes it, and sends noteworthy information in the form of sentry messages to the gateway that leads to the SOC. *Id.* at col. 3 ll. 17-21. The SOC itself is not part of the probe.

165. In the preferred embodiment, the probe is shown in Figure 2 of the ‘641 patent, and is described as software and/or hardware that can monitor components that provide status data, which it collects, filters and analyzes. *Id.* at col. 4 ll. 64–col. 5 ll. 3; col. 5 ll. 25-29. The probe then formats noteworthy data, indicative of a potentially security-related event into sentry messages, which are sent on for further analysis. *Id.* at col. 5 ll. 44-49.

166. The architecture described in the ‘641 patent performs two levels of analysis, the first at the probe to flag potential events and the second at the SOC to determine if there was in fact a security event, where it can also be determined how best to address it.

167. After the analysis at the SOC, updates are sent from the SOC to the probe where those updates are installed. *Id.* at col. 5 ll. 32-37. In the preferred embodiment, the probe to the left in Figure 1 is not the same as the SOC, shown to the right in Figure 1 (shown in detail in Figure 2). Rather, the probe receives

feedback from the SOC through which the probe can update itself with the fruits of the second, deeper level of analysis performed at the SOC. This allows the probe to filter more effectively in the future.

168. The components corresponding most closely to the claimed probe in the Hill architecture, which identify security events as they appear at the nodal level, before transmission for further, deeper analysis is the software contained within security agents 36, which interact with area servers 30. This is spelled out in Hill:

Dynamic network security system 20 includes a plurality of security agents 36 each of which is associated with one or more nodes 24. Security agents 36 are configured to concurrently detect occurrences of security events (discussed below) on associated computer nodes 24. Security agents 36 are software programs located at nodes 24 and area servers 30 that identify security events as they appear at the nodal level. Security events may include port scans, malicious software, penetration attempts, and others that are identified through either a specific code signature or through actions or attempts at actions.

Exh. 1004, col. 4 ll. 30-41.

169. Hill goes on to state that data about security events so collected is sent to a processor 40:

Data about security events is collected by security agents 36 and transmitted via links 28, links 32, and a communication link 38 to a processor 40.

Id. at col. 4 ll. 53-55.

170. What Hill never states is that updates are sent from processor 40 back to the “probe,” which looks for security events at the nodal level through security agents 36 coupled to area servers 30.

171. In fact, Hill updates processor 40, and only processor 40, by using the attack signature generated by the security event as a new training signature with which to update the security system by repeating the training process. As stated in Hill:

Following task 114, a task 116 predicts a pattern for subsequent attacks and adapts security system 20 to respond to subsequent attacks. Security system 20 is adapted by introducing first attack signature 94 into security system 20 as a new training signature and repeating training process 44 (FIG. 2). The result of introducing first attack signature 94 as a new training signature, and mapping a vector representative of the new

training signature into display map 66, is an improved response to subsequent attacks that have subsequent training signatures that most closely resemble first attack signature 94. Thus, security system 20 is able to evolve with evolving threats.

Id. at col. 9 ll. 35-45.

172. The components closest to corresponding to the claimed probe that scan at the nodal level in Hill, the security agents 36 and the area servers 30 are not updated at all. There is no dynamic feedback to the probe to update its capabilities as required by claim 1 of the ‘641 patent.

173. To attempt to finesse the operation of Hill into something similar to what is claimed in the ‘641 patent, Petitioner and Dr. Reddy decide, arbitrarily, to equate processor 40 with part of the probe as well as with part of the SOC. There is, however, no principled basis for doing so, leaving Petitioner’s attempt to find the claimed dynamic feedback to the probe entirely contrived.

174. We are first told that processor 40, area server 30 and main server 34 of Hill, together, are the probe. *See, e.g.*, Petition at 45-46 (discussing claim 1 element (d)) (“The receiver is located at the processor 40 for introducing the first attack signature as a new training signature as part of the training process for

processor 40 show in FIG 2.”). Then we are told that the secure operations center of Hill is the network display 42 in cooperation with processor 40. *See, e.g.*, Petition at 55-56 (discussing claim 10); *see also id.* at 62 (discussing claim 18). As can be seen, processor 40, alone or incorporated in main server 34, cannot correspond to the probe.

175. The structure that performs the first level analysis at the nodal level and sends information about events on for further analysis, i.e., that corresponds at a very high level to functions performed by the probe of the ‘641 patent, is neither processor 40, which allegedly corresponds to the SOC as well, nor main server 34. They are security agents 36 and area servers 30, neither of which receive dynamic feedback.

176. Dr. Reddy himself effectively admits as much. In the ‘641 patent the probe collects information about events, after performing a first analysis, so that information about the events can be sent along for further analysis. In Hill, Dr. Reddy acknowledges that it is security agents 36 and area servers 30 that provide the analogous functions, not processor 40, which is the component to which information is sent by the probe (whether or not processor 40 is integrated into main server 34):

Hill discloses that attack status information regarding security events (from security agents 36) are combined at area servers 30 in network 22 by security agents 36 associated with area servers 30. The attack information is then combined and transmitted up [sic] computer network 22 to form an attack signature.

Exh. 1003 ¶79.

177. Subsequently, Dr. Reddy maps the receipt of feedback to processor 40 of Hill. *Id.* at ¶116-117. While he speaks in terms of security system 20 as having been dynamically updated, Dr. Reddy never says that the parts of the system that actually correspond to the probe, elements 36 and 30, are ever updated in the manner required by the claims. In short, Dr. Reddy forces the argument to match the claims by treating processor 40 as part of the probe so there is feedback to something he can call the probe.

178. Since the only component that is updated (by using the signature generated during an actual attack as a new training signature) is also processor 40, Dr. Reddy and Petitioner must necessarily identify processor 40 as contained within both the probe and the SOC. However, it cannot be part of both and, for reasons already stated, is clearly not part of the probe.

C. Hill and Warshaw Both Fail to Discuss Customer Information

as Required by Independent Claim 18

179. Claim 18 is directed to a “method of operating a secure operations center as part of a security monitoring system for a customer computer network.” Exh. 1001, col. 34 ll. 22-24. Claim 18 (and its dependents) incorporates customer-specific aspects into the operation of the SOC. First, claim 18 recites the creation of an event record, generated within the SOC. *Id.* at col. 34 ll. 25-30; *see also id.* at col. 10 ll. 12-15. Next, claim 18 requires correlating the event record with customer information and a symptom record. *Id.* at col. 34 ll. 31-32. Then, the SOC uses the correlated customer information to link the event record to problem resolution information resident in databases accessible at the SOC. *Id.* at col. 34 ll. 33-34; *see also id.* at col 10. ll. 15-28.

180. The architecture of the ‘641 patent expressly considers customer information, such as customer-specific history or information about customers’ operations in addressing and resolving potential problems. According to Petitioner, the location identifiers 60 of Hill correspond to the customer-specific aspects of the claimed operation of the SOC. The location identifiers 60, however, at most identify the location of nodes under attack. In fact, neither Hill nor Warsaw, taken alone or in any combination, discuss customers, much less

creating an event record with a customer-specific link to help address and resolve potential problems as required by claim 18.

IX. IT IS NOT REASONABLE TO MODIFY HILL WITH WARSHAW TO INCORPORATE A DIFFERENT FILTERING SUBSYSTEM OR FEEDBACK TO THE PROBE.

181. There is no reason why a skilled artisan would be motivated to combine features of Warshaw into Hill to arrive at anything that might more closely resemble the system disclosed and claimed in the '641 patent.

182. Hill employs artificial intelligence in a processor, trained to recognize attack signatures in traffic. It is not directed to analyzing previously unknown attacks and has no way to identify them. The system of Hill has to be trained to operate. A first level of analysis that flags potential events in residue status data goes nowhere. There is no way to conduct a second, deeper level of analysis in Hill because there is no security analyst with tools and databases at his disposal. There is just a processor that can only see what it has been tuned to see in the way of signatures.

183. Furthermore, Warshaw discloses virtually nothing about the probe, which is the component of Hill Petitioner seeks to modify. There is no guidance for how to modify Hill with Warshaw so as to create an expectation of success.

Warshaw does not filter status data, does not advocate the use of filtering data, and does not provide nearly enough detail as to what it is doing, much less how, to suggest to anyone a mechanism for modifying Hill with any expectation of improvement or success.

184. Further still, what little Warshaw has to say would not encourage anyone to filter status data. Warshaw employs rules in its analysis level that can only look at patterns in unstructured ASCII, not status data, as that term is employed in the '641 patent.

185. In terms of feedback, neither reference discloses nor suggests a dynamic feedback mechanism, which is directed to updating the probe. Warshaw states quite pointedly that it is not directed to a discussion of the operation of the probe and barely discusses it. Hill updates processor 40; it does not update what most closely corresponds to the probe. There is no way to combine these references to arrive at this aspect of the claimed invention.

186. Finally, Hill and Warshaw say nothing about monitoring customer installations and gives no consideration to identifying security events and remedies in terms of what is happening within the networks of specific customers. Consequently, there is nothing in Hill or Warshaw that can possibly fulfill the

limitations of independent claim 18 that call for a correlation of an event record with customer information or a problem ticket into which correlated customer information has been incorporated. Hill states that the nodes can be geographically dispersed; there is no suggestion that they belong to different customers or that the Hill system (e.g., processor 40) breaks out data by customer in making its analysis.

187. In this last regard, Hill's architecture is directed to an integrated network, nodes of which can be geographically removed from one another and the network security system monitoring them. Exh. 1004, col. 4 ll. 23-28. In such a case, "remotely related nodes 24 may still be related closely to one another in the hierarchical order of network 22." *Id.* at col. 4 ll. 25-30.

X. CONCLUSION

188. To summarize, the Hill architecture differs from that described in the '641 patent in two very fundamental respects and fails to map to the claims of the '641 patent in two entirely different ways, with or without Warshaw, and/or Jansen being taken into consideration.

189. First, the probe of the '641 patent filters status data into three categories, which Hill does not; creates a category of residue status data, which Hill does not; and identifies potential security events at the probe based on an

analysis of the residue status data, prior to transmission for further analysis, which Hill does not. Hill attempts to match all traffic against signatures loaded into processor 40 through artificial intelligence training exercises. Consequently, the claim limitations directed to the analysis of post-filtering residue to flag potential events and transmit information about those potentially security-related events on for further analysis cannot be found in Hill. Warshaw cannot cure this deficiency because it disparages the use of status data and never suggests analyzing residue to flag events, prior to transmission for further analysis, so that transmission can be focused on those identified events. Warshaw transmits for subsequent analysis all data remaining after discarding just the non-threatening traffic.

190. Second, the only element in Hill that is ever updated is processor 40. Processor 40, however, is not part of a probe. It is the portion of the system to which the components most closely corresponding to the claimed probe send information about events for further analysis. Consequently, the claim limitations directed to dynamically updating the system through feedback to the probe are not disclosed or suggested by Hill, either, alone or as combined with Warshaw.

191. While effectively moot, in view of the ineffectiveness of combining, there is no reason to combine Hill with Warshaw. Warshaw discloses virtually

nothing about the probe, which is the component of Hill that Petitioner seeks to modify. More particularly, Warshaw does not filter status data, does not advocate the use of filtering data and does not provide nearly enough detail as to what it is doing, much less how, to suggest to anyone a mechanism for modifying Hill with any expectation of improvement or success. If anything, what little Warshaw has to say would discourage a POSITA from filtering status data. Furthermore, Hill is pattern-matching to known signatures created through training exercises and has no use for status data. Hill is not looking for forms of intrusion that may be entirely unknown and might first be flagged through an analysis of residue status data.

192. Lastly, neither Hill nor Warshaw concern themselves with servicing customers and would have no reason to do so as required by claim 18 and its dependents.

I declare under penalty of perjury under the laws of the United States of America that all statements made herein based on my own personal knowledge are true and correct, that statements made on the basis of information and belief are believed to be true, and the opinions expressed in this declaration are my own. If called to testify as to the contents of this declaration, I could and would do so, competently.

These statements are made with knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under section 1001 of title 18 of the United States Code.

Dated: October 10, 2019

A handwritten signature in cursive script, appearing to read "Wenke Lee", followed by a horizontal line.

