

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CROWDSTRIKE, INC.
Petitioner

v.

WEBROOT, INC.,
Patent Owner

Case IPR2023-01158
Patent No. 8,181,244

DECLARATION OF DR. MARKUS JAKOBSSON

TABLE OF CONTENTS

I.	Introduction.....	6
II.	Background and Qualifications	7
III.	Level of Ordinary Skill in the Art.....	16
IV.	Materials Considered and Relied Upon	16
V.	Legal Standards.....	18
VI.	Overview of the '244 Patent	25
	A. Brief Description of the '244 Patent	25
	B. File History of the '244 Patent	27
	C. Interpretation of the '244 Patent Claims at Issue	30
VII.	Overview of the Cited References	31
	A. Hartrell (EX1004).....	31
	B. Fan (EX1005).....	34
	C. An Obvious Combination of Hartrell and Fan.....	35
	D. Rockwood (EX1006).....	37
	E. An Obvious Combination of Hartrell, Fan, and Rockwood	38
	F. Polyakov (EX1007).....	41
	G. An Obvious Combination of Hartrell, Fan, Rockwell, and Polyakov	42
VIII.	Overview of Unpatentability Grounds.....	44
IX.	Ground 1: Hartrell, Fan, and Rockwood REnde Obvious Claims 1-10 and 13-16	44

A.	Claim 1	44
1.	[1pre] A method for identifying an origin of suspected pestware activity on a computer, the method comprising:	45
2.	[1a] monitoring, with a kernel-mode driver, activity on the computer;.....	46
3.	[1b] generating an activity log on a file storage device of the computer from the kernel-mode driver;.....	47
4.	[1c] receiving, from a user via an interface of the computer, a time of interest relating to a suspicion of pestware on the computer, wherein the time of interest includes a time interval;	49
5.	[1d] issuing a timestamp after receiving the time of interest;.....	55
6.	[1e] identifying, based upon the time of interest, indicia of pestware, wherein the identifying is initiated by the issuing the timestamp;.....	56
7.	[1f] accessing, using a hardware processor of the computer, at least a portion of a recorded history of externally networked sources that the computer received files from	58
8.	[1g] so as to identify, based at least in part upon the identified indicia of pestware, a reference to an identity of an externally networked source that is suspected of originating pestware.....	60
9.	[1h] wherein the recorded history of externally networked sources is stored on the file storage device.....	61
B.	Claim 2	62
1.	[2a] The method of claim 1, wherein the identifying includes: accessing at least one log of historical events on the computer.....	62
2.	[2b] locating at least one event in the at least one log that corresponds to the time of interest; and.....	64
3.	[2c] determining that the at least one event is associated with the indicia of pestware.	65

C.	Claim 3	65
D.	Claim 4	68
E.	Claim 5	69
F.	Claim 6	71
G.	Claim 7	72
H.	Claim 8	74
I.	Claim 9	76
1.	[9pre] A system for identifying a source of suspected pestware on a computer, the system comprising:	76
2.	[9a] a hardware processor	77
3.	[9b] a file storage device coupled to the hardware processor;	77
4.	[9c.i] a timestamp module configured to be executed by the hardware processor, to generate a time of interest including a time interval in response to detected activity on the computer that is indicative of pestware,	78
5.	[9c.ii] issuing a timestamp after receiving the time of interest;	78
6.	[9c.iii] to initiate, in response to the issuance of the timestamp, identification of indicia of pestware;	78
7.	[9d] a research portion configured to be executed by the hardware processor and to access at least one log on a file storage device of the computer to relate the time of interest to at least one externally networked source of potential pestware activity on the computer;	78
8.	[9e] a kernel-mode driver configured to monitor activity on the computer and to generate an activity log that is included within the at least one log; and	78
9.	[9f] a reporting portion configured to be executed by the hardware processor and to generate a report that identifies the at least one externally networked source of potential pestware activity.	79

J.	Claim 10	80
K.	Claim 13	80
L.	Claim 14	80
M.	Claim 15	81
N.	Claim 16	81
X.	Ground 1B: HARTRELL, FAN, ROCKWOOD, AND POLYAKOV	
	Render Obvious Claims 11 and 12	82
A.	Claim 11	82
B.	Claim 12	84
XI.	Additional Remarks	86

I. INTRODUCTION

1. My name is Dr. Markus Jakobsson, and I have been retained by counsel for Petitioner CrowdStrike, Inc. (“CrowdStrike” or “Petitioner”) as an expert witness to provide assistance regarding U.S. Patent No. 8,181,244 (“the ’244 Patent”). Specifically, I have been asked to consider the patentability of claims 1-16 of the ’244 Patent (“Challenged Claims”) in view of prior art, anticipation and obviousness considerations, and the understanding of a person of ordinary skill in the art at the time of the invention (“POSITA”) as it relates to the ’244 Patent.

2. I am being compensated for my time at my standard consulting rate. I am also being reimbursed for expenses that I incur during the course of this work. My compensation is not contingent upon the results of my study, the substance of my opinions, or the outcome of any proceeding involving the Challenged Claims. I have no financial interest in the outcome of this matter or on the pending litigation between Petitioner and Patent Owner.

3. My analysis here is based on my years of education, research and experience, as well as my investigation and study of relevant materials, including those cited herein.

4. I may rely upon these materials, my knowledge and experience, and/or additional materials to rebut arguments raised by the Patent Owner. Further, I may

also consider additional documents and information in forming any necessary opinions, including documents that may not yet have been provided to me.

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

II. BACKGROUND AND QUALIFICATIONS

6. I have summarized in this section my educational background, career history, and other qualifications relevant to this matter. I have also included a current version of my curriculum vitae, which is attached as Appendix A.

7. I received a Master of Science degree in Computer Engineering from the Lund Institute of Technology in Sweden in 1993, a Master of Science degree in Computer Science from the University of California at San Diego in 1994, and a Ph.D. in Computer Science from the University of California at San Diego in 1997, specializing in Cryptography. During and after my Ph.D. studies, I was also a researcher at the San Diego Supercomputer Center and General Atomics, where I did research on authentication, privacy, and security. Much of my work was on distributed protocols to enable security and privacy, and my Ph.D. thesis described a distributed electronic payment system. Whereas the term “cloud computing” had

not been popularized at this point, many of the principles I studied later became critical aspects of cloud computing and virtualization.

8. Since earning my doctorate degree over twenty-six years ago, I have been an employee, entrepreneur, and consultant in the computer systems, security, and cloud computing industry. I have worked extensively with technologies related to computer security, mobile security, malware detection, quantitative and qualitative fraud analysis, and disruptive security. I have worked as an engineer or technical advisor at a number of leading companies in the computer industry, including PayPal, Qualcomm, Bell Labs, and LifeLock. I have also taught and performed research at several prestigious universities, including New York University (NYU) and Indiana University (IU). Further, I have been materially involved in designing, developing, testing, and assessing technologies for computer and network security, digital rights management, trust establishment, randomness, cryptography, socio-technical fraud, malware detection, detection of malicious emails, user interfaces, authentication, and fraud detection for over twenty years.

9. From 1997 to 2001, I was a Member of Technical Staff at Bell Labs, where I did research on authentication, privacy, multi-party computation, contract exchange, digital commerce including crypto payments, and fraud detection and prevention. Example publications of mine from this time period include “Secure distributed computation in cryptographic applications” (U.S. Patent No. 6,950,937,

2001 priority date) and “Secure server-aided signature generation” (International Workshop on Public Key Cryptography PKC 2001: Public Key Cryptography 383-401), both of which relate to virtualization. At the end of my tenure at Bell Labs, I had started to research identity theft, years before the banking industry became aware of the existence of phishing attacks.

10. From 2001 to 2004, I was a Principal Research Scientist at RSA Labs, where I worked on predicting future fraud scenarios in commerce and authentication and developed solutions to those problems. Much of my work related to the development of anti-fraud mechanisms, addressing authentication abuse and identity theft. During that time, I predicted the rise of what later became known as phishing. I also laid the groundwork for what I termed “proof of work,” and described how this could be applied in the context of distributed computations such as mining of crypto payments; this work later came to influence the development of BitCoin. I was also an Adjunct Associate Professor in the Computer Science department at New York University from 2002 to 2004, where I taught cryptographic protocols, with an emphasis on authentication.

11. From 2004 to 2016, I held a faculty position at Indiana University at Bloomington, first as an Associate Professor of Computer Science, Associate Professor of Informatics, Associate Professor of Cognitive Science, and Associate Director of the Center for Applied Cybersecurity Research (CACR) from 2004 to

2008; and then as an Adjunct Associate Professor from 2008 to 2016. I was the most senior security researcher at Indiana University, where I built a research group focused on online fraud and countermeasures, resulting in over 50 publications and two books. One of these books, “Crimeware: Understanding New Attacks and Defenses” (Wiley, 2008), described man-in-the-middle attacks, online abuse in general, and countermeasures to attacks and abuses of these types.

12. While a professor at Indiana University, I was also employed by Xerox PARC, PayPal, and Qualcomm to provide thought leadership to their security groups. I was a Principal Scientist at Xerox PARC from 2008 to 2010, a Director and Principal Scientist of Consumer Security at PayPal from 2010 to 2013, a Senior Director at Qualcomm from 2013 to 2015, Chief Scientist at Agari from 2016 to 2018, and Chief of Security and Data Analytics at Amber Solutions from 2018 to 2020.

13. Agari is a cybersecurity company that develops and commercializes technology to protect enterprises, their partners and customers from advanced email phishing attacks. At Agari, my research studied and addressed trends in online fraud, especially as related to email, including problems such as Business Email Compromise, Ransomware, and other abuses based on social engineering and identity deception. My work primarily involved identifying trends in fraud and computing before they affected the market, and developing and testing

countermeasures, including technological countermeasures, user interaction and education. Example publications include “Detecting computer security risk based on previously observed communications” (U.S. Patent No. 10,715,543 B2, 2016 priority date) and “Using message context to evaluate security of requested data” (U.S. Patent No. 10,805,314 B2, 2017 priority date).

14. In 2020, after leaving Agari, I was the Chief of Security and Data Analytics at Amber Solutions, an IoT startup in the San Francisco Bay Area. Amber Solutions is a cybersecurity company that develops home and office automation technologies. At Amber Solutions, my research addressed privacy, user interfaces and authentication techniques in the context of ubiquitous and wearable computing. My work often related to authentication and cloud computing, as exemplified in “Configuration and management of smart nodes with limited user interfaces” (U.S. Patent No. 10,887,447).

15. I left Amber Solutions to take the role of Chief Scientist at ByteDance in 2019, where I guided the security research both for ByteDance and TikTok until I left to co-found Artema Labs, where I am the Chief Scientist. My work at ByteDance involved identifying new threats, developing new solutions, and guiding research in areas relating to cryptography and Internet security.

16. I am also the Chief Technology Officer at ZapFraud, a cybersecurity company that develops techniques to detect deceptive emails, such as Business

Email Compromise (“BEC”) emails. At ZapFraud, my research studies and addresses computer abuse, including social engineering, malware and privacy intrusions, and has also involved studying security aspects related to cloud computing. My work primarily involves identifying risks, developing protocols and user experiences, and evaluating the security of proposed approaches. Using virtual machines is an important tool for security researchers, as it allows them to study the impact of a given executable, website, or other context on a computational environment. My students, direct reports and I have often relied on this approach in the context of interacting with online criminals, their websites and emails they send, as virtual machines enable us to identify their motives and report them to the authorities without making us easy for them to track or identify. Doing this is important to identify new and trending attacks.

17. I also work as an independent security researcher and consultant in the fields of computer/mobile security, fraud detection/prevention, digital rights management, malware, phishing, crimeware, mobile malware, and cryptographic protocols. In addition, I am a visiting research fellow of the Anti-Phishing Working Group, an international consortium focused on unifying the global response to cybercrime in the public and private sectors.

18. I have also served on the advisory board of several companies, including Metaforic, a company that developed technology for watermarking

software and provided security software with a built-in framework that resists attacks from malware, bots, hackers and other attempts. Metaforic was acquired by Inside Secure in 2014.

19. I have founded or co-founded several successful computer security companies. In 2005 I co-founded RavenWhite Security, a provider of authentication solutions, and I am currently its Chief Technical Officer. RavenWhite owns intellectual property related to authentication, such as “Performing authentication” (U.S. Patent No. 10,079,823 2007 priority date). In 2007 I co-founded Extricatus, one of the first companies to address consumer security education. In 2009 I founded FatSkunk, a provider of mobile malware detection software; I served as Chief Technical Officer of FatSkunk from 2009 to 2013, when FatSkunk was acquired by Qualcomm and I became a Qualcomm employee. FatSkunk developed anti-virus technology. In 2013 I founded ZapFraud—the same company described above, where I currently serve as Chief Technical Officer. In 2014 I co-founded RightQuestion, a security consulting company.

20. I have additionally served as a member of the fraud advisory board at LifeLock (an identity theft protection company); a member of the technical advisory board at CellFony (a mobile security company); a member of the technical advisory board at PopGiro (a user reputation company); a member of the technical advisory board at MobiSocial dba Omlet (a social networking company); and a member of

the technical advisory board at Stealth Security (an anti-fraud company). I have provided anti-fraud consulting to KommuneData (a Danish government entity), J.P. Morgan Chase, PayPal, Boku, and Western Union.

21. I have authored seven books and over 100 peer-reviewed publications, and have been a named inventor on over 200 patents and patent applications. A large number of these relate to Internet security, authentication, malware, spoofing and related topics.

22. Examples of textbooks I have authored or co-authored on the topic of Internet security include:

- “Crimeware: Understanding New Attacks and Defenses” (Symantec Press, 2008)
- “Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft” (Wiley, 2006);
- “The Death of the Internet” (Wiley, 2012);
- “Understanding Social Engineering Based Scams” (Springer 2016); and
- “Towards Trustworthy Elections: New Directions in Electronic Voting” (Springer, 2010)

23. I have authored numerous publications regarding Internet security, including Internet security as it pertains to web browsers. Relevant examples include:

- “Web Camouflage: Protecting Your Clients from Browser-Sniffing Attacks,”
5 IEEE Security & Privacy (Nov. 2007)
- “Invasive Browser Sniffing and Countermeasures,” Conference: 15th
International Conference on World Wide Web (May 2006)
- “Remote Harm-Diagnostics”, (PDF) Privacy-preserving history mining for
web browsers (researchgate.net)
- “Privacy-Preserving History Mining for Web Browsers”, (PDF) Privacy
preserving history mining for web browsers (researchgate.net)
- “Server-Side Detection of Malware Infection”, NSPW Proceedings on New
Security Paradigms Workshop (2009), Server-side detection of malware
infection | Proceedings of the 2009 workshop on New Security Paradigms
Workshop (acm.org)

24. Related patent publications include U.S. Patent Nos. 10,498,735, 9,203,835, 8,578,174. These patents were assigned to PayPal, Inc. and Palo Alto Research Center Incorporated arising out of my employment.

25. In sum, I have extensive experience both as a developer and a researcher relating to system monitoring, software security, attack detection, and network security.

III. LEVEL OF ORDINARY SKILL IN THE ART

26. In my opinion, a person of ordinary skill in the art (“POSITA”) relating to, and at the time of, the ’244 Patent’s Critical Date (i.e., April 20, 2006) would have had a bachelor’s degree in computer science, computer engineering, or an equivalent, as well as two years of industry experience and would have had a working knowledge of endpoint monitoring systems, software security analysis, and dynamic malware analysis. Additional education in a relevant field, or relevant industry experience may compensate for a deficit in one of the other aspects of the requirements stated above.

27. Based on my knowledge, skill, and experience, I have a good understanding of the capabilities of the POSITA. Indeed, I have taught, participated in organizations with, hired, and worked closely with many such persons over the course of my career. For example, from my industry experience, I am familiar with what an engineer would have known and found predictable in the art. From teaching and supervising post-graduate students, I also have an understanding of the knowledge that a person with this academic experience possesses. Furthermore, I possess those capabilities myself.

IV. MATERIALS CONSIDERED AND RELIED UPON

28. In reaching the conclusions described in this declaration, I have relied on the documents and materials cited herein as well as those identified in this

declaration, including the '244 Patent, the prosecution history of the '244 Patent, and prior art references cited herein. These materials comprise patents, related documents, and printed publications. Each of these materials is a type of document that experts in my field would have reasonably relied upon when forming their opinions.

29. I have also relied on my education, training, research, knowledge, and personal and professional experience in the relevant technologies and systems that were already in use prior to and within the timeframe of the earliest asserted priority date of the claimed subject matter in the '244 Patent, which I have been informed by counsel is April 20, 2006.

- EX1001: U.S. Patent No. 8,181,244
- EX1002: Excerpts from the Prosecution History of the '244 Patent (“the Prosecution History”)
- EX1004: U.S. Patent Pub. No. 2007/0150957 to Gregory D. Hartrell *et al.* (“Hartrell”)
- EX1005: U.S. Patent No. 7,784,098 to Paul Fan *et al.* (“Fan”)
- EX1006: U.S. Patent No. 7,352,280 to Troy Dean Rockwood *et al.* (“Rockwood”)
- EX1007: U.S. Patent No. 7,571,482 to Alexey A. Polyakov *et al.* (“Polyakov”)
- EX1009: U.S. Patent No. 8,516,583 to Anil Francis Thomas *et al.* (“Thomas”)
- EX1010: U.S. Patent No. 7,549,055 to Vincent J. Zimmer *et al.* (“Zimmer”)

- EX1014: *Webroot, Inc. v. CrowdStrike, Inc.*, Case No. 6:22-cv-00241-ADA, Docket No. 263, “Opening Claim Construction Brief Regarding Patents Plaintiffs Webroot And OpenText Added By Amendment” (WDTX Apr. 12, 2023)
- EX1015: *Webroot, Inc. v. CrowdStrike, Inc.*, Case No. 6:22-cv-00241-ADA, Docket No. 304, “Plaintiffs’ Responsive Claim Construction Brief Regarding Patents Plaintiffs Added By Amendment” (WDTX May 5, 2023)

V. LEGAL STANDARDS

30. I am a technical expert and do not offer any legal opinions. However, counsel has informed me as to certain legal principles regarding patentability and related matters under United States patent law, which I have applied in performing my analysis and arriving at my technical opinions in this matter.

31. I have been informed that claim terms are to be given the meaning they would have to a person having ordinary skill in the art at the time of the invention, taking into consideration the patent, its file history, and, secondarily, any applicable extrinsic evidence (e.g., dictionary definitions).

32. I have also been informed that the implicit or inherent disclosures of a prior art reference may anticipate the claimed invention. Specifically, if a person having ordinary skill in the art at the time of the invention would have known that the claimed subject matter is necessarily present in a prior art reference, then the prior art reference may “anticipate” the claim. Therefore, a claim is “anticipated” by

the prior art if each and every limitation of the claim is found, either expressly or inherently, in a single item of prior art.

33. I have also been informed that a person cannot obtain a patent on an invention if the differences between the invention 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. I have been informed that a conclusion of obviousness may be founded upon more than a single item of prior art. I have been further informed that obviousness is determined by evaluating the following factors: (1) the scope and content of the prior art, (2) the differences between the prior art and the claim at issue, (3) the level of ordinary skill in the pertinent art, and (4) secondary considerations of non-obviousness. In addition, the obviousness inquiry should not be done in hindsight. Instead, the obviousness inquiry should be done through the eyes of a person having ordinary skill in the relevant art at the time the patent was filed.

34. In considering whether certain prior art renders a particular patent claim obvious, counsel has informed me that I can consider the scope and content of the prior art, including the fact that one of skill in the art would regularly look to the disclosures in patents, trade publications, journal articles, industry standards, product literature and documentation, texts describing competitive technologies, requests for comment published by standard setting organizations, and materials

from industry conferences, as examples. I have been informed that for a prior art reference to be proper for use in an obviousness analysis, the reference must be “analogous art” to the claimed invention. I have been informed that a reference is analogous art to the claimed invention if: (1) the reference is from the same field of endeavor as the claimed invention (even if it addresses a different problem); or (2) the reference is reasonably pertinent to the problem faced by the inventor (even if it is not in the same field of endeavor as the claimed invention). In order for a reference to be “reasonably pertinent” to the problem, it must logically have commended itself to an inventor's attention in considering his problem. In determining whether a reference is reasonably pertinent, one should consider the problem faced by the inventor, as reflected either explicitly or implicitly, in the specification. I believe that all of the references that my opinions in this IPR are based upon are well within the range of references a person having ordinary skill in the art would consult to address the type of problems described in the Challenged Claims.

35. I have been informed that, in order to establish that a claimed invention was obvious based on a combination of prior art elements, a clear articulation of the reason(s) why a claimed invention would have been obvious must be provided. Specifically, I am informed that a combination of multiple items of prior art renders a patent claim obvious when there was an apparent reason for one of ordinary skill in the art, at the time of the invention, to combine the prior art, which can include,

but is not limited to, any of the following rationales: (A) combining prior art methods according to known methods to yield predictable results; (B) substituting one known element for another to obtain predictable results; (C) using a known technique to improve a similar device in the same way; (D) applying a known technique to a known device ready for improvement to yield predictable results; (E) trying a finite number of identified, predictable potential solutions, with a reasonable expectation of success; (F) identifying that known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art; or (G) identifying an explicit teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine the prior art references to arrive at the claimed invention.

36. I am informed that the existence of an explicit teaching, suggestion, or motivation to combine known elements of the prior art is a sufficient, but not a necessary, condition to a finding of obviousness. This so-called “teaching suggestion-motivation” test is not the exclusive test and is not to be applied rigidly in an obviousness analysis. In determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the patentee controls. Instead, the important consideration is the objective reach of the claim. In other words, if the claim extends to what is obvious, then the claim is

invalid. I am further informed that the obviousness analysis often necessitates consideration of the interrelated teachings of multiple patents, the effects of demands known to the technological community or present in the marketplace, and the background knowledge possessed by a person having ordinary skill in the art. All of these issues may be considered to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent.

37. I also am informed that in conducting an obviousness analysis, a precise teaching directed to the specific subject matter of the challenged claim need not be sought out because it is appropriate to take account of the inferences and creative steps that a person of ordinary skill in the art would employ. The prior art considered can be directed to any need or problem known in the field of endeavor at the time of invention and can provide a reason for combining the elements of the prior art in the manner claimed. In other words, the prior art need not be directed towards solving the same specific problem as the problem addressed by the patent. Further, the individual prior art references themselves need not all be directed towards solving the same problem. I am informed that common sense is important and should be considered. Common sense teaches that familiar items may have obvious uses beyond their primary purposes.

38. I also am informed that the fact that a particular combination of prior art elements was “obvious to try” may indicate that the combination was obvious

even if no one attempted the combination. If the combination was obvious to try (regardless of whether it was actually tried) or leads to anticipated success, then it is likely the result of ordinary skill and common sense rather than innovation. I am further informed that in many fields it may be that there is little discussion of obvious techniques or combinations, and it often may be the case that market demand, rather than scientific literature or knowledge, will drive the design of an invention. I am informed that an invention that is a combination of prior art must do more than yield predictable results to be non-obvious.

39. I am informed that for a patent claim to be obvious, the claim must be obvious to a person of ordinary skill in the art at the time of the invention. I am informed that the factors to consider in determining the level of ordinary skill in the art include (1) the educational level and experience of people working in the field at the time the invention was made, (2) the types of problems faced in the art and the solutions found to those problems, and (3) the sophistication of the technology in the field.

40. I am informed that it is improper to combine references where the references teach away from their combination. I am informed that a reference may be said to teach away when a person of ordinary skill in the relevant art, 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. In general, a reference will teach away if it suggests that the line of development flowing from the reference's disclosure is unlikely to be productive of the result sought by the patentee. I am informed that a reference teaches away, for example, if (1) the combination would produce a seemingly inoperative device, or (2) the references leave the impression that the product would not have the property sought by the patentee. I also am informed, however, that a reference does not teach away if it merely expresses a general preference for an alternative invention but does not criticize, discredit, or otherwise discourage investigation into the invention claimed.

41. I am informed that the final determination of obviousness must also consider "secondary considerations" if presented. In most instances, the patentee raises these secondary considerations of non-obviousness. In that context, the patentee argues an invention would not have been obvious in view of these considerations, which include: (a) commercial success of a product due to the merits of the claimed invention; (b) a long-felt, but unsatisfied need for the invention; (c) failure of others to find the solution provided by the claimed invention; (d) deliberate copying of the invention by others; (e) unexpected results achieved by the invention; (f) praise of the invention by others skilled in the art; (g) lack of independent simultaneous invention within a comparatively short space of time; (h) teaching away from the invention in the prior art.

42. I am further informed that secondary considerations evidence is only relevant if the offering party establishes a connection, or nexus, between the evidence and the claimed invention. The nexus cannot be based on prior art features. The establishment of a nexus is a question of fact. While I understand that the Patent Owner here has not offered any secondary considerations at this time, I will supplement my opinions if the Patent Owner raises secondary considerations during this proceeding.

VI. OVERVIEW OF THE '244 PATENT

A. Brief Description of the '244 Patent

43. The '244 Patent describes a “system and method for identifying an origin of suspected pestware activity on a computer.” EX1001, Abstract. FIGS. 2-4 of the '244 Patent largely relate to the collection of “historical logs” on a computer and the traversal of these historical logs to identify the “source” of suspected pestware. *See, e.g.*, EX1001, 6:1-9; 7:22-47; 8:5-31. However, as noted in the Abstract and Summary of the '244 Patent, the claims also focus on “establishing a time of interest relating to a suspicion of pestware on the computer.” *See* EX1001, Abstract, 2:19-26. Indeed, as I note in the next section, the “time of interest” features of the independent claims were a particular focus of the Applicant’s arguments in differentiating the prior art and of the Examiner’s allowance of the claims.