Wenke Lee

EXHIBIT A

Wenke Lee
Professor and John P. Imlay Jr. Chair
School of Computer Science
College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332-0280
wenke@cc.gatech.edu
<http://www.cc.gatech.edu/~wenke/>

Contents

EDUCATIONAL BACKGROUND	2
EMPLOYMENT HISTORY	2
CURRENT FIELDS OF INTEREST	2
I. TEACHING	4
A. Courses Taught	4
B. Curriculum Development	5
C. Individual Student Guidance	7
II. RESEARCH AND CREATIVE SCHOLARSHIP	9
A. Theses	9
B. Published Journal Papers	9
C. Published Books and Parts of Books	10
D. Edited Proceedings and Books	10
E. Conference Presentations (refereed)	10
E.1 with Proceedings (refereed)	10
F. Workshop Presentations (refereed)	23
F.1 with Proceedings	23
G. Other	24
H. Research Proposals and Grants	26
I. Research Honors and Awards	32
III. SERVICE	33
A. Professional Activities	33
A.1 Membership and Activities in Professional Societies	33
A.2 Journal Edit-oral Board Activities	33
A.3 Conference Committee Activities	33
B. On-campus Georgia Tech Committees	37
IV. NATIONAL AND INTERNATIONAL PROFESSIONAL RECOGNITION	38
A. Patents	38
V. OTHER CONTRIBUTIONS	39
A. Seminar Presentations	39

EDUCATIONAL BACKGROUND

Degree	Year	University	Field
Ph.D.	1999	Columbia University	Computer Science
M.S.	1990	The City College of New York	Computer Science
B.S.	1988	Sun Yat-Sen University, China	Computer Science

EMPLOYMENT HISTORY

Title	Organization	Years
John P. Imlay Jr. Chair	College of Computing	3/2016–present
Co-Executive Director	Institute for Information Security & Privacy (IISP)	7/2015–present
	Georgia Institute of Technology	
Director	Georgia Tech Information Security Center (GTISC)	8/2012–6/2015
	Georgia Institute of Technology	
Professor	College of Computing	2/2009–present
	Georgia Institute of Technology	
Associate Professor	College of Computing	2/2005–2/2009
	Georgia Institute of Technology	
Assistant Professor	College of Computing	8/2001–2/2005
	Georgia Institute of Technology	
Assistant Professor	Department of Computer Science	8/1999–8/2001
	North Carolina State University	
Research Assistant	Department of Computer Science	1994–1999
	Columbia University	
Research Staff Member	IBM T.J. Watson Research Center	5/1997–8/1997
Member of Technical Staff	AT&T Research	5/1996–8/1996
Senior Software Analyst	Intergraph Corporation	1991–1994
Software Analyst	Intergraph Corporation	1990–1991

CURRENT FIELDS OF INTEREST

Information Security

Main research thrusts are: “Malware analysis and detection” – developing “live” and “safe” environment to uncover malware behaviors and machine-learning based techniques to automatically generate detection models and forensic analysis rules; “Mobile Platform Security and Privacy” – analyzing security vulnerabilities of mobile operating systems such as iOS and Android, and developing new approaches to protect data integrity and confidentiality; “Information Manipulation on the Internet” – studying how information services can be abused or attacked to deliver misleading information to harm people, and developing tools to detect and mitigate these new attacks; “Botnet Detection and Response” – developing detection algorithms to identify the command and control structures as well as the members of botnets, and response techniques to disrupt botnets; “Virtual

Machine Monitoring” – developing both active and passive monitoring techniques for applications and operating systems in virtual machines. “Adversarial Machine Learning” – studying the vulnerabilities of machine learning based systems due to adversarial manipulations and developing countermeasures.

I. TEACHING

A. Courses Taught

<u>Semester/Year</u>	<u>Course</u>	<u>Number of Students</u>	<u>Comments</u>
Spring 2019	OMS Cybersecurity 6262 Network Security	5	At CoC/On-line
Spring 2019	OMS Cybersecurity 6035 Introduction to Information Security	188	At CoC/On-line
Spring 2019	OMSCS 6262 Network Security	270	At CoC/On-line
Spring 2019	OMSCS 6035 Introduction to Information Security	750	At CoC/On-line
Fall 2018	OMSCS 6262 Network Security	154	At CoC/On-line
Fall 2018	OMSCS 6035 Introduction to Information Security	525	At CoC/On-line
Fall 2018	CS 6262 Network Security	47	At CoC
Summer 2018	OMSCS 6262 Network Security	191	At CoC/On-line
Summer 2018	OMSCS 6035 Introduction to Information Security	338	At CoC/On-line
Spring 2018	OMSCS 6262 Network Security	220	At CoC/On-line
Spring 2018	OMSCS 6035 Introduction to Information Security	428	At CoC/On-line
Fall 2017	OMSCS 6262 Network Security	300	At CoC/On-line
Fall 2017	OMSCS 6035 Introduction to Information Security	350	At CoC/On-line
Fall 2017	CS 4235/6035 Introduction to Information Security	120	At CoC
Summer 2017	OMSCS 6262 Network Security	300	At CoC/On-line
Summer 2017	OMSCS 6035 Introduction to Information Security	350	At CoC/On-line
Spring 2017	OMSCS 6262 Network Security	300	At CoC/On-line
Spring 2017	OMSCS 6035 Introduction to Information Security	350	At CoC/On-line
Fall 2016	OMSCS 6262 Network Security	300	At CoC/On-line
Fall 2016	OMSCS 6035 Introduction to Information Security	350	At CoC/On-line
Fall 2016	CS 4235/6035 Introduction to Information Security	80	At CoC
Summer 2016	OMSCS 6035 Introduction to Information Security	350	At CoC/On-line
Spring 2016	OMSCS 6035 Introduction to Information Security	350	At CoC/On-line
Fall 2015	OMSCS 6035 Introduction to Information Security	230	At CoC/On-line
Fall 2015	CS 4235/6035 Introduction to Information Security	80	At CoC
Fall 2014	CS 4235/6035 Introduction to Information Security	80	At CoC
Spring 2013	CS 4235 Introduction to Information Security	50	At CoC
Fall 2012	CS 4235 Introduction to Information Security	41	At CoC
Spring 2012	CS 6262 Network Security	35	At CoC

<u>Semester/Year</u>	<u>Course</u>	<u>Number of Students</u>	<u>Comments</u>
Fall 2010	CS 3210 Operating System Design	25	At CoC
Spring 2010	CS 6262 Network Security	45	At CoC
Fall 2009	CS 4237 Computer and Network Security	15	At CoC
Spring 2009	CS 4237 Computer and Network Security	15	At CoC
Fall 2008	CS 6262 Network Security	55	At CoC
Spring 2008	CS 8803ANS Advanced Systems and Network Security	25	At CoC
Fall 2007	CS 4237 Computer and Network Security	20	At CoC
Spring 2007	CS 6262 Network Security	45	At CoC
Fall 2006	CS 6265 Information Security Lab (w/ Mustaque)	18	At CoC
Fall 2006	CS 4803 Computer and Network Security	30	At CoC
Spring 2006	CS 6262 Network Security	45	At CoC
Fall 2005	CS 4803 Computer and Network Security	30	At CoC
Spring 2005	CS 6262 Network Security	30	At CoC
Fall 2004	CS 4803K Computer and Network Security	30	At CoC
Spring 2004	CS 6262 Network Security	30	At CoC
Fall 2003	CS 4803K Computer and Network Security	35	At CoC
Spring 2003	CS 8803K Advanced Systems and Network Security	16	At CoC
Fall 2002	CS 6262 Network Security	28	At CoC
Spring 2002	CS 6262 Network Security	30	New at CoC
Fall 2000	CSC 591Q Special Topics: Network Security	65	At NC State
Fall 1999	CSC 591Q Special Topics: Network Security	45	At NC State
Summer 1998	CS 3139 Data Structures and Algorithms	20	At Columbia

B. Curriculum Development

OMSCS 6262: Network Security. Graduate course in network security with topics including large-scale attacks and impacts, penetration testing and security assessments, security of Internet protocols (IP, TCP, DNS, and BGP), advanced web security, advanced malware analysis, advanced network monitoring, Internet-scale threat analysis, bitcoins and crypto-currencies, big data and security, cloud security, and attack-tolerant systems. Completely revamped in 2016 with new course materials drawing from research papers and projects developed by my Ph.D. students. Recorded videos for on-line offering via Udacity as part of OMSCS.

OMSCS 4235/6035: Introduction to Information Security. Cross-listed undergraduate and graduate introductory course in information security. It teaches the basic concepts, principles, and fundamental approaches to secure computers and networks. Its main topics include: security basics, security management and risk assessment, software security, operating systems security, database security, cryptography algorithms and protocols, network authentication and secure network applications, malware, network threats and defenses, web security, mobile security, legal and ethical issues, and privacy. Completely revamped in 2015 with new materials covering the up-to-date threat models, attack methods, and new technologies and policy considerations. Recorded videos for on-line offering

via Udacity as part of OMSCS.

CS 6262: Network Security. Graduate course focusing on the fundamental principles as well as advanced techniques in network security. This course covers cryptography and its application to network and operating system security; authentication; security for electronic mail; Java security; Firewalls and intrusion detection.

CS 4237: Computer and Network Security. New undergraduate course (first offered as CS 4803 in Fall 2003) focusing on the fundamental principles and techniques in computer and network security.

CS 8803: Advanced Systems and Network Security. Graduate course on advanced topics such as network and host-based intrusion detection, botnet detection, software security, malware analysis, and virtual machine monitoring. Students are required to study research papers and work on research projects.

C. Individual Student Guidance

1. Postdoctoral Fellows Supervised

Hong Hu: 2017-2019; Research Scientist in SCS at Georgia Tech

Sangho Lee: 2015-2018; Researcher in Microsoft Research (MSR) Redmond

Tielei Wang: 2011 - 2014; Start-up in China

Simon Pak Ho Chung: 2011 -2014; Research Scientist in SCS at Georgia Tech

Roberto Perdisci: 2009 - 2010; Associate Professor with tenure in the Department of Computer Science, University of Georgia

Daniel Xiapu Luo: 2008 - 2010; Associate Professor with tenure at Hong Kong Polytechnic University

2. Ph.D. Students Supervised

Xinzhou Qin: Graduated in Summer 2005; Juniper Networks

Yian Huang: Graduated in Summer 2006; Google

Prahlad Fogla: Graduated in Summer 2007; Google

Guofei Gu: Graduated in Summer 2008; Associate Professor with tenure in the Department of Computer Science, Texas A&M University

Bryan Payne: Graduated in Summer 2010; Director of Product & Application Security, Netflix

Monirul Sharif: Graduated in Fall 2010; Staff Software Engineer and Tech Lead Manager, Google

Kapil Singh: Graduated in Summer 2011; Research Staff Member, IBM T.J. Watson Research Center

Manos Antonakakis: Graduated in Spring 2012; Associate Professor with tenure in ECE at Georgia Tech

Martim Carbone: Graduated in Summer 2012; Staff Engineer, VMware

Junjie Zhang: Graduated in Summer 2012; Associate Professor with tenure in the Department of Computer Science and Engineering, Wright State University

Long Lu: Graduated in Summer 2013; tenure-track Assistant Professor in the College of Computer and Information Science, Northeastern University

Brendan Dolan-Gavitt: Graduated in Summer 2014; tenure-track Assistant Professor in the Department of Computer Science and Engineering, NYU Tandon School of Engineering (Polytechnic Institute)

Yacin Nadji: Graduated in Summer 2015; SecurityScorecard

Xinyu Xing: Graduated in Summer 2015; tenure-track Assistant Professor of Information Sciences and Technology, Penn State University

Chengyu Song: Graduated in Summer 2016; tenure-track Assistant Professor in the Department of Computer Science and Engineering, UC Riverside

Byoungyoung Lee: Graduated in Summer 2016; tenure-track Assistant Professor in the Department of Computer Science, Purdue University

Yeongjin Jang: Graduated in Summer 2017; tenure-track Assistant Professor in the Department of Electrical Engineering and Computer Science, Oregon State University

Kangjie Lu: Graduated in Summer 2017; tenure-track Assistant Professor in the Department of Computer Science and Engineering, University of Minnesota

Wei Meng: Graduated in Summer 2017; tenure-track Assistant Professor in the Department of Computer Science and Engineering, The Chinese University of Hong Kong

Yizheng Chen: Graduated in Fall 2017; Research Scientist at Baidu Labs

David Dagon: co-founded Damballa, Inc. in 2006; Research Scientist in ECE at Georgia Tech

Ruian Duan: In Progress

Evan Downing: In Progress

Erkam Uzun: In Progress

Yang Ji: In Progress

Chenxiong Qian: In Progress

Kyuhong Park: In Progress

Carter Yagemann: In Progress

Joey Allen: In Progress

Matthew Landen: In Progress

Burak Sahin: In Progress

Sena Sahin: In Progress

II. RESEARCH AND CREATIVE SCHOLARSHIP

A. Theses

Ph.D. Thesis

Title: A Data Mining Framework for Constructing Features and Models for Intrusion Detection Systems.

Date Completed: June 1999

Advisor: Professor Salvatore J. Stolfo.