44. The '244 Patent primarily describes the provision and use of a “time of interest” with respect to FIG. 5. *See* EX1001, FIG. 5, 8:59-61 (“in this embodiment a time of interest is initially established based upon a suspicion that pestware has infected a computer”). The '244 Patent describes a couple of non-limiting examples of how a user might provide a time of interest. *See* EX1001, 8:61-9:4. “In some embodiments, for example, a user may establish the time of interest based upon events the user observed (e.g., pop-ups or a system crash).” EX1001, 8:61-63. Alternatively, “the user is provide[d] with an alert in response to the heuristics module 110 identifying an activity that is suspicious and the user is given an option to initiate research at or around the time of interest.” EX1001, 9:1-4. According to the '244 Patent’s description of FIG. 1, “once a time of interest is established, suspicious activity is identified on the computer based upon the time of interest.” EX1001, 9:9-11.

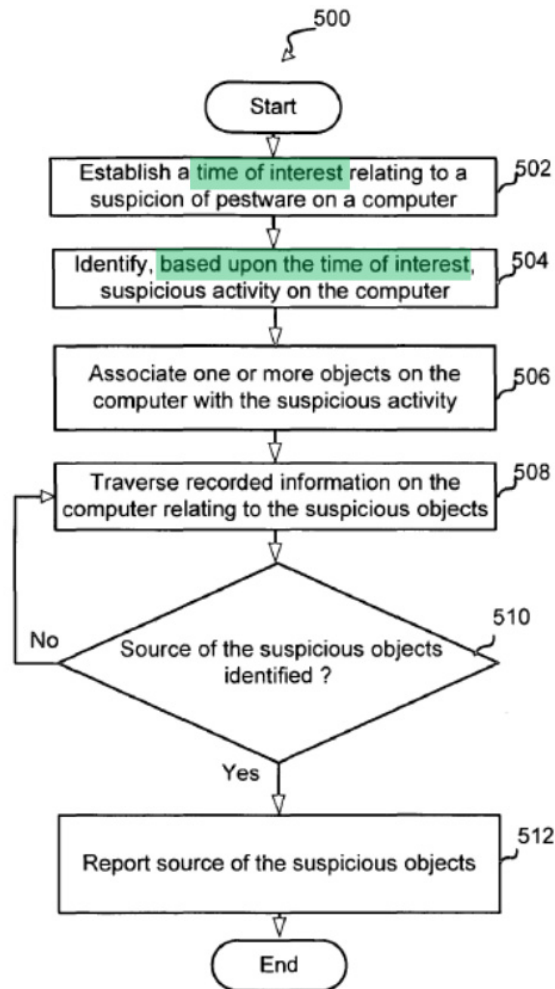


FIGURE 5

'244 Patent (EX1001), Highlighted FIG. 5

B. File History of the '244 Patent

45. The '244 Patent issued from U.S. Patent App. No. 11/408,145 (the '145 application). The original claims were initially rejected over U.S. Pub. No. 2004/0024864 (Porrás) in view of U.S. Pub. No. 2005/0268112 (Wang). EX1002, 206-215. In response, the Applicant amended independent claim 1 to specify that the claimed "time of interest" is received from a user, and amended then independent

claim 12 to specify that the “timestamp module” generates the claimed “time of interest” in response to “detected activity on the computer that is indicative of pestware.” *Id.*, 193-197.

46. The Examiner issued a final Office Action in which he added U.S. Pub. No. 2004/0225877 (Huang) to the combination of Porras and Wang to reject all pending claims. EX1002, 164-174. In response, the Applicant amended independent claims 1 and 12 to specify that the claimed “recorded history” is of “externally networked sources” and that this history is accessed so as to identify a reference to an identity of “an externally networked source.” *Id.*, 147.

47. In his next rejection of the pending claims, the Examiner relied on a combination of Porras and U.S. Pub. No. 2007/0168694 (Maddaloni). EX1002, 125-135. Applicant argued that Maddaloni was not prior art to the claims of the ’145 application (*id.*, 115), so the Examiner issued a new Office Action in which he replaced Maddaloni with U.S. Pub. No. 2007/0094725 (Borders) (*id.*, 89-98). The Applicant again amended the claims to include details about the computer of the method/system (e.g., that the “time of interest” of independent claim 1 is received via “an interface of the computer” and that the various steps of independent claims 1 and 12 were performed using a “hardware processor”). *Id.*, 65-69.

48. The Examiner next proposed and gained acceptance of an extensive Examiner's Amendment that led to allowance. *Id.*, 37-40. The amendment included several substantive additions to the claims, including:

- a “kernel-mode driver”
- that the “time of interest” includes a “time interval”
- that a “timestamp” is issued after receiving the “time of interest,” and
- that identification of indicia of pestware is “in response to the issuance of the timestamp.”

Id. In his “Reasons for Allowance,” the Examiner explained that “[n]one of the prior art of records [sic], either taken by itself or in any combination, would have anticipated or made obvious at or before the time the application was filed where ***a timestamp is issued after receiving the time of interest and the identifying of pestware is initiated by the issuing of timestamp wherein monitoring and generating an activity log is performed from the kernel-mode driver.***” *Id.*, 42.¹

49. However, these were common features of the prior art. For example, as will be explained below, Hartrell (EX1004) describes these features. Neither

¹ I have added emphasis to quotations throughout this declaration unless I note otherwise.

Hartrell nor any prior art like it was considered by the Examiner during original prosecution.

C. Interpretation of the '244 Patent Claims at Issue

50. I understand that the terms that appear in the claims of the '244 Patent should be interpreted according to their ordinary and customary meaning in light of the specification (EX1001) and prosecution file history (EX1002). In determining the ordinary and customary meaning, I understand that the words of a claim are first given their plain meaning that those words would have had to one of ordinary skill at the time of the alleged invention. I also understand that the structure of the claims, the specification and file history also may be used to better construe a claim insofar as the plain meaning of the claims cannot be understood. Moreover, I understand that even treatises and dictionaries may be used, albeit under limited circumstances, to determine the meaning attributed by one of ordinary skill to a claim term at the time of filing.

51. I understand that in parallel litigation, Patent Owner contends that no claim language requires express construction. For the purposes of my analysis, I have been instructed to apply the plain and ordinary meaning to the claim terms at issue herein unless noted otherwise. For these terms, based on the prior art's description of the claimed elements being similar to that of the '244 Patent specification, I understand that no formal claim constructions are necessary for these

terms in this proceeding because claim terms only need to be construed to the extent necessary to resolve a controversy.

52. I also understand that the words of the claims should be interpreted as they would have been interpreted by one of ordinary skill at the time the invention was made (not today). For purposes of my analysis here, I have used the April 20, 2006 purported priority date of the '244 Patent as the date of invention (i.e., the Critical Date). Without exception, however, my analysis of the proper meaning of the recited claim elements in this declaration would be correct if the date of invention was earlier or later than 2006.

VII. OVERVIEW OF THE CITED REFERENCES

53. The general descriptions that I have provided for the following references and obvious modifications thereof are incorporated into each subsection and mapping of the claims that includes citations to these references.

A. Hartrell (EX1004)

54. Hartrell describes a “malware analysis system for automating cause and effect analysis of malware infections.” EX1004, Abstract. “Upon being informed of a suspected malware infection, the malware analysis system creates a time-bounded snapshot of the monitored activities that were conducted within a time frame prior to the notification of the suspected malware infection.” *Id.* “For example, the malware analysis system can provide a snapshot of the last ten minutes

of the monitored activities (e.g., local system activity, network activity, etc.) prior to the detection of the infection.” *Id.*, [0014]. Hartrell describes analyzing the snapshot “to determine the ‘root cause’ or source of the infection.” *Id.*

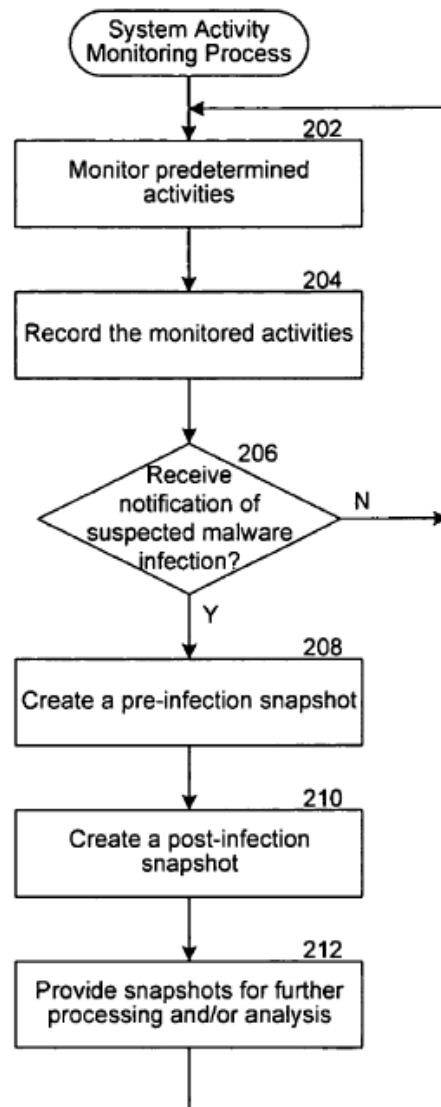


FIG. 2

Hartrell (EX1004), FIG. 2

55. In FIG. 2, Hartrell provides “a flow diagram that illustrates the processing of the system activity monitor component of the malware analysis system.” EX1004, [0007], [0027]. As shown in FIG. 2, Hartrell’s system “monitors predetermined activities on or about the computer system” (202) and records the monitored activity (204). EX1004, [0027]. The “system activity monitor component determines whether it received notification of a suspected malware infection” (206), and if “a notification of a suspected malware infection is received, then, in block 208, the system activity monitor component creates a pre-infection snapshot of the monitored activities.” *Id.*

56. “The system activity monitor component can use the report of the infection as the base time (e.g., t_0) from which to create the time-bounded snapshots of recorded activities.” *Id.*, [0030]. Starting at this t_0 timestamp of the reported infection, “the system activity monitor component may create a snapshot of the previous X seconds of the monitored activities that were recorded prior to receiving the notification event.” *Id.*, [0027]. “The snapshot of the activities can then be analyzed to determine the ‘root cause’ or source of the infection (i.e., the activities that led to the infection), including, for example, how the infected computer system was discovered, how the infected computer system was accessed, what operating system objects were manipulated, and the like.” *Id.*, [0014].

57. Hartrell describes that “the system activity monitor component may execute as a runtime process that may use any of a variety of *well-known* monitoring techniques *to monitor operating system* and/or network *activities*.” EX1004, [0027]. In other words, Hartrell does not limit the operation of the activity monitor component to any specific implementation, but instead leaves the details of its implementation to POSITAs. Fan provides details about how such an activity monitor may be configured to monitor operating system activities.

B. Fan (EX1005)

58. Like Hartrell, Fan describes software that “monitors system operations and registers an event when an operation potentially characteristic of a malware infection occurs.” EX1005, 2:35-37. Fan’s software includes a “behavior monitor module 210”² that “is real-time software including two functional kernel mode drivers that *monitor[] operating system operations*.” EX1005, 6:19-24. In other words, Fan describes a software module that performs the same type of monitoring as the activity monitor of Hartrell. *See* EX1004, [0027].

² Notably, Fan describes that behavior monitor module 210 preferably “operates in kernel mode.” EX1005, 6:19-25. For purposes of the combinations and mappings, I assume that behavior monitor module 210 operates in kernel mode consistent with Fan’s preferred embodiment.

59. Fan describes that “[m]odule 210 monitors and intercepts operating system operations such as file and registry access, process creation and process termination, and certain system APIs such as CreateProcess() CreateRemoteThread() and VirtualAlloc() [sic].” EX1005, 6:33-36. The module 210 can also intercept network traffic and events. *Id.*, 6:37-41. Fan teaches that some types of malware (e.g., rootkits) “hook into the operating system at the kernel level to hide their presence,” making them “very hard to detect.” *Id.*, 4:45-53. Because the behavior monitor module 210 includes two functional kernel mode drivers, it is “suitable for use when malware has infected a kernel mode driver.” *See id.* 3:10-13, 6:19-60.

C. An Obvious Combination of Hartrell and Fan

60. Like the '244 Patent, Hartrell and Fan both describe systems for monitoring computer systems for suspected malware. *See, e.g.*, EX1004, Abstract (“A malware analysis system for automating cause and effect analysis of malware infections is provided. The malware analysis system monitors and records computer system activities.”); EX1005, 2:35-37 (“In one embodiment the present invention monitors system operations and registers an event when an operation potentially characteristic of a malware infection occurs.”); EX1001, 1:36-39.

61. As described above, Hartrell describes a system activity monitor component 102 that “provides run-time *monitoring of the operating system*

resources for changes to the file system, common file formats, configurations (registry), network activities, use of common application program interfaces (APIs), or any other operating system object.” EX1004, [0019]. However, Hartrell does not limit or specify further implementation details for the operation of the system activity monitor component 102. Rather, Hartrell describes that “the system activity monitor component may execute as a runtime process that may use any of a variety of *well-known* monitoring techniques *to monitor operating system* and/or network *activities*.” EX1004, [0027]. Fan provides an example of one well-known technique for monitoring operating system activities.

62. Thus, a POSITA would have found it obvious to implement Hartrell’s system activity monitor component 102 using “functional kernel mode drivers,” as described by Fan. A POSITA would have been motivated to use “functional kernel mode drivers” to implement Hartrell’s system activity monitor component 102, because Fan teaches that they are “suitable for use when malware has infected a kernel mode driver,” whereas “[t]raditional cleanup software is unable to remove malware from such drivers.” EX1005, 3:10-13, 6:19-60. The kernel mode drivers of Fan are assumed to be “trusted software and thus can execute any instruction and reference any memory address.” EX1005, 6:24-32. As a result, Fan’s functional kernel mode drivers are able to provide access to the type of operating system data (*e.g.*, the registry, network activities, and other operating system objects) to which

Hartrell's system activity monitor component 102 requires access. *See* EX1004, [0019]; EX1005, 6:19-54. Further, a POSITA would have had a reasonable expectation of success in implementing this combination, because Hartrell describes that these techniques were "well known," and Fan gives specific implementation details for them. *See* EX1004, [0027]; EX1005, 6:19-54.

D. Rockwood (EX1006)

63. Rockwood describes a "system and method for intruder tracking using advanced correlation in a network security system." EX1006, 1:8-11. According to Rockwood, prior art systems would detect attacks by correlating "incoming data with stored data according to rules designed to detect specific patterns in network traffic, audit trails, and other data sources." EX1006, 1:22-26. However, these prior art systems did not "determine the likely identity of attackers." EX1006, 1:25-26. Thus, Rockwood describes a "method for correlating event information" that "reduces the number of operations required to correlate detected events with stored events, resulting in faster correlations," and that "enables an intrusion detection system to determine the likely identity of attackers." EX1006, 1:36-67.

64. Among other things, Rockwood describes a correlation engine 140 that "is operable to correlate detected events 110 to detect an attack" EX1006, 4:49-51. "Correlation engine 140 is further operable to correlate detected events 110 with attacker profiles 224 . . . to identify the source of an attack." EX1006, 4:51-54. The

correlation engine 140 utilizes a ruleset 152 “to process, correlate, aggregate, and/or filter event information of detected events 110.” EX1006, 5:16-20.

65. Rockwood also describes a console 50 that includes a graphical user interface 52 (GUI) (*see* EX1006, 5:39-59) through which “a user, such as an operator (not shown), may view and process the event information of detected events 110” (*see* EX1006, 5:27-34) and provide “proximity limits” used by the correlation engine 140 to correlate various detected events to identify an attack and attacker (*see* EX1006, 8:6-34). Proximity limits “define a portion of n-dimensional space surrounding” detected events and are used to correlate those events. *See* EX1006, 8:6-34. The proximity limits provided by the user through the console’s GUI may include, for example, a time period around a target event. *See* EX1006, 9:4-16. In this regard, Rockwood empowers the user of console 50 to provide input to the correlation engine 140 to refine the correlation process.

E. An Obvious Combination of Hartrell, Fan, and Rockwood

66. Hartrell describes that the “system activity monitor component can be notified of a malware infection.” EX1004, [0019]. “Upon receiving the notification, the system activity monitor component creates and provides *a time-bounded snapshot of activities that occurred before* and/or after *the notification of infection* to the other components of the malware analysis system.” *Id.* “For example, the system activity monitor component may create a snapshot of the previous X seconds

of the monitored activities that were recorded prior to receiving the notification event.” *Id.*, [0027].

67. Hartrell describes that its malware analysis system also includes a commonality analysis component 106 that analyzes “the normalized activities in the pre-infection snapshot and/or the post infection snapshot to find any recurring activities” so that “the malware analysis system can tailor and provide a recommendation for responding to the detected infection.” EX1004, [0018]. One function of “the commonality analysis component tags similar activities as being suspicious.” EX1004, [0035]. However, Hartrell does not specifically describe or otherwise limit the manner in which the commonality analysis component determines activities are “similar.” Rather, Hartrell describes that “[o]ne skilled in the art will appreciate that the commonality analysis component may use any of a variety of well-known comparative analysis techniques to identify statistically significant *commonalities*.” EX1004, [0035]. Thus, Hartrell leaves these implementation details to a POSITA.

68. Such a POSITA would have found it obvious to utilize the teachings of Rockwood when implementing Hartrell’s commonality analysis component. As I described previously, Rockwood’s correlation engine 140 performs a similar function to Hartrell’s commonality analysis component (i.e., finding commonalities between detected events to identify attacks and attackers). A POSITA would have

found it obvious to employ various techniques of Rockwood's correlation engine 140, including, for example, providing a user of the computing device on which Hartrell's malware analysis system is implemented a GUI through which to specify proximity limits in order to define an n-dimensional space by which Hartrell's commonality analysis component would be able to identify commonalities between detected events. As described above, the proximity limits provided by the user through the console's GUI may include, for example, a time period around a target event for identifying related/common events. *See* EX1006, 9:4-16. This is consistent with Hartrell's creation of a "snapshot of the previous X seconds of the monitored activities that were recorded prior to receiving" notification of a suspected malware infection. *See* EX1004, [0027]. Hartrell does not describe or limit who defines the period of its snapshots, yet Rockwood would have taught a POSITA that it would have been obvious to allow a user of Hartrell's computing device a GUI through which to provide this definition.

69. A POSITA would have been motivated to utilize the teachings of Rockwood when implementing Hartrell's commonality analysis component, because Rockwood describes that its techniques "reduces the number of operations required to correlate detected events with stored events, resulting in faster correlations," and provides the controls that allow a user to "better . . . respond to attacks, preempt future attacks, and gather evidence against attackers." EX1006,

1:56-67. A POSITA would have recognized that a user of Hartrell's computing device would have been well positioned to identify suspicious activity on the computing device and the parameters around which a search ought to take place. Indeed, it was well known to allow a user, through a user interface, to indicate that they believe a malware infection had taken place and/or to alert a user of a possible infection and request their confirmation to take action. *See, e.g.,* EX1007, 52-61 ("Those skilled in the art and others will appreciate that programs designed to protect a computer from malware may initiate a scan for malware in many different circumstances. For example, existing programs in which the RootKit detector may be integrated, such as antivirus software, anti-spyware software, system utilities and the like, may be configured to perform a scan for malware 'on demand' *when a user or software system generates a command using a Graphical User Interface or similar input system.*").

F. Polyakov (EX1007)

70. Polyakov describes a RootKit detector designed to "identify[] a RootKit on a computer that is designed to conceal malware that is resident on the computer." EX1007, 3:32-34. Like the behavior monitor module described by Fan, Polyakov describes that "the RootKit detector leverage[s] services provided by kernel debugger facilities to automatically obtain data from specified data structures that are maintained by an operating system." EX1007, 3:35-38.

71. With respect to analyzing the data obtained by the kernel debugger facilities, Polyakov describes that the RootKit detector includes an integrity checker 212 that employs “three primary techniques” to identify a RootKit on a computer. EX1007, 5:17-23. One of the techniques described by Polyakov for analyzing the returned data is the use of “heuristics” to analyze “‘suspicious modifications’ that are made in data structures maintained by the operating system 204 to determine whether the modifications were generated by a RootKit.” EX1007, 5:42-45.

G. An Obvious Combination of Hartrell, Fan, Rockwell, and Polyakov

72. As I described above with respect to the combination of Hartrell and Fan, a POSITA would have found it obvious to implement Hartrell’s system activity monitor component 102 using “functional kernel mode drivers,” as described by Fan. A POSITA would have understood that Fan’s functional kernel mode drivers would have returned data similar to the kernel debugger facilities described by Polyakov—i.e., various “data structures that are maintained by an operating system.” *Compare* EX1005, 6:33-41, 6:56-60 to EX1007, 5:42-65. Both Fan and Polyakov utilize the data that can be obtained in “kernel mode” in order to identify malware that may otherwise be able to disguise itself to “user mode” detection services (e.g., rootkits). *Compare* EX1005, 4:45-53 to EX1007, 3:32-46.

73. As I noted above, Polyakov teaches “three primary techniques” to identify a RootKit on a computer based on the data returned by its kernel debugger

facilities. EX1007, 5:17-23. A POSITA would have found it obvious to implement Hartrell's malware analysis system 100 (e.g., as part of or in communication with the system activity monitor component 102, which uses "any of a variety of well-known monitoring techniques to monitor operating system and/or network activities") to use the "three primary techniques" to identify a RootKit taught by Polyakov to analyze the records of the monitored activities and help provide notification of whether there is a suspected malware infection. *See* EX1004, [0027]; EX1007, 5:17-6:12. Polyakov teaches that these techniques are employed by its integrity checker 212, and a POSITA would have found it obvious to integrate at least this functionality of the integrity checker 212 as a component or subcomponent into or in communication with Hartrell's malware analysis system 100. Indeed, Polyakov describes the integrity checker 212 as a "software component" executed by computer 200 (*see* EX1007, 3:47-58, 5:6-12), and a POSITA would have found it obvious that the functionality of integrity checker 212 could be integrated as a software component or subcomponent into or in communication with Hartrell's malware analysis system 100, which describes that "locally executing anti-malware software" informs malware analysis system 100 of "the detection of a malware infection or suspected infection" (*see* EX1004, [0014]).

74. A POSITA would have been motivated to make this combination, because Hartrell does not specifically describe or otherwise limit how, in step 206

of FIG. 2, the system activity monitor component determines or receives notification that there is a suspected malware infection. *See* EX1004, [0027]. Rather, Hartrell leaves these details to a POSITA. *Id.* Polyakov provides at least three well-known techniques that a POSITA could utilize in implementing Hartrell’s malware analysis system 100 to provide the notification that there is a suspected malware infection. EX1007, 5:17-23. Indeed, Polyakov specifically describes with respect to the second of these well-known techniques that “[t]hose skilled in the art and others will recognize that heuristic techniques that search for behaviors characteristic of malware *are particularly effective* in identifying previously unknown RootKit.” EX1007, 5:62-65. Thus, under Polyakov’s own explicit teachings, a POSITA would have recognized the effectiveness of employing the techniques it describes—and particularly the use of heuristics to identify malware—in detecting possible infections.

VIII. OVERVIEW OF UNPATENTABILITY GROUNDS

Ground	Claim(s)	Prior Art Basis
1A	1-10 and 13-16	§103: Hartrell, Fan, and Rockwood
1B	11 and 12	§103: Hartrell, Fan, Rockwood, and Polyakov

IX. GROUND 1: HARTRELL, FAN, AND ROCKWOOD RENDER OBVIOUS CLAIMS 1-10 AND 13-16

A. Claim 1

75. Claim 1 of the ’244 Patent recites:

1. A method for identifying an origin of suspected pestware activity on a computer, the method comprising:
monitoring, with a kernel-mode driver, activity on the computer;
generating an activity log on a file storage device of the computer from the kernel-mode driver;
receiving, from a user via an interface of the computer, a time of interest relating to a suspicion of pestware on the computer, wherein the time of interest includes a time interval;
issuing a timestamp after receiving the time of interest;
identifying, based upon the time of interest, indicia of pestware, wherein the identifying is initiated by the issuing the timestamp; and
accessing, using a hardware processor of the computer, at least a portion of a recorded history of externally networked sources that the computer received files from so as to identify, based at least in part upon the identified indicia of pestware, a reference to an identity of an externally networked source that is suspected of originating pestware;
wherein the recorded history of externally networked sources is stored on the file storage device.

76. As I explain in more detail below through my feature-by-feature analysis, the combination of the Hartrell, Fan, and Rockwood references disclose or render obvious claim 1.

1. [1pre] A method for identifying an origin of suspected pestware activity on a computer, the method comprising:

77. The '244 Patent describes that “[p]ersonal computers and business computers are continually attacked by trojans, spyware, and adware, collectively referred to as ‘malware,’ ‘spyware’ or ‘pestware.’” EX1001, 1:43-45. In other words, the '244 Patent uses the terms “malware” and “pestware” interchangeably.

78. Hartrell describes that “[a] difficulty with the information provided after an infection is detected by today’s anti-malware software is that system administrators are unable to determine the root cause or *origin* of that malicious code.” EX1004, [0003]. Hartrell describes a system that solves this problem. *See* EX1004, [0014]. Specifically, “[u]pon being informed of the detection of a *malware* infection or suspected infection, for example, by locally executing *anti-malware software*, the malware analysis system provides a snapshot (or evidence) of the activities that were conducted within a specified time frame before the detected or suspected infection for analysis.” EX1004, [0014]. “The snapshot of the activities can then be *analyzed to determine the ‘root cause’ or source of the infection* (i.e., the activities that led to the infection), including, for example, how the infected computer system was discovered, how the infected computer system was accessed, what operating system objects were manipulated, and the like.” *Id.* In one example scenario, Hartrell describes that “[t]he malware analysis system performs the commonality analysis and discovers that each of these ten computer systems visited the same web site within one minute prior to the infection.” *Id.*, [0037].