University: Columbia University

B. Published Journal Papers (refereed)

1. Kangjie Lu, Meng Xu, Chengyu Song, Taesoo Kim, and Wenke Lee. Stopping Memory Disclosures via Diversification and Replicated Execution. *IEEE Transactions on Dependable and Secure Computing (TDSC)*, October 2018.
2. Junjie Zhang, Roberto Perdisci, Wenke Lee, Unum Sarfraz, and Xiapu Luo. Building a Scalable System for Stealthy P2P-Botnet Detection. *IEEE Transactions on Information Forensics and Security*, January 2014.
3. Roberto Perdisci, Davide Ariu, Prahlad Fogla, Giorgio Giacinto, and Wenke Lee. McPAD: A Multiple Classifier System for Accurate Payload-Based Anomaly Detection. *Computer Networks*, Vol. 53, 2009.
4. Roberto Perdisci, Andrea Lanzi, and Wenke Lee. Classification of Packed Executables for Accurate Computer Virus Detection. *Pattern Recognition Letters* Vol. 29, No. 14 (October 2008).
5. Prahlad Fogla and Wenke Lee. q-Gram Matching Using Tree Models. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, No. 4 (April 2006).
6. W. Fan, M. Miller, S. Stolfo, Wenke Lee, and P. Chan. Using Artificial Anomalies to Detect Unknown and Known Network Intrusions. *Knowledge and Information Systems*, Vol. 6, No. 5 (September 2004), Springer.
7. Yongguang Zhang, Wenke Lee, and Yian Huang. Intrusion Detection Techniques for Mobile Wireless Networks. *ACM/Kluwer Wireless Networks Journal (ACM WINET)*, 9(5), September 2003.
8. Joao B.D. Cabrera, Lundy Lewis, Xinzhou Qin, Wenke Lee, and Raman K. Mehra. Proactive Intrusion Detection and Distributed Denial of Service Attacks - A Case Study in Security Management. *Journal of Network and Systems Management*. 10(2), June 2002.
9. Wenke Lee, Wei Fan, Matt Miller, Sal Stolfo, and Erez Zadok. Toward Cost-Sensitive Modeling for Intrusion Detection and Response. *Journal of Computer Security*, 10(1,2), 2002.
10. Wenke Lee and Sal Stolfo. A Framework for Constructing Features and Models for Intrusion Detection Systems. *ACM Transactions on Information and System Security*, 3(4), November 2000.

11. Wenke Lee, Sal Stolfo, and Kui Mok. Adaptive Intrusion Detection: A Data Mining Approach. *Artificial Intelligence Review*, Kluwer Academic Publishers, 14(6):533-567, December 2000.
12. Wenke Lee and Gail E. Kaiser. Interfacing Oz with the PCTE OMS: A Case Study of Integrating a Legacy System with a Standard Object Management System. *Journal of Systems Integration*, Kluwer Academic Publishers, 9(4):329-358, December 1999.

C. Published Books and Parts of Books

1. Xinzhou Qin and Wenke Lee. Discovering Novel Attack Strategies from INFOSEC Alerts. *Data Warehousing and Data Mining Techniques for Cyber Security*. Anoop Singhal (eds), Springer, 2007.
2. Yongguang Zhang and Wenke Lee. Security in Mobile Ad-Hoc Networks. *Ad Hoc Networks: Technologies and Protocols*. P. Mohapatra and S. Krishnamurthy (eds), Springer, 2004.
3. Xinzhou Qin, Wenke Lee, Lundy Lewis, Joao B.D. Cabrera. Using MIB II Variables for Network Intrusion Detection. *Applications of Data Mining in Computer Security*. D. Barbara and S. Jajodia (eds), Kluwer Academic Publishers, May 2002.
4. Joao B.D. Cabrera, Lundy Lewis, Xinzhou Qin, Wenke Lee, Raman K. Mehra. Proactive Intrusion Detection - A Study on Temporal Data Mining. *Applications of Data Mining in Computer Security*. D. Barbara and S. Jajodia (eds), Kluwer Academic Publishers, May 2002.
5. Wenke Lee, Sal Stolfo, and Kui Mok. Algorithms for Mining System Audit Data. *Data Mining, Rough Sets, and Granular Computing*. T. Y. Lin, Y. Y. Yao, and L. A. Zadeh (eds), Physica-Verlag, 2002.
6. Wenke Lee and Naser Barghouti. Jadve: An Extensible Data Visualization Environment. In *Object-Oriented Applications Frameworks*, M. Fayad, D. Schmidt, and R. Johnson (eds), John Wiley & Sons, 1999.

D. Edited Proceedings and Books

1. *Botnet Detection: Countering the Largest Security Threat (Advances in Information Security)*, Wenke Lee, Cliff Wang, and David Dagon (Eds.), Springer, 2007.
2. *Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection (RAID 2001)*, Wenke Lee, Ludovic Me, and Andreas Wespi (Eds.), Lecture Notes in Computer Science, Vol. 2212, Springer, 2001.

E. Conference Presentations (refereed)

E.1. Conference Presentations with Proceedings

1. Chenxiong Qian, Hong Hu, Mansour Alharthi, Pak Ho Chung, Taesoo Kim, and Wenke Lee. RAZOR: A Framework for Post-deployment Software Debloating. In *Proceedings of the 28th USENIX Security Symposium*. Santa Clara, CA, August 2019.

2. Ruian Duan, Ashish Bijlani, Yang Ji, Omar Alrawi, Yiyuan Xiong, Moses Ike, Brendan Saltaformaggio, and Wenke Lee. Automating Patching of Vulnerable Open-Source Software Versions in Application Binaries. In *Proceedings of The 2019 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February 2019.
3. Joey Allen, Matthew Landen, Sanya Chaba, Yang Ji, Simon Pak Ho Chung, and Wenke Lee. Improving Accuracy of Android Malware Detection with Lightweight Contextual Awareness. In *Proceedings of the 34th Annual Computer Security Applications Conference (ACSAC)*. December, 2018.
4. Dinuka Sahabandu, Baicen Xiao, Andrew Clark, Sangho Lee, Wenke Lee, and Radha Pooven-
dran. DIFT Games: Dynamic Information Flow Tracking Games for Advanced Persistent
Threats. In *Proceedings of The 57th IEEE Conference on Decision and Control (CDC)*. Mi-
ami Beach, FL, December 2018.
5. Shana Moothedath, Dinuka Sahabandu, Andrew Clark, Sangho Lee, Wenke Lee, and Radha
Poovendran. Multi-Stage Dynamic Information Flow Tracking Game. In *Proceedings of The
9th Conference on Decision and Game Theory for Security (GameSec)*. Seattle, WA, October
2018.
6. Hong Hu, Chenxiong Qian, Carter Yagemann, Simon Pak Ho Chung, Bill Harris, Taesoo
Kim, and Wenke Lee. Enforcing Unique Code Target Property for Control-Flow Integrity. In
*Proceedings of The 25th ACM Conference on Computer and Communications Security (CCS
2018)*. Toronto, Canada, October 2018.
7. Andrea Possemato, Andrea Lanzi, Simon Pak Ho Chung, Wenke Lee, and Yanick Fratantonio.
ClickShield: Are You Hiding Something? Towards Eradicating Clickjacking on Android. In
*Proceedings of The 25th ACM Conference on Computer and Communications Security (CCS
2018)*. Toronto, Canada, October 2018.
8. Yang Ji, Sangho Lee, Mattia Fazzini, Joey Allen, Evan Downing, Taesoo Kim, Alessandro
Orso, and Wenke Lee. Enabling Refinable Cross-Host Attack Investigation with Efficient
Data Flow Tagging and Tracking. In *Proceedings of the 27th USENIX Security Symposium*.
Baltimore, MD, August 2018
9. Wei Meng, Chenxiong Qian, Shuang Hao, Kevin Borgolte, Giovanni Vigna, and Christo-
pher Kruegel, and Wenke Lee. Rampart: Protecting Web Applications from CPU-Exhaustion
Denial-of-Service Attacks. In *Proceedings of the 27th USENIX Security Symposium*. Balti-
more, MD, August 2018
10. Erkam Uzun, Simon Pak Ho Chung, Irfan Essa, and Wenke Lee. rtCaptcha: A Real-Time
CAPTCHA Based Liveness Detection System. In *Proceedings of The 2018 Network and
Distributed System Security Symposium (NDSS)*. San Diego, CA, February 2018.
11. Antonio Bianchi, Yanick Fratantonio, Aravind Machiry, Christopher Kruegel, Giovanni Vi-
gna, Simon Pak Ho Chung, and Wenke Lee. Broken Fingers: On the Usage of the Fingerprint
API in Android. In *Proceedings of The 2018 Network and Distributed System Security Sym-
posium (NDSS)*. San Diego, CA, February 2018.

12. Ruian Duan, Ashish Bijlani, Meng Xu, Taesoo Kim, and Wenke Lee. Identifying Open-Source License Violation and 1-day Security Risk at Large Scale. In *Proceedings of The 24th ACM Conference on Computer and Communications Security (CCS 2017)*. Dallas, Texas, October 2017.
13. Yang Ji, Sangho Lee, Evan Downing, Weiren Wang, Mattia Fazzini, Taesoo Kim, Alessandro Orso, and Wenke Lee. RAIN: Refinable Attack Investigation with On-demand Inter-Process Information Flow Tracking. In *Proceedings of The 24th ACM Conference on Computer and Communications Security (CCS 2017)*. Dallas, Texas, October 2017.
14. Ren Ding, Chenxiong Qian, Chengyu Song, Bill Harris, Taesoo Kim, and Wenke Lee. Efficient Protection of Path-Sensitive Control Security. In *Proceedings of the 26th USENIX Security Symposium*. Vancouver, BC, Canada, August 2017.
15. Meng Xu, Kangjie Lu, Taesoo Kim, and Wenke Lee. Bunshin: Compositing Security Mechanisms through Diversification. In *Proceedings of the 2017 USENIX Annual Technical Conference*. Santa Clara, CA, July 2017.
16. Yanick Fratantonio, Chenxiong Qian, Pak Chung, and Wenke Lee. Cloak and Dagger: From Two Permissions to Complete Control of the UI Feedback Loop. In *Proceedings of The 2017 IEEE Symposium on Security and Privacy*. San Jose, CA, May 2017 (**Distinguished Practical Paper Award**).
17. Kangjie Lu, Marie-Therese Walter, David Pfaff, Stefan Nuernberger, Wenke Lee, and Michael Backes. Unleashing Use-Before-Initialization Vulnerabilities in the Linux Kernel Using Targeted Stack Spraying. In *Proceedings of The 2017 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February 2017.
18. Le Guan, Jun Xu, Shuai Wang, Xinyu Xing, Lin Lin, Heqing Huang, Peng Liu, and Wenke Lee. From Physical to Cyber: Escalating Protection for Personalized Auto Insurance. In *Proceedings of The 14th ACM Conference on Embedded Networked Sensor Systems (SenSys 2016)*. Stanford, CA, November 2016.
19. Kangjie Lu, Chengyu Song, Taesoo Kim, and Wenke Lee. UniSan: Proactive Kernel Memory Initialization to Eliminate Data Leakages. In *Proceedings of The 23rd ACM Conference on Computer and Communications Security (CCS 2016)*. Vienna, Austria, October 2016.
20. Yizheng Chen, Panagiotis Kintis, Manos Antonakakis, Yacin Nadji, David Dagon, Wenke Lee, and Michael Farrell. Financial Lower Bounds of Online Advertising Abuse - A Four Year Case Study of the TDSS/TDL4 Botnet. In *Proceedings of The 13th Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA 2016)*. Sebastian, Spain, July 2016.
21. Chengyu Song, Hyungon Moon, Monjur Alam, Insu Yun, Byoungyoung Lee, Taesoo Kim, and Wenke Lee, Yunheung Paek. HDFI: Hardware-Assisted Data-Flow Isolation. In *Proceedings of The 2016 IEEE Symposium on Security and Privacy*. San Jose, CA, May 2016.