2. [1a] monitoring, with a kernel-mode driver, activity on the computer;

79. Hartrell describes a system activity monitor component 102 that “provides run-time *monitoring of the operating system resources* for changes to the file system, common file formats, configurations (registry), network activities, use

of common application program interfaces (APIs), or any other operating system object.” EX1004, [0019]. “The system activity monitor component may ***run on and monitor the activity of a computer system***, such as, by way of example, a local desktop operating system, a server or network device.” *Id.*

80. As I described above in my description of the references and their obvious combinations, a POSITA would have found it obvious to implement Hartrell’s system activity monitor component 102 using “functional kernel mode drivers,” as described by Fan. Specifically, Fan describes a “behavior monitor module 210” that “is real-time software including ***two functional kernel mode drivers***” and that “***monitors operating system operations***.” EX1005, 6:19-24. Fan describes that “[m]odule 210 monitors and intercepts operating system operations such as file and registry access, process creation and process termination, and certain system APIs such as CreateProcess() CreateRemoteThread() and VirtualAlloc().” EX1005, 6:33-36. The module 210 can also intercept network traffic and events. *Id.*, 6:37-41.

3. [1b] generating an activity log on a file storage device of the computer from the kernel-mode driver;

81. The ’244 Patent describes that “[t]he activity monitor 116 in several embodiments monitors activities on the computer and stores information about the activities in the activity log 121. EX1001, 5:21-23. The activity log 121 may “include information (e.g., historical logs) relating to events on the computer 100.”

EX1001, 3:34-40; 6:4-9. The '244 Patent also describes activity log 220 “may include information about processes running on the computer 100 and files corresponding to the processes,” and/or may include “a history of API calls and respective times made by processes running on the protected computer.” EX1001, 6:9-16.

82. Hartrell describes that “[t]he computing device on which the malware analysis system is implemented may include a central processing unit, memory, input devices (e.g., keyboard and pointing devices), output devices (e.g., display devices), and *storage devices* (e.g., disk drives).” EX1004, [0024]. “While executing, the system activity monitor component *records the monitored activities in a data store*, which may be in memory, on physical media, or other logical data store.” EX1004, [0019]. In this regard, the records stored by Hartrell’s system activity monitor are the activity logs described in the '244 Patent. *See* EX1001, 5:21-32.

83. A POSITA would have understood or at least found it obvious that the record of monitored activities stored by the computer executing Hartrell’s activity monitor is an activity log. For example, Hartrell describes that:

The system activity monitor component may be configured to record information regarding the monitored activity, such as, by way of example: the identified operating system object or network object involved in the monitored activity (e.g., *file name*, socket, IP address,

logical paths, etc.); the details of the change (e.g., create file, listen on network socket, etc.); if applicable, the details of the object prior to the change; the source(s) of the change (e.g., *process id*, user security context, logical storage identity from which data originated, logical storage identity where the change occurred, network sources such as uniform resource locator (URL) or internet protocol (IP) address, the API call used to make the change, etc.); a date and time stamp in which the event occurred.

EX1004, [0019].

84. Indeed, Fan describes that “*a log is kept* of all events received” by its behavior monitor module, which “two functional kernel mode drivers that monitors operating system operations.” EX1005, 8:35-37, 6:19-21. Thus, it would have at least been obvious to a POSITA that, in combination with Fan’s functional kernel mode drivers, Hartrell’s activity monitor would generate an activity log on a file storage device of the computer from the kernel-mode driver in the same manner as Fan describes the operation of its behavior monitor module.

4. [1c] receiving, from a user via an interface of the computer, a time of interest relating to a suspicion of pestware on the computer, wherein the time of interest includes a time interval;

85. The ’244 Patent describes the reception of a “time of interest” with respect to FIG. 5. In particular, “a time of interest is initially established based upon a suspicion that pestware has infected a computer.” EX1001, 8:59-61.

In some embodiments, for example, a user may establish the time of interest based upon events the user observed (e.g., pop-ups or a system crash). In one embodiment, as discussed with reference to FIG. 1, in response to a user request, the time stamp module 112 may issue a time stamp and initiate research that is related to activity occurring at or around the time of the time stamp. In variations, the user is provide [sic] with an alert in response to the heuristics module 110 identifying an activity that is suspicious and the user is given an option to initiate research at or around the time of interest.

EX1001, 8:61-9:4. In other words, the '244 Patent provides various examples by which a user may provide a "time of interest," some of which simply involve a user approving "research at or around the time of interest." *Id.* The '244 Patent also describes that the "time of interest may be established, for example, to include a time period before the time stamp is issued so that a search for suspicious activity is limited to a particular time period." EX1001, 9:11-14.

86. Hartrell describes that the "system activity monitor component can be notified of a malware infection." EX1004, [0019]. "Upon receiving the notification, the system activity monitor component creates and provides *a time-bounded snapshot of activities that occurred before* and/or after *the notification of infection* to the other components of the malware analysis system." *Id.* Hartrell teaches that "[t]he system activity monitor component can use *the report of the infection* as the base time (e.g., t0) from which to create the time-bounded snapshots of recorded

activities.” *Id.*, [0030]. “For example, the system activity monitor component may create a snapshot of the previous X seconds of the monitored activities that were recorded prior to receiving the notification event.” *Id.*, [0027].

87. In other words, Hartrell describes that the system activity monitor component receives a time of interest that includes a time interval (i.e., the period of the snapshot, which can be “X seconds of the monitored activities that were recorded prior to receiving the notification event”). This is the same type of “time of interest” described in the ’244 Patent. *See* EX1001, 8:59-9:14 (describing that the “time of interest” is related to an alert of suspicious activity and that the “time of interest may be established, for example, to include a time period before the time stamp is issued so that a search for suspicious activity is limited to a particular time period.”). Furthermore, Hartrell describes that at least one purpose of this snapshot is to identify “certain reoccurring activities” that may be “*categorized* as either potentially normal or *suspicious* with a malware state.” EX1004, [0021], [0035]. In other words, in Hartrell’s own words, the snapshot of period X seconds is related to the categorization of certain activities that are “*suspicious* with a malware state.” *Id.*

88. I have been informed that Patent Owner has argued that the term “suspicion of pestware on the computer” should be construed under this term’s plain and ordinary meaning. Under the interpretation advanced by the Patent Owner, a POSITA would have understood or at least found it obvious that Hartrell’s use of its

snapshot to categorize certain activities that are “*suspicious* with a malware state” means that the period of the snapshot (i.e., the claimed “time of interest”) is related “to a suspicion of pestware on the computer,” as recited in [1c]. In its claim construction briefing, Patent Owner contends that “terms like ‘suspicion,’ ‘suspicious,’ or ‘suspected’ referring to pestware” is a well-understood concept and requires no construction. EX1015, 14-15. For example, PO suggests a “suspected pestware object” (a term used in related U.S. Patent No. 8,719,932³) is simply the object identified ‘based upon the activity’”—referring to the monitored activity in the limitations above. *Id.* PO argues that the ’244 Patent uses “the ‘suspicious’ terms in similar ways. *Id.* PO points to examples in the specification that it contends support its interpretation. *Id.*, 16-17 (citing ’932 Patent, 4:58-66 and characterizing disclosure as “identifying a suspected pestware object by analyzing activity based on weighted factors associated with pestware activity and determining that one or more of the weighted factors exceeds a threshold”). PO contends that “[a] POSITA would understand that activities that exceed the threshold may be...‘suspected’ of pestware.” *Id.*, 17. This is effectively the same process utilized by Hartrell when

³ As PO explains in its briefing, U.S. Patent No. 8,719,932 and the ’244 Patent challenged in this IPR share “the same title and figures, and substantially the same specification.” EX1015, 10.

evaluating the “certain reoccurring activities” in the snapshot. EX1004, [0021], [0035].

89. With respect to the time-bounded snapshot created by Hartrell’s system activity monitor component, Hartrell generally describes receiving a notification of malware infection, but does not explicitly describe that this notification or the period of the snapshot created as a result of the notification are received “from a user via an interface of the computer.” *See* EX1004, [0019], [0027. However, as described in Sections IV.A.4-5, *supra*, a POSITA would have found it obvious in light of the teachings of Rockwood that the period of the snapshot and initiation of a scan would be received “from a user via an interface of the computer,” as recited in [1c].

90. As I described in my overview of Rockwood, Rockwood describes a “system and method for intruder tracking using advanced correlation in a network security system.” EX1006, 1:8-11. Rockwood describes a correlation engine 140 that “is operable to correlate detected events 110 to detect an attack” EX1006, 4:49-51. Rockwood also describes a console 50 that includes a graphical user interface 52 (GUI) (*see* EX1006, 5:39-59) through which “a user, such as an operator (not shown), may view and process the event information of detected events 110” (*see* EX1006, 5:27-34) and provide “proximity limits” used by the correlation engine 140 to correlate various detected events to identify an attack and attacker (*see* EX1006, 8:6-34). Proximity limits “define a portion of n-dimensional space

surrounding” detected events and are used to correlate those events. *See* EX1006, 8:6-34. The proximity limits provided by the user through the console’s GUI may include, for example, a time period around a target event. *See* EX1006, 9:4-16.

91. Based on these teachings, a POSITA would have found it obvious to, for example, provide a user of the computing device on which Hartrell’s malware analysis system is implemented a GUI through which to specify proximity limits in order to define an n-dimensional space by which Hartrell’s commonality analysis component would be able to identify commonalities between detected events. As described above, the proximity limits provided by the user through the console’s GUI may include, for example, a time period around a target event for identifying related/common events. *See* EX1006, 9:4-16. This is consistent with Hartrell’s creation of a “snapshot of the previous X seconds of the monitored activities that were recorded prior to receiving” notification of a suspected malware infection. *See* EX1004, [0027]. Hartrell does not describe or limit who defines the period of its snapshots, yet Rockwood would have taught a POSITA that it would have been obvious to allow a user of Hartrell’s computing device a GUI through which to provide this definition. Accordingly, it would have been obvious to a POSITA that the period of the snapshot (i.e., the “time of interest”) received by Hartrell’s system activity monitor component is received “from a user via an interface of the computer.”

5. [1d] issuing a timestamp after receiving the time of interest;

92. As I described with respect to [1c], Hartrell describes that the “system activity monitor component can be notified of a malware infection.” EX1004, [0019]. “Upon receiving the notification, the system activity monitor component creates and provides a time-bounded snapshot of activities that occurred before and/or after the notification of infection to the other components of the malware analysis system.” *Id.* Hartrell teaches that “[t]he system activity monitor component can use the report of the infection ***as the base time (e.g., t0) from which to create the time-bounded snapshots of recorded activities.***” *Id.*, [0030]. In other words, the “base time (e.g., t0)” is a timestamp that is issued after it receives a notification of a potential malware infection.

93. Hartrell describes a categorization component 104 that “normalizes the snapshot data for use by the commonality analysis component.” EX1004, [0019]-[0020]. FIG. 6 illustrates an example of the resulting normalized data, which includes a “time field [that] specifies, ***relative to ‘t0’ (i.e., the time at which the notification of the malware infection is received by the malware analysis system),*** the time the activity identified in the corresponding recorded activity field was recorded by the system activity monitor component of the malware analysis system.” EX1004, [0032]. A POSITA would have recognized or at least found it obvious that to display monitored and stored activities “relative to ‘t0’”, the system would need

to issue and store t_0 at the time it receives the notification of the malware infection. Thus, it would have at least been obvious to a POSITA that Hartrell's system activity monitor component issues a timestamp (e.g., "base time (e.g., t_0)" and/or the "time field" included with each of the normalized data) after receiving the time of interest.

6. [1e] identifying, based upon the time of interest, indicia of pestware, wherein the identifying is initiated by the issuing the timestamp;

94. As I described with respect to [1d], Hartrell describes a categorization component 104 that "normalizes the snapshot data for use by the commonality analysis component." EX1004, [0019]-[0020]. "In some embodiments, the categorization component may map the normalized activities along a malware state model, *with the initial infection notification as the reference* for pre- and post-infection activities." EX1004, [0020]. In other words, the creation of the snapshot and the subsequent processing of the snapshot by the categorization component is initiated by the received notification of a suspected malware infection and issuance of the base time t_0 . See EX1004, [0027], [0029]-[0030].

95. Hartrell describes that the "the categorization component maps each normalized activity to a malware state as provided by a malware state model." EX1004, [0029]. The "malware states comprise a propagation state 402, an activation state 404, a carrier state 406, an execution state 408, and a persistence state 410." EX1004, [0030]. "The categorization component can use the table of

mappings of normalized actions to intelligently map each normalized activity to a malware state.” EX1004, [0031]. “FIG. 5 illustrates a table 500 of example mappings of normalized actions to pre- and post-notification malware states.” *Id.* Hartrell teaches that, “if the normalized action is contained in the pre-infection snapshot, the normalized action is assigned the malware state that is specified in the corresponding pre-notification malware state field,” and that, “if the normalized action is contained in the post-infection snapshot, the normalized action is assigned the malware state that is specified in the corresponding post-notification malware state field.” *Id.* In other words, the categorization component identifies, based upon the time of interest (i.e., before or after the notification of infection), indicia of pestware (i.e., assigned malware state).

96. Additionally, Hartrell describes that the commonality analysis component of the malware analysis system “compares the normalized activities that are recorded for a particular malware infection to previously stored normalized activities for the same malware infection,” and “tags similar activities as being suspicious.” EX1004, [0035]. Thus, the commonality analysis component also identifies indicia of pestware (i.e., similar activities as being suspicious). Because the normalized activities compared by the commonality analysis component come from snapshots (i.e., a list of activities from a period of time before a notification of

infection), the comparison performed by the commonality analysis component is “based upon the time of interest.” *See* EX1004, [0018], [0031], [0035].

7. ***[1f] accessing, using a hardware processor of the computer, at least a portion of a recorded history of externally networked sources that the computer received files from***

97. Hartrell describes that “the system activity monitor component may execute as a runtime process that may use any of a variety of well-known monitoring techniques to monitor operating system and/or ***network activities***.” EX1004, [0027]. Specifically, the “system activity monitor component may be configured to record information regarding the monitored activity, such as, by way of example: . . . the details of the change (e.g., ***create file***, listen on network Socket, etc.); [and] . . . the ***source(s) of [a] change*** (e.g., . . . ***network sources*** such as uniform resource locator (URL) or internet protocol (IP) address, the API call used to make the change, etc.)” EX1004, [0019].

98. Once Hartrell’s system receives a notification of malware infection, a snapshot is created and “the categorization component normalizes the activities contained in the snapshots provided by the system activity monitor component into predetermined set of common categories of system or ***network events*** (also referred to as ‘normalized actions’).” EX1004, [0029]. For example, one of the examples of normalized actions illustrated in FIG. 6 is “File Transfer.” EX1004, FIG. 6. In other

words, the categorization component accesses a portion of a recorded history of externally networked sources that the computer received files from.

99. Hartrell gives an example of the malware analysis system “monitoring all activities that modify local data stores (e.g., file system, registry, databases, etc.) and network activities (e.g., TCP/IP connections, web site visits, etc.).” EX1004, [0037]. In this example, the “malware analysis system performs the commonality analysis and discovers that each of [] ten computer systems visited the same web site within one minute prior to the infection.” *Id.* A POSITA would have recognized or at least found it obvious that for a computer to “visit” a web site, it would request and receive data from the web site (e.g., an HTML file).

100. Hartrell describes that “the system activity monitor component may execute on all of the computer systems that are to be monitored, and one or more of the categorization component, the commonality analysis component, and the response recommendation generator *may execute on one of the aforementioned computer systems.*” EX1004, [0023]. In other words, Hartrell describes that a single computer may execute all four components (i.e., a system activity monitor component 102, a categorization component 104, a commonality analysis component 106, and a response recommendation generator 108). *Id.* Furthermore, “[t]he computing device on which the malware analysis system is implemented *may include a central processing unit*, memory, input devices (e.g., keyboard and

pointing devices), output devices (e.g., display devices), and storage devices (e.g., disk drives).” EX1004, [0024]. Thus, the categorization component 104 and a commonality analysis component 106 accessing the network activity contained in the snapshot created by the system activity monitor component would have been accessed “using a hardware processor of the computer,” as recited in [1f].

8. [1g] so as to identify, based at least in part upon the identified indicia of pestware, a reference to an identity of an externally networked source that is suspected of originating pestware

101. As I described with respect to [1f], Hartrell describes that the “system activity monitor component may be configured to record information regarding the monitored activity, such as, by way of example:. . . the details of the change (e.g., *create file*, listen on network Socket, etc.); [and] . . . the *source(s) of [a] change* (e.g., . . . *network sources* such as uniform resource locator (URL) or internet protocol (IP) address, the API call used to make the change, etc.)” EX1004, [0019]. Hartrell describes that the “snapshot of the activities can then be *analyzed to determine the ‘root cause’ or source of the infection* (i.e., the activities that led to the infection), including, for example, how infected computer system was discovered, how the infected computer system was accessed, what operating system objects were manipulated, and the like.” EX1004, [0014].

102. Hartrell gives an example of the malware analysis system “monitoring all activities that modify local data stores (e.g., file system, registry, databases, etc.)

and network activities (e.g., TCP/IP connections, web site visits, etc.).” EX1004, [0037]. In this example, the “malware analysis system performs the commonality analysis and *discovers that each of [] ten computer systems visited the same web site within one minute prior to the infection.*” *Id.* A POSITA would have recognized or at least found it obvious that, in this example, the commonality analysis component is identifying, based at least in part upon the identified indicia of pestware, a reference to an identity of an externally networked source that is suspected of originating pestware.

9. [1h] wherein the recorded history of externally networked sources is stored on the file storage device

103. As I explained with respect to [1f], Hartrell describes that a single computer may execute all four components (i.e., a system activity monitor component 102, a categorization component 104, a commonality analysis component 106, and a response recommendation generator 108) of the malware analysis system. *Id.* Furthermore, Hartrell describes that “[t]he computing device on which the malware analysis system is implemented *may include* a central processing unit, memory, input devices (e.g., keyboard and pointing devices), output devices (e.g., display devices), and *storage devices (e.g., disk drives).*” EX1004, [0024]. Thus, when Hartrell describes that “the system activity monitor component *records the monitored activities in a data store*, which may be in memory, on physical media, or other logical data store,” a POSITA would have understood or at

least found it obvious that the monitored activities recorded by the system activity monitor component are store on the file storage device of the computer executing the malware analysis system. EX1004, [0019].

104. Hartrell describes that “the system activity monitor component may execute as a runtime process that may use any of a variety of well-known monitoring techniques to monitor operating system and/or *network activities*.” EX1004, [0027]. Thus, the monitored activities recorded by the system activity monitor component that are stored on the file storage device of the computer executing the malware analysis system include the *network activities* that make up the recorded history of externally networked sources, which were described with respect to [1f].

B. Claim 2

105. Claim 2 of the '244 Patent recites:

<p>2. The method of claim 1, wherein the identifying includes: accessing at least one log of historical events on the computer; locating at least one event in the at least one log that corresponds to the time of interest; and determining that the at least one event is associated with the indicia of pestware.</p>
--

**1. [2a] *The method of claim 1, wherein the identifying includes:
accessing at least one log of historical events on the computer***

106. Hartrell describes that a system activity monitor component 102 that “provides run-time *monitoring of the operating system resources* for changes to the

file system, common file formats, configurations (registry), network activities, use of common application program interfaces (APIs), or any other operating system object.” EX1004, [0019]. As described with respect to [1h], Hartrell describes that a single computer may execute all four components (i.e., a system activity monitor component 102, a categorization component 104, a commonality analysis component 106, and a response recommendation generator 108) of the malware analysis system. *Id.* Furthermore, Hartrell describes that “[t]he computing device on which the malware analysis system is implemented **may include** a central processing unit, memory, input devices (e.g., keyboard and pointing devices), output devices (e.g., display devices), and **storage devices (e.g., disk drives).**” EX1004, [0024]. Thus, when Hartrell describes that “the system activity monitor component **records the monitored activities in a data store**, which may be in memory, on physical media, or other logical data store,” a POSITA would have understood or at least found it obvious that the monitored activities recorded by the system activity monitor component are stored on the file storage device of the computer executing the malware analysis system. EX1004, [0019]. As described above with respect to [1b], a POSITA would have at least found it obvious that the records stored by Hartrell’s systems activity monitor are the activity logs described in the ’244 Patent.

107. Accordingly, when the system activity monitor component 102 received the notification of malware infection and generates the snapshot, the system

activity monitor component is accessing the records of the monitored activities stored in the file storage device of the computer executing the malware analysis system, and passing the generated snapshot to categorization component 104. *See* EX1004, [0019]-[0020], [0023]-[0024], [0027]-[0029].

108. As I described with respect to [1e], the categorization component 104 “normalizes the snapshot data for use by the commonality analysis component,” which includes identifying, based upon the time of interest (i.e., before or after the notification of infection), indicia of pestware (i.e., assigned malware state). EX1004, [0019]-[0020]. Because the categorization component 104 makes this identification based on the snapshot data received from the system activity monitor component, the identifying includes accessing at least one log of historical events on the computer (i.e., the snapshot).

2. [2b] locating at least one event in the at least one log that corresponds to the time of interest; and

109. As I described with respect to [1e], Hartrell describes that “the categorization component *maps each normalized activity to a malware state* as provided by a malware state model.” EX1004, [0029]. In other words, the categorization component locates and maps each of the events in the snapshot. And the entire snapshot corresponds to the time of interest. *See* EX1004, [0014] (“the malware analysis system provides a snapshot (or evidence) of the activities that were

conducted within a specified time frame before the detected or suspected infection for analysis.”).

3. [2c] determining that the at least one event is associated with the indicia of pestware.

110. As I described with respect to [1e], Hartrell describes that the “the categorization component maps each normalized activity to a malware state as provided by a malware state model.” EX1004, [0029]. The “malware states comprise a propagation state 402, an activation state 404, a carrier state 406, an execution state 408, and a persistence state 410.” EX1004, [0030]. “The categorization component can use the table of mappings of normalized actions to intelligently map each normalized activity to a malware state.” EX1004, [0031]. In other words, the categorization component identifies indicia of pestware (i.e., assigned malware state).

111. Additionally, Hartrell describes that the commonality analysis component of the malware analysis system “compares the normalized activities that are recorded for a particular malware infection to previously stored normalized activities for the same malware infection,” and “tags similar activities as being suspicious.” EX1004, [0035]. Thus, the commonality analysis component also identifies indicia of pestware (i.e., similar activities as being suspicious).

C. Claim 3

112. Claim 3 of the '244 Patent recites:

3. The method of claim 2, wherein the determining that the at least one event is associated with the indicia of pestware includes following references in the at least one log to a suspected pestware object.

113. The '244 Patent does not explicitly use the phrase “following references,” but based on the context in which the term “references” is used, a POSITA would have understood “following references” to encompass any descriptive information in logs that can be used by the research module 108 to link disparate events captured by the logs. For example, the '244 Patent describes that the historical logs illustrated in FIG. 2 “are logs of historical events that occurred on the computer, and each of the logs includes information that *may be used to reference other information* when, for example, activities, processes and files are traced so as to determine an origin of the pestware or suspected pestware.” EX1001, 6:1-9. In another example, the '244 Patent describes that events such as the crashing or spontaneous restart of certain applications (e.g., explorer.exe) will show up in the debugging logs, and will be listed as “time stamped events, that also *reference* files on the system.” EX1001, 6:59-65. From these teachings, a POSITA would have understood claim 3 to relate to following the descriptive information in logs (i.e., “references”) that can be used by the research module 108 to link disparate events captured by the logs and ultimately to the indicia of pestware.

114. Hartrell describes that the “the commonality analysis component compares the normalized activities that are recorded for a particular malware infection to previously stored normalized activities for the same malware infection.” EX1004, [0035]. As part of this process, “the commonality analysis component tags similar activities as being suspicious,” and “assigns a malware state to the suspicious activity.” *Id.* Hartrell describes an example scenario implementing this process in which:

The malware analysis system [] normalizes the snapshot data . . . , and conducts commonality analysis on the ten provided snapshots to find any recurring activities. The malware analysis system performs the commonality analysis and ***discovers that each of these ten computer systems visited the same web site*** within one minute prior to the infection.

EX1004, [0037].

115. A POSITA would have recognized or at least found it obvious that the commonality analysis would have included “following references in the at least one log to a suspected pestware object,” because the commonality analysis would have considered the various progressive states assigned by the categorization component to find the state in which the source of the malware infection. This approach is consistent with Hartrell’s description of the malware state model (*see* EX1004, [0030]), and Hartrell’s objective of analyzing the snapshot “to determine the ‘root

cause’ or *source of the infection (i.e., the activities that led to the infection)*” (*id.*, [0014]).

D. Claim 4

116. Claim 4 of the ’244 Patent recites:

<p>4. The method of claim 3, wherein following the references includes following at least one reference in the activity log from the event to a suspected pestware process.</p>
--

117. As described with respect to [3], Hartrell describes analyzing the snapshot “to determine the ‘root cause’ or *source of the infection (i.e., the activities that led to the infection)*,” which a POSITA would have recognized or at least found obvious included “following references in the at least one log to a suspected pestware object.” Hartrell further describes that:

The system activity monitor component may be configured to record information regarding the monitored activity, such as, by way of example: *the identified operating system object* or network object involved in the monitored activity (e.g., *file name*, socket, IP address, logical paths, etc.); the details of the change (e.g., create file, listen on network socket, etc.); if applicable, the details of the object prior to the change; *the source(s) of the change* (e.g., *process id*, user security context, logical storage identity from which data originated, logical storage identity where the change occurred, *network sources such as uniform resource locator (URL)* or internet protocol (IP) address, the

API call used to make the change, etc.); a date and time stamp in which the event occurred.

EX1004, [0019].

118. Accordingly, Hartrell describes that the source of the change can be a process, and when it does, the snapshot will include a process_id for that process. *Id.* In these instances, when Hartrell describes analyzing the snapshot “to determine the ‘root cause’ or *source of the infection (i.e., the activities that led to the infection)*,” Hartrell is describing that following the references includes following at least one reference in the activity log from the event to a suspected pestware process.