22. Wei Meng, Byoungyoung Lee, Xinyu Xing, and Wenke Lee. TrackMeOrNot: Enable Flexible Control on Web Tracking. In *Proceedings of The 25th International World Wide Web Conference (WWW)*, April 2016. San Diego, CA, February 2016.
23. Wei Meng, Ren Ding, Simon P. Chung, Steven Han, and Wenke Lee. The Price of Free: Privacy Leakage in Personalized Mobile In-Apps Ads. In *Proceedings of The 2016 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February 2016.
24. Kangjie Lu, Wenke Lee, Stefan Nrnberger, and Michael Backes. How to Make ASLR Win the Clone Wars: Runtime Re-Randomization. In *Proceedings of The 2016 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February 2016.
25. Chengyu Song, Byoungyoung Lee, Kangjie Lu, William Harris, Taesoo Kim and Wenke Lee. Enforcing Kernel Security Invariants with Data Flow Integrity. In *Proceedings of The 2016 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February, 2016.
26. Meng Xu, Yeongjin Jang, Xinyu Xing, Taesoo Kim, and Wenke Lee. UCognito: Private Browsing without Tears. In *Proceedings of The 22nd ACM Conference on Computer and Communications Security (CCS)*. Denver, CO, October 2015.
27. Kangjie Lu, Chengyu Song, Byoungyoung Lee, Simon P. Chung, Taesoo Kim, and Wenke Lee. ASLR-Guard: Stopping Address Space Leakage for Code Reuse Attacks. In *Proceedings of The 22nd ACM Conference on Computer and Communications Security (CCS)*. Denver, CO, October 2015.
28. Byoungyoung Lee, Chengyu Song, Taesoo Kim, and Wenke Lee. Type Casting Verification: Stopping an Emerging Attack Vector. In *Proceedings of The 24th USENIX Security Symposium*. Washington, D.C., August 2015. (Awarded the **Internet Defense Prize** by Facebook and USENIX)
29. Xinyu Xing, Wei Meng, Byoungyoung Lee, Udi Weinsberg, Anmol Sheth, Roberto Perdisci, and Wenke Lee. Unraveling the Relationship Between Ad-Injecting Browser Extensions and Malvertising. In *Proceedings of The 24th International World Wide Web Conference (WWW)*. Florence, Italy, May 2015.
30. Chengyu Song, Chao Zhang, Tielei Wang, Wenke Lee, and David Melski. Exploiting and Protecting Dynamic Code Generation. In *Proceedings of The 2015 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February, 2015.
31. Byoungyoung Lee, Chengyu Song, Yeongjin Jang, Tielei Wang, Taesoo Kim, Long Lu, and Wenke Lee. Preventing Use-after-free with Dangling Pointers Nullification. In *Proceedings of The 2015 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February, 2015.
32. Kangjie Lu, Zhichun Li, Vasileios P. Kemerlis, Zhenyu Wu, Long Lu, Cong Zheng, Zhiyun Qian, Wenke Lee, and Guofei Jiang. Checking More and Alerting Less: Detecting Privacy Leakages via Enhanced Data-flow Analysis and Peer Voting. In *Proceedings of The 2015 Network and Distributed System Security Symposium (NDSS)*. San Diego, CA, February, 2015.

33. Wei Meng, Xinyu Xing, Anmol Sheth, Udi Weinsberg, and Wenke Lee. Your Online Interests Pwned! A Pollution Attack Against Targeted Advertising. In *Proceedings of the 21st ACM Conference on Computer and Communications Security (CCS)*. Scottsdale, AZ, November 2014.
34. Yeongjin Jang, Chengyu Song, Simon Chung, Tielei Wang, and Wenke Lee. Allly Attacks: Exploiting Accessibility in Operating Systems. In *Proceedings of the 21st ACM Conference on Computer and Communications Security (CCS)*. Scottsdale, AZ, November 2014.
35. Billy Lau, Pak Ho Chung, Chengyu Song, Yeongjin Jang, Wenke Lee, and Alexandra Boldyreva. Mimesis Aegis: A Mimicry Privacy Shield - A System's Approach to Data Privacy on Public Cloud. In *Proceedings of the 23rd USENIX Security Symposium*. San Diego, CA, August 2014.
36. Tielei Wang, Yeongjin Jang, Yizheng Chen, Pak Ho Chung, Billy Lau, and Wenke Lee. On the Feasibility of Large-Scale Infections of iOS Devices. In *Proceedings of the 23rd USENIX Security Symposium*. San Diego, CA, August 2014.
37. Byoungyoung Lee, Long Lu, Tielei Wang, Taesoo Kim, and Wenke Lee. From Zygote to Morula: Fortifying Weakened ASLR on Android. In *Proceedings of the 35th IEEE Symposium on Security and Privacy (Oakland)*, San Jose, CA, May 2014.
38. Xinyu Xing, Wei Meng, Dan Doozan, Nick Feamster, Wenke Lee, and Alex C. Snoeren. Exposing Inconsistent Web Search Results with Bobble. In *Proceedings of The 2014 Passive and Active Measurement Conference (PAM)*, Los Angeles, CA, March 2014.
39. Yeongjin Jang, Simon P. Chung, Bryan D. Payne, and Wenke Lee. Gyrus: A Framework for User-Intent Monitoring of Text-Based Networked Applications. In *Proceedings of the 21st Annual Network and Distributed System Security Symposium (NDSS)*, San Diego, CA, February 2014.
40. Yacin Nadji, Manos Antonakakis, Roberto Perdisci, and Wenke Lee. Beheading Hydras: Performing Effective Botnet Takedowns. In *Proceedings of the 20th ACM Conference on Computer and Communications Security (CCS)*, Berlin, Germany, November 2013.
41. Brendan Dolan-Gavitt, Tim Leek, Josh Hodosh, and Wenke Lee. Tappan Zee (North) Bridge: Mining Memory Accesses for Introspection. In *Proceedings of the 20th ACM Conference on Computer and Communications Security (CCS)*, Berlin, Germany, November 2013.
42. Xinyu Xing, Wei Meng, Dan Doozan, Alex C. Snoeren, Nick Feamster, and Wenke Lee. Take this Personally: Pollution Attacks on Personalized Services. In *Proceedings of the 22nd USENIX Security Symposium*, Washington, D.C., August 2013.
43. Tielei Wang, Kangjie Lu, Long Lu, Simon Chung, and Wenke Lee. Jekyll on iOS: When Benign Apps Become Evil. In *Proceedings of the 22nd USENIX Security Symposium*, Washington, D.C., August 2013.

44. Yacin Nadji, Manos Antonakakis, Roberto Perdisci, and Wenke Lee. Connected Colors: Unveiling the Structure of Criminal Networks. In *Proceedings of the 16th International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, St. Lucia, October 2013 (to appear).
45. Junjie Zhang, Yinglian Xie, Fang Yu, David Soukal, and Wenke Lee. Intention and Origination: An Inside Look at Large-Scale Bot Queries. In *Proceedings of The 20th Annual Network and Distributed System Security Symposium (NDSS)*, San Diego, CA, February 2013.
46. Charles Lever, Manos Antonakakis, Bradley Reaves, Patrick Traynor and Wenke Lee. The Core of the Matter: Analyzing Malicious Traffic in Cellular Carriers. In *Proceedings of The 20th Annual Network and Distributed System Security Symposium (NDSS)*, San Diego, CA, February 2013.
47. Long Lu, Zhichun Li, Zhenyu Wu, Wenke Lee, and Guofei Jiang. CHEX: Statically Vetting Android Apps for Component Hijacking Vulnerabilities. In *Proceedings of the 19th ACM Conference on Computer and Communications Security (CCS)*, Raleigh, NC. October 2012.
48. Martim Carbone, Matthew Conover, Bruce Montague, and Wenke Lee. Secure and Robust Monitoring of Virtual Machines through Guest-Assisted Introspection. In *Proceedings of the 15th International Symposium on Research in Attacks, Intrusions, and Defenses (RAID)*. Amsterdam, The Netherlands. September, 2012.
49. Manos Antonakakis, Roberto Perdisci, Yacin Nadji, Nikolaos Vasiloglou, Saeed Abu-Nimeh, Wenke Lee, and David Dagon. From Throw-Away Traffic to Bots: Detecting the Rise of DGA-Based Malware. In *Proceedings of the 21st USENIX Security Symposium*. Bellevue, WA. August 2012.
50. Kapil Singh, Helen Wang, Alexander Moshchuk, Collin Jackson, and Wenke Lee. Practical End-to-End Web Content Integrity. In *Proceedings of the 21st International World Wide Web Conference (WWW)*, Lyon, France, April 2012.
51. Xiapu Luo, Peng Zhou, Junjie Zhang, Roberto Perdisci, Wenke Lee, and Rocky K.C. Chang. Exposing Invisible Timing-Based Traffic Watermarks with BACKLIT. In *Proceedings of The 27th Annual Computer Security Applications Conference (ACSAC 2011)*, Orlando, FL, December 2011.
52. Yacin Nadji, Manos Antonakakis, Roberto Perdisci, and Wenke Lee. Understanding the Prevalence and Use of Alternative Plans in Malware with Network Games. In *Proceedings of The 27th Annual Computer Security Applications Conference (ACSAC 2011)*, Orlando, FL, December 2011.
53. Long Lu, Roberto Perdisci, and Wenke Lee. SURF: Detecting and Measuring Search Poisoning. In *Proceedings of The 18th ACM Conference on Computer and Communications Security (CCS)*. Chicago, IL, October 2011.
54. Manos Antonakakis, Roberto Perdisci, Wenke Lee, Nick Vasiloglou, and David Dagon. Detecting Malware Domains at the Upper DNS Hierarchy. In *Proceedings of The 20th USENIX Security Symposium*, San Francisco, CA August 2011.

55. Junjie Zhang, Roberto Perdisci, Wenke Lee, Unam Sarfraz, and Xiapu Luo. Detecting Stealthy P2P Botnets Using Statistical Traffic Fingerprints. In *Proceedings of The 41st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2011)*, Hong Kong, China, June 2011.
56. Xiapu Luo, Peng Zhou, Edmond W. W. Chan, Rocky K. C. Chang, and Wenke Lee. A Combinatorial Approach to Network Covert Communications with Applications in Web Leaks. In *Proceedings of The 41st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2011)*, Hong Kong, China, June 2011.
57. Brendan Dolan-Gavitt, Tim Leek, Michael Zhivich, Jon Giffin, and Wenke Lee. Virtuoso: Narrowing the Semantic Gap in Virtual Machine Introspection. In *Proceedings of The 2010 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2011.
58. Junjie Zhang, Jay Stokes, Christian Seifert, and Wenke Lee. ARROW: Generating Signatures to Detect Drive-By Downloads. In *Proceedings of The 20th International World Wide Web Conference (WWW)*, Hyderabad, India, March 2011.
59. Junjie Zhang, Xiapu Luo, Roberto Perdisci, Guofei Gu, Wenke Lee, and Nick Feamster. Boosting the Scalability of Botnet Detection Using Adaptive Traffic Sampling. In *Proceedings of The 6th ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, Hong Kong, March 2011.
60. Xiapu Luo, Peng Zhou, Edmond W.W. Chan, Wenke Lee, Rocky K. C. Chang, and Roberto Perdisci. HTTPOS: Sealing Information Leaks with Browser-side Obfuscation of Encrypted Flows. In *Proceedings of The 18th Annual Network and Distributed System Security Symposium (NDSS)*, San Diego, CA, February 2011.
61. Qing Hui, Xiapu Luo, and Wenke Lee. Control of Low-Rate Denial-of-Service Attacks on Web Servers and TCP Flows. In *Proceedings of The 49th IEEE Conference on Decision and Control (CDC)*, Atlanta, GA, December 2010.
62. Long Lu, Vinod Yegneswaran, Phil Porras, and Wenke Lee. BLADE: An Attack-Agnostic Approach for Preventing Drive-By Malware Infections. In *Proceedings of The 17th ACM Conference on Computer and Communications Security (CCS)*, Chicago, IL, October 2010.
63. Xiapu Luo, Junjie Zhang, Roberto Perdisci, and Wenke Lee. On the Secrecy of Spread-Spectrum Flow Watermarks. In *Proceedings of The 15th European Symposium on Research in Computer Security (ESORICS)*, Athens, Greece, September 2010.
64. Manos Antonakakis, David Dagon, Xiapu Luo, Roberto Perdisci, and Wenke Lee. A Centralized Monitoring Infrastructure for Improving DNS Security. In *Proceedings of The 13th International Symposium on Recent Advances in Intrusion Detection (RAID)*, Ottawa, Ontario, Canada, September 2010.
65. Manos Antonakakis, Roberto Perdisci, David Dagon, Wenke Lee, and Nick Feamster. Building a Dynamic Reputation System for DNS. In *Proceedings of The 19th USENIX Security Symposium*, Washington, DC, August 2010.