E. Claim 5

119. Claim 5 of the '244 Patent recites:

<p>5. The method of claim 2, wherein the at least one log includes at least one log selected from the group consisting of an activity log, a browser history, browser cache, browser settings, operating system settings, an event log, a debugging log, a firewall log, file information and monitoring software logs.</p>
--

120. As I described with respect to [1b], a POSITA would have understood or at least found obvious that the record of monitored activities stored by the computer executing Hartrell’s activity monitor is an activity log. For example, Hartrell describes that:

The system activity monitor component may be configured to record information regarding the monitored activity, such as, by way of example: the identified operating system object or network object involved in the monitored activity (e.g., ***file name***, socket, IP address, logical paths, etc.); the details of the change (e.g., create file, listen on network socket, etc.); if applicable, the details of the object prior to the change; the source(s) of the change (e.g., ***process id***, user security context, logical storage identity from which data originated, logical storage identity where the change occurred, ***network sources such as uniform resource locator (URL)*** or internet protocol (IP) address, the API call used to make the change, etc.); a date and time stamp in which the event occurred.

EX1004, [0019].

121. This is a record of activities performed on the computer, and is therefore a “log,” as a POSITA would have understood that term. Indeed, the list of activities monitored and recorded by Hartrell’s activity monitor are consistent with the logs described in the ’244 Patent. EX1001, 6:9-13 (“an activity log 220, which may be generated by an activity monitor . . . and may include information about ***processes*** running on the computer 100 and ***files*** corresponding to the processes”).

122. Because the stored record of activities monitored by Hartrell’s activity monitor is an activity log, the snapshot created by Hartrell’s activity monitor, which is just a subset of the total activity log (e.g., “the previous X seconds of the monitored

activities that were recorded prior to receiving the notification event”), is also an activity log.

F. Claim 6

123. Claim 6 of the '244 Patent recites:

<p>6. The method of claim 1, wherein the user input is in response to an alert provided to the user regarding a detected suspicious activity on the computer.</p>

124. As described with respect to [1c], Hartrell does not explicitly describe that the period of the snapshot is received “from a user via an interface of the computer.” Rather, Hartrell generally describes receiving a notification of malware infection, and then provides non-limiting examples from where it may be received. *See* EX1004, [0019], [0027]. For the reasons explained with respect to [1c], a POSITA would have found it obvious in light of the teachings of Rockwood that the period of the snapshot and initiation of a scan would be received “from a user via an interface of the computer,” as recited in [1c].

125. Furthermore, Fan describes that “*a user* or administrator of the computer system in question makes a determination or realizes that malware has infected the computer system or is present on the computer system, and desires to take action to remove the malware.” EX1005, 11:32-37. Indeed, it was well known to either allow a user, through a user interface, to indicate that they believe a malware infection had taken place or to alert a user of a possible infection and request their

confirmation to take action. *See, e.g.,* EX1006, 52-61 (“Those skilled in the art and others will appreciate that programs designed to protect a computer from malware may initiate a scan for malware in many different circumstances. For example, existing programs in which the RootKit detector may be integrated, such as antivirus software, anti-spyware software, system utilities and the like, may be configured to perform a scan for malware ‘on demand’ ***when a user or software system generates a command using a Graphical User Interface or similar input system.***”). From these teachings and their knowledge of typical antivirus and anti-pestware software available before the Critical Date, a POSITA would have known that typical antivirus and anti-pestware software provided alerts of suspected infections, and that a user seeing such an alert would be able to access and interact with a GUI through which the user could schedule a full scan of the computer. Accordingly, it would have been obvious to a POSITA that the user input is in response to an alert provided to the user regarding a detected suspicious activity on the computer, as recited in claim 6.

G. Claim 7

126. Claim 7 of the '244 Patent recites:

<p>7. The method of claim 1, including: reporting the identity of the externally networked source to an externally networked pestware research entity so as to enable the pestware research entity to research whether the externally networked source is a source of pestware.</p>

127. Hartrell describes that, “[i]n some embodiments, the malware analysis system can provide the snapshot of the activities to a *remote* (e.g., centralized) data store.” EX1004, [0014]. “For example, the system activity monitor component may provide the snapshots to a *remote data store* that is accessible by the other components of the malware analysis system and/or an administrator.” EX1004, [0027]. And while Hartrell provides several examples of where various components and their analysis may take place, Hartrell clarifies that “[o]ne skilled in the art will appreciate that the malware analysis system may be deployed in other ways.” EX1004, [0023].

128. Based on these teachings, a POSITA would have found it obvious to “report the identity of the externally networked source to an externally networked pestware research entity so as to enable the pestware research entity to research whether the externally networked source is a source of pestware,” as this would simply substitution of the user’s computer with an externally networked pestware research entity to achieve the same goal sought by Hartrell: “determine the ‘root cause’ or source of the infection (i.e., the activities that led to the infection).” EX1004, [0014].

129. Indeed, it was well known to the POSITA to configured malware detection systems like the one described by Hartrell to share its analysis with third parties that a POSITA would have at least found obvious are externally networked

pestware research entities in order to allow for the aggregation of threat data and broader dissemination of remediation solutions. *See, e.g.,* EX1007, 9:24-35 (“RootKit sample is reported to a ***remote computer associated with a trusted entity*** at block 318. . . . In this instance, data obtained that describes the state of the computer infected with RootKit is reported to a trusted entity so that developers may perform an analysis of the data.”); EX1008, 9:55-62 (“In an alternative embodiment of the present invention, the aggregation routine 310 maintains an Application Program Interface (API) that allows ***third-party providers*** to report suspicious events. In this instance, ***anti-malware services or other event detection system created by third parties may ‘plug-in’*** to the aggregation routine 310 and report Suspicious events by issuing one or more API calls.”); EX1009, 3:49-54 (“the virus database may be accessible via a network interface and maintained by a third party, such as, for example, an independent software vendor (ISV), independent BIOS vendor (IBV), an original equipment manufacturer (OEM), or an independent hardware vendor (IHV); however, these are merely a few non-limiting examples”).

H. Claim 8

130. Claim 8 of the '244 Patent recites:

<p>8. The method of claim 1, wherein the externally networked source is identified by an identifier selected from the group consisting of an I.P. address and a URL.</p>

131. As described with respect to [1f], Hartrell describes that “the system activity monitor component may execute as a runtime process that may use any of a variety of well-known monitoring techniques to monitor operating system and/or **network activities**.” EX1004, [0027]. Specifically, the “system activity monitor component may be configured to record information regarding the monitored activity, such as, by way of example: . . . the details of the change (e.g., **create file**, listen on network Socket, etc.); [and] . . . the **source(s) of [a] change** (e.g., . . . **network sources** such as **uniform resource locator (URL) or internet protocol (IP) address**, the API call used to make the change, etc.) . . .” EX1004, [0019].

132. Hartrell gives an example of the malware analysis system “monitoring all activities that modify local data stores (e.g., file system, registry, databases, etc.) and network activities (e.g., TCP/IP connections, web site visits, etc.).” EX1004, [0037]. In this example, the “malware analysis system performs the commonality analysis and discovers that each of [] ten computer systems **visited the same web site** within one minute prior to the infection.” *Id.* In light of Hartrell’s description of the contents of the snapshot, a POSITA would have recognized or at least found it obvious that the identified web site is an externally networked sources that would be identified by uniform resource locator (URL) or internet protocol (IP) address. *See* EX1004, [0019].

I. Claim 9

133. Claim 9 of the '244 Patent recites:

9. A system for identifying a source of suspected pestware on a computer, the system comprising:
a hardware processor;
a file storage device coupled to the hardware processor;
a timestamp module configured to be executed by the hardware processor, to generate a time of interest including a time interval in response to detected activity on the computer that is indicative of pestware, to issue a timestamp after generating the time of interest, and to initiate, in response to the issuance of the timestamp, identification of indicia of pestware;
a research portion configured to be executed by the hardware processor and to access at least one log on a file storage device of the computer to relate the time of interest to at least one externally networked source of potential pestware activity on the computer;
a kernel-mode driver configured to monitor activity on the computer and to generate an activity log that is included within the at least one log; and
a reporting portion configured to be executed by the hardware processor and to generate a report that identifies the at least one externally networked source of potential pestware activity.

134. As I explain in more detail below through my feature-by-feature analysis, the combination of the Hartrell, Fan, and Rockwood references disclose or render obvious claim 1.

1. [9pre] A system for identifying a source of suspected pestware on a computer, the system comprising:

135. This limitation is substantially the same as [1pre] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

2. [9a] a hardware processor

136. Hartrell describes that a single computer may execute all four components (i.e., a system activity monitor component 102, a categorization component 104, a commonality analysis component 106, and a response recommendation generator 108) of the malware analysis system. EX1004, [0023]. Furthermore, Hartrell describes that “[t]he computing device on which the malware analysis system is implemented ***may include a central processing unit***, memory, input devices (e.g., keyboard and pointing devices), output devices (e.g., display devices), and storage devices (e.g., disk drives).” EX1004, [0024].

3. [9b] a file storage device coupled to the hardware processor;

137. Hartrell describes that a single computer may execute all four components (i.e., a system activity monitor component 102, a categorization component 104, a commonality analysis component 106, and a response recommendation generator 108) of the malware analysis system. EX1004, [0023]; Furthermore, Hartrell describes that “[t]he computing device on which the malware analysis system is implemented ***may include*** a central processing unit, memory, input devices (e.g., keyboard and pointing devices), output devices (e.g., display devices), and ***storage devices (e.g., disk drives)***.” EX1004, [0024].

- 4. [9c.i] a timestamp module configured to be executed by the hardware processor, to generate a time of interest including a time interval in response to detected activity on the computer that is indicative of pestware,**

138. This limitation is substantially the same as [1c] and [1d] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

- 5. [9c.ii] issuing a timestamp after receiving the time of interest;**

139. This limitation is substantially the same as [1d] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

- 6. [9c.iii] to initiate, in response to the issuance of the timestamp, identification of indicia of pestware;**

140. This limitation is substantially the same as [1e] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

- 7. [9d] a research portion configured to be executed by the hardware processor and to access at least one log on a file storage device of the computer to relate the time of interest to at least one externally networked source of potential pestware activity on the computer;**

141. This limitation is substantially the same as [1f] and [1g] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

- 8. [9e] a kernel-mode driver configured to monitor activity on the computer and to generate an activity log that is included within the at least one log; and**

142. This limitation is substantially the same as [1a] and [1b] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

9. [9f] a reporting portion configured to be executed by the hardware processor and to generate a report that identifies the at least one externally networked source of potential pestware activity.

143. Hartrell describes that the malware analysis system 100 includes a response recommendation generator 108 that “processes the information provided by the commonality analysis component and generates a recommendation for responding to the malware infection.” EX1004, [0019], [0022]. Specifically, “the response recommendation generator identifies a suspicious normalized activity,” such as an “activity that was tagged as being suspicious by the commonality analysis component.” *Id.*, [0036]. Next, “the response recommendation generator generates a recommendation or multiple recommendations for responding to the suspicious normalized activity.” *Id.* The “response recommendation generator can provide the recommended response to an administrator.” *Id.*

144. In an example scenario described by Hartrell, the “malware analysis system performs the commonality analysis and discovers that each of these ten computer systems visited the same web site.” *Id.*, [0037]. As a result, the response recommendation generator “may alert the system operator of all ten infected computer systems to the pattern, and provide a recommendation to block this web site at their organization’s egress points to prevent future infections.” *Id.* A POSITA would have recognized or found obvious that these response recommendations are

reports that identify the at least one externally networked source of potential pestware activity, as recited in [9f].

J. Claim 10

145. Claim 10 of the '244 Patent recites:

10. The system of claim 9, wherein the at least one log includes information that relates the time of interest to an event, and wherein the at least one log includes information that relates the event to a suspected pestware object, and wherein the at least one log includes information that relates the suspected pestware object to the at least one externally networked source of potential pestware activity.

146. Claim 10 is substantially the same as [2a]-[2c] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

K. Claim 13

147. Claim 13 of the '244 Patent recites:

13. The system of claim 9, wherein the at least one log includes at least one log selected from the group consisting of an activity log, a browser history, browser cache, browser settings, operating system settings, an event log, a debugging log, a firewall log, file information and monitoring software logs.

148. Claim 13 is substantially the same as [5] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

L. Claim 14

149. Claim 14 of the '244 Patent recites:

14. The system of claim 9, wherein the timestamp module is further configured to generate the time of interest as a point in time.

150. As I described with respect to [1e], Hartrell teaches that “the system activity monitor component determines whether it received notification of a suspected malware infection.” EX1004, [0027]. “The system activity monitor component can use the report of the infection *as the base time (e.g., t0) from which to create the time-bounded snapshots of recorded activities.*” *Id.*, [0030]. In other words, the “base time (e.g., t0)” is a timestamp, and a POSITA would have recognized that the time of interest and timestamp are both a “point in time,” because they are the time at which the system activity monitor component received notification of a suspected malware infection.