66. Kapil Singh, Samrit Sangal, Nehil Jain, Patrick Traynor, and Wenke Lee. Evaluating Bluetooth as a Medium for Botnet Command and Control. In *Proceedings of The 7th Conference on Detection of Intrusions and Malware Vulnerability Assessment (DIMVA)*, Bonn, Germany, July 2010.
67. Kapil Singh, Alexander Moshchuk, Helen J. Wang, and Wenke Lee. On the Incoherencies in Web Browser Access Control Policies. In *Proceedings of The 2010 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2010.
68. Roberto Perdisci, Wenke Lee, and Nick Feamster. Behavioral Clustering of HTTP-based Malware and Signature Generation using Malicious Network Traces. In *Proceedings of The 7th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, San Jose, CA, April 2010.
69. Roberto Perdisci, Iginio Corona, David Dagon, and Wenke Lee. Detecting Malicious Flux Service Networks through Passive Analysis of Recursive DNS Traces. In *Proceedings of The 25th Annual Computer Security Applications Conference (ACSAC 2009)*, Honolulu, HI, December 2009.
70. Guofei Gu, Vinod Yegneswaran, Phillip Porras, Jennifer Stoll, and Wenke Lee. Active Botnet Probing to Identify Obscure Command and Control Channels. In *Proceedings of The 25th Annual Computer Security Applications Conference (ACSAC 2009)*, Honolulu, HI, December 2009.
71. Monirul Sharif, Wenke Lee, Weidong Cui, and Andrea LANZI. Secure In-VM Monitoring Using Hardware Virtualization. In *Proceedings of The 16th ACM Conference on Computer and Communications Security (CCS 2009)*, Chicago, IL, November, 2009.
72. Martim Carbone, Weidong Cui, Long Lu, Wenke Lee, Marcus Peinado, and Xuxian Jiang. Mapping Kernel Objects to Enable Systematic Integrity Checking. In *Proceedings of The 16th ACM Conference on Computer and Communications Security (CCS 2009)*, Chicago, IL, November, 2009.
73. Kapil Singh, Sumeer Bhola, and Wenke Lee. xBook: Redesigning Privacy Control in Social Networking Platforms. In *Proceedings of The 18th USENIX Security Symposium*, Montreal, Canada, August, 2009.
74. Roberto Perdisci, Manos Antonakakis, Xiapu Luo, and Wenke Lee. WSEC DNS: Protecting Recursive DNS Resolvers from Poisoning Attacks. In *Proceedings of The 39th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2009)*, Lisbon, Portugal, June 2009.
75. Monirul Sharif, Andrea LANZI, Jon Giffin, and Wenke Lee. Automatic Reverse Engineering of Malware Emulators. In *Proceedings of The 2009 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2009. (Best Student Paper Award)
76. Andrea LANZI, Monirul Sharif, and Wenke Lee. K-Tracer: A System for Extracting Kernel Malware Behavior. In *Proceedings of The 16th Annual Network and Distributed System Security Symposium (NDSS 2009)*, San Diego, CA, February 2009.

77. David Dagon, Manos Antonakakis, Kevin Day, Xiapu Luo, Christopher P. Lee, and Wenke Lee. Recursive DNS Architectures and Vulnerability Implications. In *Proceedings of The 16th Annual Network and Distributed System Security Symposium (NDSS 2009)*, San Diego, CA, February 2009.
78. Roberto Perdisci, Andrea Lanzi, and Wenke Lee. McBoost: Boosting Scalability in Malware Collection and Analysis Using Statistical Classification of Executables. In *Proceedings of The 24th Annual Computer Security Applications Conference (ACSAC 2008)*, Anaheim, CA, December 2008.
79. Artem Dinaburg, Paul Royal, Monirul Sharif, and Wenke Lee. Ether: Malware Analysis via Hardware Virtualization Extensions. In *Proceedings of the 15th ACM Conference on Computer and Communications Security (CCS 2008)*, Alexandria, VA, October 2008.
80. David Dagon, Manos Antonakakis, Paul Vixie, Tatuya Jinmei, and Wenke Lee. Increased DNS Forgery Resistance Through 0x20-Bit Encoding. In *Proceedings of the 15th ACM Conference on Computer and Communications Security (CCS 2008)*, Alexandria, VA, October 2008.
81. Monirul Sharif, Vinod Yegneswaran, Hassen Saidi, Phillip Porras, and Wenke Lee. Eureka: A Framework for Enabling Static Malware Analysis. In *Proceedings of the 13th European Symposium on Research in Computer Security (ESORICS)*, Malaga, Spain, October 2008.
82. Guofei Gu, Roberto Perdisci, Junjie Zhang, and Wenke Lee. BotMiner: Clustering Analysis of Network Traffic for Protocol- and Structure-Independent Botnet Detection. In *Proceedings of The 17th USENIX Security Symposium (Security'08)*, San Jose, CA, July 2008.
83. Kapil Singh, Abhinav Srivastava, Jon Giffin, and Wenke Lee. Evaluating Email's Feasibility for Botnet Command and Control. In *Proceedings of the 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2008)*, Anchorage, Alaska, June 2008.
84. Bryan D. Payne, Martim Carbone, Monirul Sharif, and Wenke Lee. Lares: An Architecture for Secure Active Monitoring Using Virtualization. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2008.
85. Guofei Gu, Alvaro A. Cardenas, and Wenke Lee. Principled Reasoning and Practical Applications of Alert Fusion in Intrusion Detection Systems. In *Proceedings of the ACM Symposium on InformAction, Computer and Communications Security (ASIACCS'08)*, Tokyo, Japan, March 2008.
86. David Dagon, Niels Provos, Chris Lee, and Wenke Lee. Corrupted DNS Resolution Paths: The Rise of a Malicious Resolution Authority. In *Proceedings of The 15th Annual Network and Distributed System Security Symposium (NDSS 2008)*, San Diego, CA, February 2008.
87. Guofei Gu, Junjie Zhang, and Wenke Lee. BotSniffer: Detecting Botnet Command and Control Channels in Network Traffic. In *Proceedings of The 15th Annual Network and Distributed System Security Symposium (NDSS 2008)*, San Diego, CA, February 2008.

88. Monirul Sharif, Andrea Lanzi, Jonathon Giffin, and Wenke Lee. Impeding Malware Analysis using Conditional Code Obfuscation. In *Proceedings of The 15th Annual Network and Distributed System Security Symposium (NDSS 2008)*, San Diego, CA, February 2008.
89. Bryan D. Payne, Martim Carbone, and Wenke Lee. Secure and Flexible Monitoring of Virtual Machines. In *Proceedings of The 23rd Annual Computer Security Applications Conference (ACSAC 2007)*, Miami Beach, FL, December 2007.
90. David Dagon, Guofei Gu, Chris Lee and Wenke Lee. A Taxonomy of Botnet Structures. In *Proceedings of The 23rd Annual Computer Security Applications Conference (ACSAC 2007)*, Miami Beach, FL, December 2007.
91. Guofei Gu, Zesheng Chen, Phillip Porras and Wenke Lee. Misleading and Defeating Importance-Scanning Malware Propagation. In *Proceedings of The 3rd International Conference on Security and Privacy in Communication Networks (SecureComm'07)*, Nice, France, September 2007.
92. Takehiro Takahashi and Wenke Lee. An Assessment of VoIP Covert Channel Threats. In *Proceedings of The 3rd International Conference on Security and Privacy in Communication Networks (SecureComm'07)*, Nice, France, September 2007.
93. Monirul Sharif, Kapil Singh, Jonathon Giffin and Wenke Lee. Understanding Precision in Host Based Intrusion Detection: Formal Analysis and Practical Models. In *Proceedings of The 10th International Symposium on Recent Advances in Intrusion Detection (RAID)*, Surfers Paradise, Australia, September 2007.
94. Guofei Gu, Phillip Porras, Vinod Yegneswaran, Martin Fong, Wenke Lee. BotHunter: Detecting Malware Infection Through IDS-Driven Dialog Correlation. In *Proceedings of The 16th USENIX Security Symposium (Security'07)*, Boston, MA, August 2007.
95. David Cash, Yan Zong Ding, Yevgeniy Dodis, Wenke Lee, Richard Lipton, and Shabsi Wal-fish. Intrusion-Resilient Key Exchange in the Bounded Retrieval Model. In *Proceedings of The Fourth IACR Theory of Cryptography Conference (TCC 2007)*, Amsterdam, The Netherlands, February 2007.
96. Roberto Perdisci, Guofei Gu, and Wenke Lee. Using an Ensemble of One-Class SVM Classifiers to Harden Payload-based Anomaly Detection Systems. In *Proceedings of The 2006 IEEE International Conference on Data Mining (ICDM '06)*, Hong Kong, China, December 2006.
97. Paul Royal, Mitch Halpin, David Dagon, Robert Edmonds, and Wenke Lee. PolyUnpack: Automating the Hidden-Code Extraction of Unpack-Executing Malware. In *Proceedings of The 22th Annual Computer Security Applications Conference (ACSAC 2006)*, Miami Beach, FL, December 2006.
98. Prahlad Fogla and Wenke Lee. Evading Network Anomaly Detection Systems: Formal Reasoning and Practical Techniques. In *Proceedings of The 13th ACM Conference on Computer and Communications Security (CCS 2006)*, Alexandria, VA, October 2006.

99. Guofei Gu, Prahlad Fogla, David Dagon, Wenke Lee, and Boris Skoric. Towards an Information-Theoretic Framework for Analyzing Intrusion Detection Systems. In *Proceedings of The 11th European Symposium Research Computer Security (ESORICS 2006)*, Hamburg, Germany, September 2006.
100. Prahlad Fogla, Monirul Sharif, Roberto Perdisci, Oleg Kolesnikov, and Wenke Lee. Polymorphic Blending Attacks. In *Proceedings of The 15th USENIX Security Symposium (SECURITY '06)*, Vancouver, B.C., Canada, August 2006.
101. Collin Mulliner, Giovanni Vigna, David Dagon, and Wenke Lee. Using Labeling to Prevent Cross-Service Attacks Against Smart Phones. In *Proceedings of The 3rd Conference on Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA 2006)*, Berlin, Germany, July 2006.
102. Hongmei Deng, Roger Xu, Jason H. Li, Frank Zhang, Renato Levy, and Wenke Lee. Agent-Based Cooperative Anomaly Detection for Wireless Ad Hoc Networks. In *Proceedings of The 12th International Conference on Parallel and Distributed Systems (ICPADS 2006)*, Minneapolis, Minnesota, July 2006.
103. Guofei Gu, Prahlad Fogla, Wenke Lee, and Douglas Blough. DSO: Dependable Signing Overlay. In *Proceedings of The 4th International Conference on Applied Cryptography and Network Security (ACNS '06)*, Singapore, June 2006.
104. Roberto Perdisci, David Dagon, Wenke Lee, Prahlad Fogla, and Monirul Sharif. Misleading Worm Signature Generators Using Deliberate Noise Injection (full paper). In *Proceedings of the 2006 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2006.
105. Guofei Gu, Prahlad Fogla, David Dagon, Wenke Lee, and Boris Skoric. Measuring Intrusion Detection Capability: An Information-Theoretic Approach. In *Proceedings of ACM Symposium on InformAction, Computer and Communications Security (ASIACCS '06)*, Taipei, Taiwan, March 2006.
106. David Dagon, Cliff Zou, and Wenke Lee. Modeling Botnet Propagation Using Time Zones. In *Proceedings of The 13th Annual Network and Distributed System Security Symposium (NDSS 2006)*, San Diego, CA, February 2006.
107. Yongguang Zhang, Yi-an Huang, and Wenke Lee. An Extensible Environment for Evaluating Secure MANET. In *Proceedings of The 1st International Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm 2005)*, Athens, Greece, September 2005.
108. Jonathon T. Giffin, David Dagon, Somesh Jha, Wenke Lee, and Barton P. Miller. Environment-Sensitive Intrusion Detection. In *Proceedings of The 8th International Symposium on Recent Advances in Intrusion Detection (RAID 2005)*, Seattle, WA, September 2005.
109. David Dagon, Wenke Lee, and Richard Lipton. Protecting Secret Data from Insider Attacks. *Proceedings of The Ninth International Conference on Financial Cryptography and Data Security (FC'05)*, Roseau, Dominica, February, 2005.

110. Guofei Gu, David Dagon, Xinzhou Qin, Monirul I. Sharif, Wenke Lee, and George F. Riley. Worm Detection, Early Warning, and Response Based on Local Victim Information. *Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC 2004)*, Tucson, Arizona, December 2004.
111. Xinzhou Qin and Wenke Lee. Attack Plan Recognition and Prediction Using Causal Networks. *Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC 2004)*, Tucson, Arizona, December 2004.
112. Joao B.D. Cabrera, Jaykumar Gosar, Wenke Lee, and Raman K. Mehra. On the Statistical Distribution of Processing Times in Network Intrusion Detection. *Proceedings of the 43rd IEEE Conference on Decision and Control (CDC 2004)*, Bahamas, December 2004.
113. George F. Riley, Monirul I. Sharif, and Wenke Lee. Simulating Internet Worms. *Proceedings of the 12th Annual Meeting of the IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Volendam, The Netherlands, October 2004.
114. Yian Huang and Wenke Lee. Attack Analysis and Detection for Ad Hoc Routing Protocols. *Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID 2004)*, Sophia Antipolis, France, September 2004.
115. David Dagon, Xinzhou Qin, Guofei Gu, Wenke Lee, Julian Grizzard, John Levin, and Henry Owen. HoneyStat: Local Worm Detection Using Honeypots. *Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID 2004)*, Sophia Antipolis, France, September 2004.
116. Xinzhou Qin and Wenke Lee. Discovering Novel Attack Strategies from INFOSEC Alerts. *Proceedings of the 9th European Symposium on Research in Computer Security (ESORICS 2004)*, Sophia Antipolis, France, September 2004.
117. H. Feng, Jonathon T. Giffin, Yong Huang, Somesh Jha, Wenke Lee, and Barton P. Miller. Formalizing Sensitivity in Static Analysis for Intrusion Detection. *Proceedings of the 2004 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2004.
118. Xinzhou Qin and Wenke Lee. Statistical Causality Analysis of INFOSEC Alert Data. In *Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection (RAID 2003)*, Pittsburgh, PA, September 2003.
119. H. Feng, Oleg Kolesnikov, Prahlad Fogla, Wenke Lee, and Weibo Gong. Anomaly Detection Using Call Stack Information. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2003.
120. Yi-an Huang, Wei Fan, Wenke Lee, and Philip S. Yu. Cross-Feature Analysis for Detecting Ad-Hoc Routing Anomalies. In *Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS)*, Providence, RI, May 2003.

121. Joao B.D. Cabrera, Wenke Lee, R.K. Prasanth, Lundy Lewis, and Raman K. Mehra. Optimization and Control Problems in Real Time Intrusion Detection. *Proceedings of the 41st IEEE Conference on Decision and Control (CDC 2002)*, Las Vegas, December, 2002.
122. Wenke Lee, Joao B.D. Cabrera, Ashley Thomas, Niranjana Balwalli, Sunmeet Saluja, and Yi Zhang. Performance Adaptation in Real-Time Intrusion Detection Systems. In *Proceedings of the 5th International Symposium on Recent Advances in Intrusion Detection (RAID 2002)*, Zurich, Switzerland, October 2002.
123. Xinzhou Qin, Wenke Lee, Lundy Lewis, and Joao B.D. Cabrera. Integrating Intrusion Detection and Network Management. *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS 2002)*, Florence, Italy, May 2002.
124. Wei Fan, Matt Miller, Sal Stolfo, Wenke Lee, and Phil Chan. Using Artificial Anomalies to Detect Unknown and Known Network Intrusions. *Proceedings of The First IEEE International Conference on Data Mining (ICDM)*, San Jose, CA, November 2001.
125. Wenke Lee, Sal Stolfo, Phil Chan, Eleazar Eskin, Wei Fan, Matt Miller, Shlomo HersHKop, and Junxin Zhang. Real Time Data Mining-based Intrusion Detection. *Proceedings of the 2001 DARPA Information Survivability Conference and Exposition (DISCEX II)*, Anaheim, CA, June 2001.
126. Wenke Lee and Dong Xiang. Information-Theoretic Measures for Anomaly Detection. *Proceedings of The 2001 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2001.
127. Joao B.D. Cabrera, L. Lewis, X. Qin, Wenke Lee, Ravi Prasanth, B. Ravichandran, and Raman Mehra. Proactive Detection of Distributed Denial of Service Attacks Using MIB Traffic Variables - A Feasibility Study. *Proceedings of The Seventh IFIP/IEEE International Symposium on Integrated Network Management (IM 2001)*, Seattle, WA, May 2001.
128. Yongguang Zhang and Wenke Lee. Intrusion Detection in Wireless Ad-Hoc Networks. *Proceedings of The Sixth International Conference on Mobile Computing and Networking (MobiCom 2000)*, Boston, MA, August 2000.
129. Wei Fan, Wenke Lee, Sal Stolfo, and Matt Miller. A Multiple Model Cost-Sensitive Approach for Intrusion Detection. *Proceedings of The Eleventh European Conference on Machine Learning (ECML 2000)*, LNAI 1810, Barcelona, Spain, May 2000.
130. Sal Stolfo, Wei Fan, Wenke Lee, Andreas Prodromidis, and Phil Chan. Cost-based Modeling for Fraud and Intrusion Detection: Results from the JAM Project. *Proceedings of the 2000 DARPA Information Survivability Conference and Exposition (DISCEX '00)*, Hilton Head, SC, January 2000.
131. Wenke Lee, Sal Stolfo, and Kui Mok. Mining in a Data-flow Environment: Experience in Network Intrusion Detection. *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '99)*, San Diego, CA, August 1999.

132. Wenke Lee, Sal Stolfo, and Kui Mok. A Data Mining Framework for Building Intrusion Detection Models. *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, Oakland, CA, May 1999.
133. Wenke Lee, Sal Stolfo, and Kui Mok. Mining Audit Data to Build Intrusion Detection Models. *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD '98)*, New York, NY, August 1998.
134. Wenke Lee and Sal Stolfo. Data Mining Approaches for Intrusion Detection. *Proceedings of the Seventh USENIX Security Symposium (SECURITY '98)*, San Antonio, TX, January 1998.
135. Sal Stolfo, Andreas Prodromidis, Shelley Tselepis, Wenke Lee, Wei Fan, and Phil Chan. JAM: Java Agents for Meta-learning over Distributed Databases. *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining (KDD '97)*, Newport Beach, CA, August 1997.
136. Naser S. Barghouti, John Mocenigo, and Wenke Lee. Grappa: A GRAPh PACKage in Java. *Proceedings of the Fifth Annual Symposium on Graph Drawing (Graph Drawing '97)*, Rome, Italy, September 1997.
137. Wenke Lee, Gail Kaiser, Paul Clayton, and Eric Sherman. OzCare: A Workflow Automation System for Care Plans. *Proceedings of the American Medical Informatics Association Annual Fall Symposium*, Washington DC, October 1996.

F. Workshop Presentations (refereed)

F.1. Workshop Presentations with Proceedings

1. Yi-an Huang and Wenke Lee. Hotspot-Based Traceback for Mobile Ad Hoc Networks. In *Proceedings of The ACM Workshop on Wireless Security (WiSe 2005)*, Cologne, Germany, September 2005.
2. Monirul Sharif, George Riley, and Wenke Lee. Comparative Study between Analytical Models and Packet-Level Worm Simulations. In *Proceedings of The 19th Workshop on Parallel and Distributed Simulation (PADS 2005)*, Monterey, CA, June 2005.
3. Chris Clark, Wenke Lee, David Schimmel, Didier Contis, Mohamed Kone, and Ashley Thomas. A Hardware Platform for Network Intrusion Detection and Prevention. *The 3rd Workshop on Network Processors and Applications (NP3)*, Madrid, Spain, February 2004.
4. Yian Huang and Wenke Lee. A Cooperative Intrusion Detection System for Ad Hoc Networks. *ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '03)*, Fairfax VA, October 2003.
5. Mustaque Ahamad, Wenke Lee, Ling Liu, Leo Mark, Edward Omicienski, Calton Pu, and Andre dos Santos. Guarding the Next Internet Frontier: Countering Denial of Information Attacks. *Proceedings of the 2002 New Security Paradigms Workshop*, Virginia Beach, Virginia, September 2002.

6. Xinzhou Qin, Wenke Lee, Lundy Lewis, and Joao B.D. Cabrera. Using MIB III Variables for Network Anomaly Detection - A Feasibility Study. *ACM Workshop on Data Mining for Security Applications*, Philadelphia, PA, November 2001.
7. Yongguang Zhang, Harrick Vin, Lorenzo Alvisi, Wenke Lee, and Son K. Dao. Heterogeneous Networking: A New Survivability Paradigm. *Proceedings of the 2001 New Security Paradigms Workshop*, Cloudcroft, New Mexico, September 2001.
8. Wenke Lee, Wei Fan, Matt Miller, Sal Stolfo, and Erez Zadok. Toward Cost-Sensitive Modeling for Intrusion Detection and Response. *ACM Workshop on Intrusion Detection Systems*, Athens, Greece, November 2000.
9. Wenke Lee, Rahul Nimbalkar, Kam Yee, Sunil Patil, Pragnesh Desai, Thuan Tran, and Sal Stolfo. A Data Mining and CIDF Based Approach for Detecting Novel and Distributed Intrusions. *Proceedings of The Third International Workshop on Recent Advances in Intrusion Detection (RAID 2000)*, LNCS 1907, Toulouse, France, October 2000.
10. Wenke Lee, Chris Park, and Sal Stolfo. Towards Automatic Intrusion Detection using NFR. *1st USENIX Workshop on Intrusion Detection and Network Monitoring*, April 1999.
11. Wenke Lee, Sal Stolfo, and Phil Chan. Learning Patterns from Unix Process Execution Traces for Intrusion Detection. *AAAI Workshop: AI Approaches to Fraud Detection and Risk Management*, July 1997.
12. Sal Stolfo, Wei Fan, Wenke Lee, Andreas Prodromidis, and Phil Chan. Credit Card Fraud Detection Using Meta-Learning: Issues and Initial Results. *AAAI Workshop: AI Approaches to Fraud Detection and Risk Management*, July 1997.
13. Gail Kaiser and Wenke Lee. Pay No Attention to the Man Behind the Curtain. *NSF Workshop on Workflow and Process Automation*, May 1996.
14. Wenke Lee. Data Modeling and Management for Large Spatial Databases. *The Third International Workshop in Geographic Information Systems*, Beijing, China, August 1993.

G. Other

1. Matt Blaze, Sampath Kannan, Insup Lee, Oleg Sokolsky, Jonathan Smith, Angelos Keromytis, and Wenke Lee. Dynamic Trust Management. *IEEE Computer*, February 2009.
2. Martim Carbone, Diego Zamboni, and Wenke Lee. Taming Virtualization. *IEEE Security & Privacy*, vol. 6, no. 1, January/February 2008.
3. Bryan D. Payne, Reiner Sailer, Ramon Caceres, Ronald Perez, and Wenke Lee. A Layered Approach to Simplified Access Control in Virtualized Systems. *ACM SIGOPS Operating Systems Review*, 4(2), July 2007.
4. Wenke Lee. 2002. Applying Data Mining to Intrusion Detection: The Quest for Automation, Efficiency, and Credibility. *SIGKDD Explorations*, 4(2), December 2002.

5. Wenke Lee and Wei Fan. 2001. Mining System Audit Data: Opportunities and Challenges. *SIGMOD Record*, 30(4), December 2001.
6. Salvatore J. Stolfo, Wenke Lee, Philip K. Chan, Wei Fan, and Eleazar Eskin. 2001. Data Mining-Based Intrusion Detectors: An Overview of the Columbia IDS Project. *SIGMOD Record*, 30(4), December 2001.

H. Research Proposals and Grants

a. Approved and Funded

1. **Interactive Editing Techniques for Subsetting and Dialecting Network Protocols**
Taesoo Kim (PI); co-PIs: Wenke Lee, Alex Orso, and Brendan Saltaformaggio
ONR
Funded at \$2,976,781, 8/1/2018-7/31/2021
2. **Comprehensive System Debloating via Path-Based Learning and Late-Stage OS Composition**
Wenke Lee (for Bill Harris) (PI); co-PIs: Bill Harris, Taesoo Kim, Alessandro Orso, and Santosh Pande
ONR
Funded at \$7,500,000, 8/1/2017-7/31/2022
3. **CyberCorps Scholarship-for-Service (SFS)**
Mustaque Ahamad (PI); co-PIs: Wenke Lee, Taesoo Kim, Sy Goodman, and Manos Antonakakis
NSF
Funded at \$5,000,000, 9/1/2016-10/31/2021.
4. **Multi-Layer Adaptive and Proactive Strategic Cyber Defense**
Radha Poovendran (PI, University of Washington); co-PIs: Wenke Lee (Georgia Tech) and Tamer Basar (UIUC)
ARO
Funded at \$1,000,000, 8/1/2016-7/31/2019.
5. **Intel Science & Technology Center for Adversary-Resilient Security Analytics (ISTC-ARSA)**
Wenke Lee (PI); co-PIs: Taesoo Kim, Polo Chau (CSE), and Le Song (CSE)
Funded as a \$1,500,000 gift.
6. **THEIA: Tagging and Tracking of Multi-Level Host Events for Transparent Computing and Information Assurance**
Wenke Lee (PI); co-PIs: Taesoo Kim, Alex Orso, and Simon Chung
DARPA
Funded at \$4,193,126, 6/26/2015-6/26/2019.
7. **ADAPT: Analytical Framework for Actionable Defense against Advanced Persistent Threats**
Radha Poovendran (PI, University of Washington); co-PIs: Wenke Lee (Georgia Tech), Shankar Sastry and Anthony Joseph (UC Berkeley), Joao Hespanha (UCSB), Tamer Basar (UIUC), and Maryam Fazel and Linda Bushnell (University of Washington)
ONR MURI
Funded at \$7,500,000, 6/1/2016-5/31/2021.

8. **Embedasploit: A “Pen-Test in a Box” for Industrial Control Systems**
Wenke Lee (PI); co-PIs: Sal Stolfo at Columbia, Brendan Dolan-Gavitt at NYU Tandon School of Engineering
Office of Naval Research
Funded at \$1,250,351, 5/1/2015-4/30/2018.
9. **BFT++: Attack Tolerance in Hard Real-Time Systems**
Taesoo Kim (PI); co-PI: Wenke Lee
Office of Naval Research
Funded at \$1,245,720, 4/1/2015-3/31/2018.
10. **TWC SBE: TTP Option: Medium: Collaborative: EPICA: Empowering People to Overcome Information Controls and Attacks**
Wenke Lee (PI); co-PIs: Nick Feamster, Hongyuan Zha, and Hans Klein
NSF
Funded at \$1,100,000, 8/1/14-7/31/17.
11. **Comprehensive Understanding of Malicious Overlay Networks**
Homeland Security Advanced Project Research Agency (HSARPA of DHS)
Wenke Lee (PI); co-PIs: David Dagon at Georgia Tech, Chris Smoak of GTRI, Roberto Perdisci (UGA), April Lorenzen of Dissect Cyber, Paul Vixie of Internet Systems Consortium, Jody Westby of Global Cyber Risk, and Matt Jonkman of Open Information Security Foundation.
Funded at \$1,873,078 for project total, with \$673,078 to CoC/GT, 10/1/12–9/30/15.
12. **EAGER: The Conceptual Landscape of Information Manipulation**
Wenke Lee (PI); co-PIs: Nick Feamster, and Hans Klein (Public Policy)
NSF
Funded at \$200,000, 8/1/12–7/31/14.
13. **Integrating Security Concepts in an Undergraduate Computer Sciences Curriculum**
Intel
Wenke Lee (PI); co-PIs: Mustaque Ahamad, Nick Feamster, and Paul Royal
Funded as unrestricted gift at \$50,000, 11/2011;
additional, unrestricted gift at \$35,000, 3/2012;
additional, unrestricted gift at \$50,000, 11/2012;
a total of \$135,000.
14. **Monitoring Free and Open Access to Information on the Internet**
Google
Nick Feamster and Wenke Lee (co-PIs)
Funded as unrestricted gift: \$1,000,000 base for 12/15/2010–12/14/2012;
\$500,000 option for 12/15/2012–12/14/2013;
additional gift at \$500,000, for 12/15/13–12/14/2014;
a total of \$2,000,000.

15. **Dimensions of Network Intelligence**
HSARPA of DHS
Wenke Lee (PI) and Paul Royal (co-PI)
Funded at \$1,022,000, 9/2012-9/2017.
16. **Dynamic Reputation for DNS**
Google
Wenke Lee (PI)
Funded as unrestricted gift at \$75,000, 7/2010.
17. **PEASOUP: Preventing Exploits Against Software Of Uncertain Provenance**
IARPA/STONESOUP Program (GrammaTech subcontract)
Wenke Lee (co-PI), with David Melski (PI) at GrammaTech, Jack Davidson and John Knight (co-PIs) at University of Virginia, and Tom Bracewell (co-PI) at Raytheon
Funded at \$13,000,000 for project total, with \$1,695,887 to Georgia Tech, 6/1/2010–5/31/2014.
18. **Research in Virtual Machine Monitoring Techniques**
Sandia National Labs/Sandia Corp.
Wenke Lee (PI)
Funded at \$50,000, 9/15/2010–9/15/2011.
19. **TC: SMALL: A Foundational and Practical Platform for Host Security Applications**
NSF CNS/Trustworthy Computing
Wenke Lee (PI)
Funded at \$429,571, 7/15/2010–7/15/2013.
20. **Botnet Attribution and Removal: From Axioms to Theory to Practice**
ONR MURI
Wenke Lee (PI); Co-PIs: Nick Feamster, and Jon Giffin at Georgia Tech; Farnam Jahanian, Mike Bailey, and Kang Shin at University of Michigan, Ann Arbor; Giovanni Vigna and Chris Kruegel at University of California, Santa Barbara; John Mitchell at Stanford University
Funded: \$7,500,000 project total, \$2,600,000 for Georgia Tech, 5/1/2009–4/30/2014.
21. **Collaborative Research: CT-L: CLEANSE: Cross-Layer Large-Scale Efficient Analysis of Network Activities to Secure the Internet**
NSF CNS/Cyber Trust
Wenke Lee (PI); Co-PIs: Nick Feamster, Jon Giffin, Mustaque Ahamad, and Xiaoming Huo (ISyE) at Georgia Tech; Mike Reiter and Fabian Monrose at University of North Carolina at Chapel Hill; Farnam Jahanian and Mike Bailey at University of Michigan; Phil Porras and Vinod Yegneswaran at SRI International; and Paul Vixie at Internet Software Consortium.
Funded: \$1,829,297 project total, \$758,297 for Georgia Tech, 9/1/2008–8/31/2012.
22. **SMITE: Scalable Monitoring in the Extreme**
DARPA
Wenke Lee and Nick Feamster (co-PIs at Georgia Tech); BBN subcontract
Funded: \$574,999 for Georgia Tech, 4/1/2008–3/31/2010

23. **Countering Botnets: Anomaly-Based Detection, Comprehensive Analysis, and Efficient Mitigation**
DHS HSARPA
Wenke Lee (PI); with Nick Feamster and Jon Giffin at Georgia Tech, David Dagon at Damballa, Rick Wesson and Adam Waters at Support Intelligence, Paul Vixie at Internet Systems Consortium, and Jody Westby at Global Cyber Risk
Funded: \$1,050,730 project total, \$300,000 for Georgia Tech, 4/1/2008–3/31/2010.
Add-on: \$305,066 project total, \$190,066 to Georgia Tech, 7/1/2009–9/30/2010.
24. **CRI-IAD: Collaborative Research: Enabling Security and Network Management Research for Future Networks**
NSF CRI
Nick Feamster (co-PI), Alex Gray, and Wenke Lee at Georgia Tech, and Z. Morley Mao (co-PI) and Farnam Jahanian at University of Michigan
Funded: \$397,426 for Georgia Tech (3/1/2008–2/28/2011)
25. **Collaborative Research: CT-T: Logic and Data Flow Extraction for Live and Informed Malware Execution**
NSF CNS/CyberTrust
Wenke Lee (co-PI); co-PIs: Paul Barford at University of Wisconsin at Madison, and Phil Porras at SRI International
Funded: \$220,000 for Georgia Tech (9/1/2007–8/31/2009)
26. **Foundational and Systems Support for Quantitative Trust Management**
Office of Naval Research MURI (University of Pennsylvania subcontract)
Sampath Kannan (PI); co-PIs: Insup Lee, Oleg Sokolsky, Matt Blazes, and Jonathan Smith at University of Pennsylvania, Wenke Lee at Georgia Tech, and Angelos Keromytis at Columbia University.
Funded: \$1,125,000 for Georgia Tech (project total: \$4,000,000) (5/1/2007–4/30/2012)
27. **Collaborative Research: CT-ISG: Modeling and Measuring Botnets**
NSF CNS/CyberTrust
Wenke Lee (PI) and Cliff C. Zou (co-PI) at University of Central Florida
Funded: \$350,000 for project total, Georgia Tech at \$175,000, (8/15/2006-8/15/2009)
28. **CT-ISG: Trusted Passage: Managing Distributed Trust to Meet the Needs of Emerging Applications**
NSF CNS/CyberTrust
Mustaque Ahamad (PI), Wenke Lee (co-PI), and Karsten Schwan (co-PI)
Funded: \$350,000, (8/15/2006-8/15/2009)
29. **Cyber Threat Analytics**
ARO via subcontract from SRI International
Funded: \$80,000, (7/15/2006-7/15/2007)
30. **ARL CTA**
ARL via subcontract from SPARTA/Telcordia
Funded: \$75,000, (7/15/2006-7/15/2007)

31. **Computer Network Defenses Technologies**
AFRL via subcontract from Scientific Systems Co. Funded: \$30,000, (9/13/2006-3/13/2007)
32. **ARO Workshop on Research in Botnets**
ARO
Funded: \$25,000, (4/1/2006-6/1/2007)
33. **Next-Generation Botnet Detection and Response**
DARPA (via ARO)
Funded: \$291,346, (12/15/2005-6/14/07)
34. **An Information-Theoretic Framework for Evaluating and Optimizing Intrusion Detection Performance**
Army Research Office
Funded: \$199,745, (4/1/2005-1/30/07)
35. **Preventing SQL Code Injection by Combining Static and Runtime Analysis, funded by Department of Homeland Security**
Department of Homeland Security
Alex Orso (PI) and Wenke Lee (co-PI)
Funded: \$200,000, (7/1/2005-12/31/2006)
36. **Anomaly and Misuse Detection in Network Traffic Streams – Checking and Machine Learning Approaches**
Office of Naval Research MURI (University of Pennsylvania subcontract)
Sampath Kannan (PI), Insup Lee, Oleg Sokolsky, and Linda Zhao at University of Pennsylvania, Wenke Lee at Georgia Tech, and Diana Spears and William Spears at University of Wyoming.
Funded: \$400,000 for Georgia Tech (\$2,000,000 project total) (7/15/2004–6/1/2006)
37. **Intrusion Detection and Security Management Technologies for Mobile Ad Hoc Networks**
Army Research Lab (Scientific Systems Company Inc. subcontract)
Funded: \$65,000 (2/1/2005-2/1/2006)
38. **A Course Module on Wireless Security**
Microsoft, Curriculum Development Grant
Funded: \$30,000 (unrestricted gift)
39. **Intrusion Detection Techniques for Mobile Ad Hoc Networks**
NSF, CISE/CCR/Trusted Computing
Funded: \$275,000 (8/15/2003-7/31/2006)
REU Funded: \$15,000 (9/10/2004-7/31/2006)
40. **High-Speed Intrusion Detection Sensors**
Army Research Lab (Scientific Systems Company Inc. subcontract)
Funded: \$65,000 (8/7/2003-8/6/2005)

41. **Distributed Intrusion Detection Feasibility Study**
U.S. Government (Columbia University subcontract)
Funded: \$45,000 (6/1/2003-6/30/2004)
42. **Agile Security for Storing Sensitive and Critical Information**
NSF, CISE/Trusted Computing
Mustaque Ahamad (PI), and co-PIs: Wenke Lee, Doug Blough, and H. Venkateswaran
Funded: \$460,000 (8/15/2002-7/31/2005)
43. **CAREER: Adaptive Intrusion Detection Systems**
NSF, CISE/CAREER
Funded: \$350,000 (8/15/2002-7/31/2007)
44. **ITR/SI: Guarding the Next Internet Frontier: Countering Denial of Information**
NSF, ITR
Mustaque Ahamad (PI), and co-PIs: Wenke Lee, Andre dos Santos, Ling Liu, Leo Mark, Ed Omiecinski, and Calton Pu
Funded: \$1,694,769 (9/1/2001-8/31/2006)
45. **Vulnerability Assessment Tools for Complex Information Networks**
Army Research Office MURI (Harvard University subcontract)
Yu-Chi Ho (PI) at Harvard University, co-PIs: Avrom Pfeffer at Harvard University, Christos G. Cassandras at Boston University, Wei-Bo Gong at University of Massachusetts, Amherst, and Wenke Lee at Georgia Tech
Funded: \$400,000 for Georgia Tech (\$2,000,000 project total) (5/1/2001-4/30/2006)
46. **A Data Mining Approach for Building Cost-sensitive and Light Intrusion Detection Models.**
DARPA, ATO
Wenke Lee (PI), and co-PIs: Salvatore J. Stolfo at Columbia University and Philip Chan at Florida Institute of Technology
Funded: \$854,000 for Georgia Tech (\$2,000,001 project total) (6/1/2000-8/31/2003)

I. Research Honors and Awards

1. CCS Test of Time Award, The 25th ACM Conference on Computer and Communications Security (CCS), for the paper “Artem Dinaburg, Paul Royal, Monirul Sharif, and Wenke Lee. Ether: Malware Analysis via Hardware Virtualization Extensions” published in CCS 2008 that have had the greatest impact on security research and practice over the past decade, 2018.
2. ACM Fellow, 2017.
3. Distinguished Practical Paper Award, for the paper “Yanick Fratantonio, Chenxiong Qian, Pak Chung, and Wenke Lee. Cloak and Dagger: From Two Permissions to Complete Control of the UI Feedback Loop” in Proceedings of The 2017 IEEE Symposium on Security and Privacy, 2017.
4. Internet Defense Prize, awarded by Facebook and USENIX, for the paper “Byoungyoung Lee, Chengyu Song, Taesoo Kim, and Wenke Lee. Type Casting Verification: Stopping an Emerging Attack Vector” in Proceedings of The 24th USENIX Security Symposium, 2015.
5. Cyber Is A Global Sport Award (awarded to David Dagon and Wenke Lee), Department of Homeland Security, Science and Technology Directorate, Cyber Security Division, 2015.
6. Outstanding Achievement in Research Program Development Award, (awarded to Mustaque Ahamad, Wenke Lee, Paul Royal, and David Dagon), Georgia Institute of Technology, 2015.
7. Outstanding Community Service Award, the IEEE Technical Committee on Security and Privacy, 2013
8. Outstanding Faculty Leadership for the Development of GRAs Award, Georgia Institute of Technology, 2012.
9. Sigma Xi Faculty Best Paper Award, Georgia Institute of Technology, 2010.
10. Best Student Paper Award, for the paper “Monirul Sharif, Andrea Lanzi, Jon Giffin, and Wenke Lee. Automatic Reverse Engineering of Malware Emulators” in Proceedings of The 2009 IEEE Symposium on Security and Privacy, 2009.
11. Outstanding Senior Faculty Research Award, College of Computing, Georgia Institute of Technology, 2009.
12. NSF CAREER Award, 2002.
13. Best Paper Award, Applied Research Category, the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '99), August 1999 (with Kui Mok and Sal Stolfo).
14. Honorable mention (runner-up) for Best Paper Award, Applied Research Category, the 4th International Conference on Knowledge Discovery and Data Mining (KDD '98), August 1998.
15. Honorable mention (runner-up) for Best Paper Award, Applied Research Category, the 3rd International Conference on Knowledge Discovery and Data Mining (KDD '97), August 1997.

III. SERVICE

A. Professional Activities

A.1 Membership and Activities in Professional Societies

1. Fellow, Association for Computing Machinery (ACM); Member, Institute of Electrical and Electronics Engineers (IEEE), and IEEE Computer Society

A.2 Journal Editorial Board Activities

1. IEEE Transactions on Dependable and Secure Computing (TDSC) 2010 - 2011.
2. ACM Transactions on Information and System Security (TISSEC) 2005 - 2011.

A.3 Conference Committee Activities

1. The 2019 IEEE Symposium on Security & Privacy.
2. The 2018 IEEE Symposium on Security & Privacy.
3. The 2018 IEEE European Symposium on Security & Privacy.
4. The 24th ACM Conference on Computer and Communications Security (CCS 2017).
5. The 26th USENIX Security Symposium (Security '17).
6. Co-chair, Security and Privacy Track, WWW 2017.
7. Annual Network and Distributed System Security Symposium (NDSS) 2017.
8. The 25th USENIX Security Symposium (Security '16).
9. The 23rd ACM Conference on Computer and Communications Security (CCS 2016).
10. The 2016 IEEE European Symposium on Security & Privacy.
11. Annual Network and Distributed System Security Symposium (NDSS) 2016.
12. The 22nd ACM Conference on Computer and Communications Security (CCS 2015).
13. Co-chair, Security and Privacy Track, WWW 2015.
14. Annual Network and Distributed System Security Symposium (NDSS) 2015.
15. European Symposium on Research in Computer Security (ESORICS) 2014.
16. The 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2014).
17. Security and Privacy Track, WWW 2014.

18. *PC Chair*, The 2013 IEEE Symposium on Security and Privacy.
19. The 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 2013).
20. The 20th ACM Conference on Computer and Communications Security (CCS 2013).
21. The 16th International Symposium on Recent Advances in Intrusion Detection (RAID 2013).
22. The 19th ACM Conference on Computer and Communications Security (CCS 2012).
23. The 9th Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DMIVA 2012).
24. The 7th USENIX Workshop on Hot Topics in Security (HotSec '12).
25. *PC co-Chair*, The 2012 IEEE Symposium on Security and Privacy.
26. Financial Cryptography and Data Security (FC 2012).
27. The 19th Annual Network and Distributed System Security Symposium (NDSS 2012).
28. The 8th Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA 2011).
29. The 41st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2011).
30. The 20th USENIX Security Symposium (Security '11).
31. The 6th USENIX Workshop on Hot Topics in Security (HotSec '11).
32. The 2011 IEEE Symposium on Security and Privacy.
33. The 13th International Symposium on Recent Advances in Intrusion Detection (RAID 2010).
34. The 15th European Symposium on Research in Computer Security (ESORICS 2010).
35. The 7th Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA 2010).
36. Security and Privacy Track, ICDCS 2010.
37. The 40th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2010).
38. Security and Privacy Track, WWW 2010.
39. *PC Chair*, The 17th Annual Network and Distributed System Security Symposium (NDSS 2010).
40. Financial Cryptography and Data Security (FC 2010).

41. The 25th Annual Computer Security Applications Conference (ACSAC 2009).
42. The 16th ACM Conference on Computer and Communications Security (CCS 2009).
43. 14th European Symposium on Research in Computer Security (ESORICS 2009).
44. The 18th USENIX Security Symposium (Security '09).
45. *PC Chair*, The Second USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET), 2009.
46. *PC Co-Chair*, The Second ACM Conference on Wireless Network Security (WiSec), 2009.
47. The 2009 IEEE Symposium on Security and Privacy.
48. The 16th Annual Network and Distributed System Security Symposium (NDSS 2009).
49. The 24th Annual Computer Security Applications Conference (ACSAC 2008).
50. Internet Measurement Conference (IMC) 2008.
51. 13th European Symposium on Research in Computer Security (ESORICS 2008).
52. The 15th ACM Conference on Computer and Communications Security (CCS 2008).
53. The 17th USENIX Security Symposium (Security '08).
54. 3rd USENIX Workshop on Hot Topics in Security (HotSec '08).
55. The 2008 IEEE Symposium on Security and Privacy.
56. The 15th Annual Network and Distributed System Security Symposium (NDSS 2008).
57. The 23rd Annual Computer Security Applications Conference (ACSAC 2007).
58. The 14th ACM Conference on Computer and Communications Security (CCS 2007).
59. The 10th International Symposium on Recent Advances in Intrusion Detection (RAID 2007).
60. The 16th USENIX Security Symposium (Security '07).
61. The 2007 IEEE Symposium on Security and Privacy.
62. The 14th Annual Network and Distributed System Security Symposium (NDSS 2007).
63. The 2nd International Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm 2006).
64. The 15th USENIX Security Symposium (Security '06).
65. The 13th Annual Network and Distributed System Security Symposium (NDSS 2006).

66. The 1st International Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm 2005).
67. The 2005 IEEE Symposium on Security and Privacy.
68. The 2005 ACM Workshop on Wireless Security (WiSe 2005).
69. The 2005 IEEE International Conference on Data Mining (ICDM 2005).
70. The Fourteenth International World Wide Web Conference (WWW 2005).
71. The 2005 International Conference on Distributed Computing Systems (ICDCS).
72. The 2004 ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC 2004).
73. The 2004 ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN 2004).
74. The 2004 ACM Workshop on Wireless Security (WiSe 2004).
75. Steering Committee and the 7th International Symposium on Recent Advances in Intrusion Detection (RAID 2004).
76. The 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2004.
77. The 2004 SIAM International Conference on Data Mining, 2004.
78. The 2004 IEEE International Conference on Data Mining (ICDM 2004).
79. The 2004 IEEE Information Assurance Workshop.
80. Co-Chair of the the ICDM Workshop on Data Mining for Computer Security (DMSEC 2003).
81. The 2003 IEEE International Conference on Data Mining (ICDM 2003).
82. The 2003 ACM Workshop on Wireless Security (WiSe 2003).
83. Steering Committee and the 6th International Symposium on Recent Advances in Intrusion Detection (RAID 2003).
84. The 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003.
85. The 2003 IEEE Information Assurance Workshop.
86. Steering Committee for the 5th International Symposium on Recent Advances in Intrusion Detection (RAID 2002).
87. The 2002 ACM New Security Paradigms Workshop.
88. The 2002 IEEE Symposium on Security and Privacy.

89. *PC Co-Chair*, The 4th International Symposium on Recent Advances in Intrusion Detection (RAID 2001).
90. Program Committee and Organizational Committee for the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2000.

B. On-campus Georgia Tech Committees

1. 5-Year Review Committee for the Chair of School of Computer Science, College of Computing, 2016.
2. Faculty Recruiting Committee, School of Computer Science, 2014.
3. Faculty Recruiting Committee, School of Computer Science, 2012-13.
4. RPT Committee, College of Computing, 2010-2012.
5. College of Computing Dean Search Committee, 2009-2010, College of Computing.
6. Chair, Faculty Recruiting Committee, 2008-2009, School of Computer Science, College of Computing.
7. Chair Advisory Committee, 2008-2009, School of Computer Science, College of Computing.
8. Co-Chair, Faculty Recruiting Committee, 2007-2008, School of Computer Science, College of Computing.
9. RPT Executive Committee, 2007-2008, College of Computing.
10. Faculty Recruiting Committee, 2006-2007, College of Computing.
11. Faculty Recruiting Committee, 2005-2006, College of Computing.
12. Undergraduate Curriculum Task Force, 2004-2005, College of Computing.
13. Ph.D. Admissions Committee, 2003-2004, College of Computing.
14. Faculty Recruiting Committee, 2001-2003, College of Computing.

E. Consulting and Advisory Appointments

1. Secure, Accessible & Fair Elections (SAFE) Commission, State of Georgia, 2018-2019
2. Advisory Board of the Faculty of Engineering, The Chinese University of Hong Kong. 2017-
3. Board of Trustees, Pace Academy. 2017-

IV. NATIONAL AND INTERNATIONAL PROFESSIONAL RECOGNITION

A. Patents

1. “Method and system for network-based detecting of malware from behavioral clustering”. Roberto Perdisci, Wenke Lee, and Gunter Ollmann. U.S. Patent Number: 8,826,438, September 2014.
2. “Method and system for detecting malicious domain names at an upper DNS hierarchy”. Manos Antonakakis, Roberto Perdisci, Wenke Lee, and Nikolaos Vasiloglou. U.S. Patent Number: 8,631,489, January 2014.
3. “Method and System for Detecting Malware”. Manos Antonakakis, Roberto Perdisci, Wenke Lee, and Gunter Ollmann. U.S. Patent Number: 8,578,497, November 2013.
4. “Method and System for Detecting and Responding to Attacking Networks”. David Dagon, Nick Feamster, Wenke Lee, Robert Edmonds, Richard Lipton, and Anirudh Ramachandran. U.S. Patent Number: 8,566,928, October 2013.
5. “Methods for Cost-Sensitive Modeling for Intrusion Detection and Response”. Wei Fan, Wenke Lee, Matt Miller, and Sal Stolfo. U.S. Patent Number: 7818797, 2010.
6. “Method and System for Using Intelligent Agents for Financial Transactions, Services, Accounting, and Advice”. Dan Schutzer, Will Foster, Huanrui Hu, Wenke Lee, Sal Stolfo, and Wei Fan. U.S. Patent Number: 5920848, 1999.

V. OTHER CONTRIBUTIONS

A. Seminar Presentations

1. "Security Overlay", Stony Brook University, March 2016.
2. "Security Overlay", Princeton University, October 2015.
3. "Internet Monitoring via DNS Traffic Analysis", Google-UMD Cybersecurity Seminar, November 2012.
4. "Botnet Detection and Response: Challenges and Opportunities", Carnegie Mellon University, September 2007.
5. "Botnet Detection and Response", Indiana University, March 2007.
6. "Botnet Detection and Response", University of Pittsburgh, October 2006.
7. "Architecture Considerations for Anomaly Detection", University of Texas at San Antonio, November, 2005
8. "Advanced Intrusion Detection Techniques", National Security Research Institute, Daejeon, South Korea, February, 2005
9. "U.S. Policy and Recent Detection Techniques for Cyber Threats and Attacks", National Cyber Security Center, Seoul South Korea, February, 2005
10. "Architecture Considerations for Anomaly Detection", Purdue University, February, 2005
11. "Architecture Considerations for Anomaly Detection", Graduate Center of The City University of New York, February, 2005
12. "Architecture Considerations for Anomaly Detection", The University of Alabama, November, 2004
13. "Architecture Considerations for Anomaly Detection", Carnegie Mellon University, September, 2004
14. "Architecture Considerations for Anomaly Detection", University of Wisconsin at Madison, August, 2004
15. "Local Worm Detection and Response: Algorithms and Analytical Models", Lawrence Livermore National Laboratory, June 2004
16. "Intrusion Detection", Carnegie Mellon University, October 2003
17. "Algorithms for Recognizing New Attack Step Relationships", University of Pennsylvania, September 2003
18. "Intrusion Detection", Emory University, October 2001

19. "Developing Data Mining Techniques for Intrusion Detection: A Progress Report", Center for Education and Research in Information Assurance and Security, Purdue University, October 2000
20. "A Data Mining Framework for Building Intrusion Detection Models", Institute for Security Technology Studies, Dartmouth College, Hanover, NH, July 2000
21. "A Data Mining Framework for Building Intrusion Detection Models", Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA, February 2000
22. "A Data Mining Framework for Building Intrusion Detection Models", HRL Laboratories, Malibu, CA, November 1999