M. Claim 15

151. Claim 15 of the '244 Patent recites:

15. The system of claim 9, wherein the at least one externally networked source of potential pestware activity is identified by an identifier selected from the group consisting of an I.P. address and a URL.

152. Claim 15 is substantially the same as [8] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

N. Claim 16

153. Claim 16 of the '244 Patent recites:

16. The system of claim 9, wherein the at least one log includes at least one log selected from the group consisting of an activity log, a browser history, browser cache, browser settings, operating system settings, an event log, a debugging log, a firewall log, file information and monitoring software logs.

154. Claim 16 is substantially the same as [5] and is obvious over the Hartrell-Fan-Rockwood combination for at least similar reasons.

X. GROUND 1B: HARTRELL, FAN, ROCKWOOD, AND POLYAKOV RENDER OBVIOUS CLAIMS 11 AND 12

A. Claim 11

155. Claim 11 of the '244 Patent recites:

11. The system of claim 10, including:
a heuristics module configured to be executed by the hardware processor and to evaluate whether the event is a suspected pestware-related event before the research portion accesses information in the at least one log that relates the event to the suspected pestware object.

156. As I described in my overview of Polyakov and its combination with Hartrell, Fan, and Rockwood, a POSITA would have found it obvious to implement Hartrell's malware analysis system 100 to use the "three primary techniques" to identify a RootKit taught by Polyakov to analyze the records of the monitored activities and help provide notification of whether there is a suspected malware infection. *See* EX1004, [0027]; EX1007, 5:17-6:12. As further described in Sections IV.B.1-2, *supra*, a POSITA would have found it obvious that the

functionality of Polyakov's integrity checker 212 could be integrated as a software component or subcomponent into or in communication with Hartrell's malware analysis system 100, which describes that "locally executing anti-malware software" informs malware analysis system 100 of "the detection of a malware infection or suspected infection" (*see* EX1004, [0014]). A POSITA would have understood or at least found it obvious that, in the proposed combination, a software component employing the "heuristic techniques" described by Polyakov would be a "heuristics module configured to be executed by the hardware processor," as recited in claim 11. Polyakov describes that the software components of its system are executed by computer 200 (*see* EX1007, 3:46-58), and that the computer 200 includes a Central Processing Unit (CPU) that executes the processes on computer 200 (*see* EX1007, 7:30-37), which a POSITA would have understood to include the software components of its RootKit detector.

157. Further, Hartrell describes that the "notification of a suspected malware infection" is received by the system activity monitor component in step 206 of FIG. 2, which is before the system activity monitor component accesses the records of monitored activities to create the pre-infection and post-infection snapshots in steps 208 and 210 of FIG. 2. *See* EX1004, [0027]. Thus, a POSITA would have understood or at least found it obvious that the integrated software components/sub-components implementing Polyakov's "heuristic techniques" would have "evaluate

whether the event is a suspected pestware-related event before the research portion accesses information in the at least one log that relates the event to the suspected pestware object,” as recited in claim 11.

B. Claim 12

158. Claim 12 of the '244 Patent recites:

<p>12. The system of claim 11, wherein the heuristics module determines that the at least one event is suspected to be pestware-related by analyzing other events in connection with the event, wherein the other events are also related to the time of interest.</p>

159. With respect to its “heuristic techniques,” Polyakov describes that “heuristics are employed to analyze ‘suspicious modifications’ that are made in data structures maintained by the operating system 204 to determine whether the modifications were generated by a RootKit.” EX1007, 5:42-45. Polyakov goes on to describe that “[t]hose skilled in the art and others will recognize that heuristic techniques that search for behaviors characteristic of malware are particularly effective in identifying previously unknown RootKit.” EX1007, 5:62-65. Thus, Polyakov’s “heuristic techniques” are meant to identify and analyze a number of different “behaviors,” not any one behavior. A POSITA would have understood or at least found it obvious that these combinations of multiple behaviors would help distinguish between a normal program that may have “legitimate reasons” for performing any one behavior and malicious programs that use a set of multiple

known behaviors to conceal malware. *See* EX1007, 5:42-65. Indeed, this is consistent with Hartrell's disclosure that the system activity monitor component monitors multiple "predetermined activities on or about the computer system." *See* EX1004, [0027].

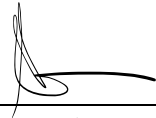
160. Based on these teachings a POSITA would have at least found it obvious that the software component/sub-component in the proposed combination that utilizes Polyakov's heuristic techniques (i.e., the claimed heuristics module) would "determine[] that the at least one event is suspected to be pestware-related by analyzing other events in connection with the event, wherein the other events are also related to the time of interest." A POSITA would have understood or at least found it obvious from the teachings of Polyakov that known heuristic techniques often identified suspected malware based on a series of successive or parallel behaviors that together differentiate the behavior of malware (e.g., a rootkit) from legitimate programs. *See, e.g.*, EX1008, 6:65-7:35 (describing that any one activity may be suspicious but not dispositive of malware, but that a collection of suspicious activities provides a "strong heuristic indicator" of malware)).

XI. ADDITIONAL REMARKS

I currently hold the opinions expressed in this declaration. But my analysis may continue, and I may acquire additional information and/or attain supplemental insights that may result in added observations.

I hereby declare that all statements made are of my own knowledge are true and that all statements made on information and belief are believed to be true. I further declare that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of the Title 18 of the United States Code and that such willful false statements may jeopardize the validity of this proceeding.

Dated: June 30, 2023

By: 
Dr. Markus Jakobsson

APPENDIX A

CV and Research Statement

Markus Jakobsson

www.linkedin.com/in/markusjakobsson

www.markus-jakobsson.com

1 At a Glance

- **Focus.** *Identification of security problems, trends and solution along four axes – computational, structural, physical and social; quantitative and qualitative fraud analysis; development of disruptive security technologies.*
- **Education.** *PhD* (Computer Science/Cryptography, University of California at San Diego, 1997); *MSc* (Computer Science, University of California at San Diego, 1994); *MSc* (Computer Engineering, Lund Institute of Technology, Sweden, 1993).
- **Large research labs.** *San Diego Supercomputer Center* (Researcher, 1996-1997); *Bell Labs* (Member of Technical Staff, 1997-2001); *RSA Labs* (Principal Research Scientist, 2001-2004); *Xerox PARC* (Principal Scientist, 2008-2010); *PayPal* (Principal Scientist of Consumer Security, Director, 2010-2013); *Qualcomm* (Senior Director, 2013-2015); *Agari* (Chief Scientist, 2016–2018); *Amber Solutions Inc* (Chief of Security and Data Analytics, 2018 – 2019); *ByteDance* (Principal Scientist, 2020-2021)
- **Academia.** *New York University* (Adjunct Associate Professor, 2002-2004); *Indiana University* (Associate Professor & Associate Director, 2004-2008; Adjunct Associate Professor, 2008-2016).
- **Entrepreneurial activity.** *ZapFraud* (Anti-scam technology; CTO and founder, 2012-current); *RavenWhite Security* (Authentication solutions; CTO and founder, 2005-); *RightQuestion* (Consulting; Founder, 2007-current); *FatSkunk* (Malware detection; CTO and founder, 2009-2013 – FatSkunk was acquired by Qualcomm); *LifeLock* (Id theft protection; Member of fraud advisory board, 2009-2013); *CellFony* (Mobile security; Member of technical advisory board, 2009-2013); *PopGiro* (User Reputation; Member of technical advisory board, 2012-2013); *MobiSocial* (Social networking, Member of technical advisory board, 2013); *Cequence Security* (Anti-fraud, Member of technical advisory board, 2013–current)
- **Anti-fraud consulting.** *KommuneData* [Danish govt. entity] (1996); *J.P. Morgan Chase* (2006-2007); *PayPal* (2007-2011); *Boku* (2009-2010); *Western Union* (2009-2010).

- **Intellectual Property, Testifying Expert Witness.** *Inventor of 100+ patents; expert witness in 25+ patent litigation cases* (McDermott, Will & Emery; Bereskin & Parr; WilmerHale; Hunton & Williams; Quinn Emanuel Urquhart & Sullivan; Freed & Weiss; Berry & Domer; Fish & Richardson; DLA Piper; Cipher Law Group; Kecker & Van Nest). Details and references upon request.
- **Publications.** Books: *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft* (Wiley, 2006); *Crime-ware: Understanding New Attacks and Defenses* (Symantec Press, 2008); *Towards Trustworthy Elections: New Directions in Electronic Voting* (Springer Verlag, 2010); *Mobile Authentication: Problems and Solutions* (Springer Verlag, 2012); *The Death of the Internet* (Wiley, 2012); *Understanding Social Engineering* (Springer Verlag, 2016); *Security, Privacy and User Interaction* (Springer Verlag, 2020); *100+ peer-reviewed publications*

2 At a Glance

Ten years before Bitcoin was created, I formalized the notion of Proof of Work and described its use for mining of crypto payments. I later developed energy-efficient alternatives to this paradigm, and showed how to enable mining on mobile devices, which is not possible for Bitcoin. I am the founder of the academic discipline of phishing and have developed techniques to predict fraud trends years before they emerge, enabling countermeasures to be developed before they are needed. I developed the notion of implicit authentication, which is now ubiquitous; I also founded a company that developed the first retroactive virus detection technology, with guarantees of detection; the company was acquired by Qualcomm in 2013. I have worked as chief scientist and similar positions in startups as well as industry behemoths, such as PayPal. I have several hundred patents to my name and am a prominent security researcher with hundreds of peer reviewed publications and an array of textbooks. My 1997 PhD thesis, from University of California at San Diego, was on distributed electronic payment systems with revocable privacy.

3 Work History (Highlights)

1. **Member of Technical Staff, Bell Labs (1997-2001).** Markus was part of the security research group at Bell Labs. He formalized the notion of *proof of work*, later an integral part of BitCoin.
2. **Principal Scientist, RSA Labs (2001-2005).** Markus posited that phishing would become a mainstream problem, and developed ethical techniques for identifying likely trends based on human subject experiments.
3. **Associate Professor, Indiana University (2005-2008).** Markus was hired to lead the newly formed security group at Indiana University, and

created a research group comprising approximately 10 professors and 30 students, studying social engineering and fraud.

4. **Principal Scientist, Xerox PARC (2008-2010).** Markus was hired to lead the research efforts of the Xerox PARC security group, and developed the notion of *implicit authentication*, a technology that is now ubiquitous.
5. **Principal Scientist, PayPal (2010-2013).** Markus did research on security and user interfaces, and developed techniques to reduce the losses associated with *liar buyer fraud*.
6. **Senior Director, Qualcomm (2013-2016).** Qualcomm acquires FatSkunk, a company founded by Markus. At FatSkunk, Markus developed *retroactive* malware detection with provable security guarantees. A simplified version of this is now deployed with almost all Qualcomm chipsets.
7. **Chief Scientist, Agari (2016-2018).** Markus developed a technique to acquire fraudster intelligence by compromising scammer email accounts – *while staying within the law* – resulting in the extradition of several African scam lords to the U.S.
8. **Chief of Security and Data Analytics, Amber Solutions (2018-2020).** Markus developed usable configuration methods supporting improved security and privacy for IoT installations.
9. **Chief Scientist, ByteDance (2020-2021).** Markus oversaw the establishment of a research group and a research agenda at ByteDance, and contributed to their intellectual property and product security.
10. **Chief Scientist, Artema LABS (2021-).** Managing the research, product strategy, and patent strategy for Artema LABS, a Los Angeles based startup in the Crypto/NFT sector.

4 Publication List

Books (1-8); book chapters, journals, conference publications and other scientific publications (9-147), issued /published U.S. patents (148-234). For an updated list, and for international patents, please see www.markus-jakobsson.com/publications and appropriate patent search engines.

References

- [1] M. Jakobsson, *Security, Privacy and User Interaction*, ISBN 978-3-030-43753-4, 110 pages, 2020.
- [2] M. Jakobsson, *Understanding Social Engineering Based Scams*, ISBN 978-1-4939-6457-4, 130 pages, Springer, 2016.

- [3] M. Jakobsson, *Mobile Authentication: Problems and Solutions*, ISBN 1461448778, 125 pages, Springer, 2013.
- [4] M. Jakobsson, (editor) *The Death of the Internet*, ASIN B009CN2JVE, 359 pages, IEEE Computer Society Press, 2012.
- [5] D. Chaum, M. Jakobsson, R. L. Rivest, P. Y. Ryan, J. Benaloh, and M. Kutyłowski, (editors), *Towards Trustworthy Elections: New Directions in Electronic Voting*, 411 pages, (Vol. 6000), Springer, 2010.
- [6] M. Jakobsson and Z. Ramzan (editors), *Crimeware: Trends in Attacks and Countermeasures*, ISBN 0321501950, Hardcover, 582 pages, Symantec Press / Addison Wesley, 2008.
- [7] M. Jakobsson and S. A. Myers (editors), *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft*, ISBN 0-471-78245-9, Hardcover, 739 pages, Wiley, 2006.
- [8] M. Jakobsson, M. Yung, J. Zhou, *Applied Cryptography and Network Security: Second International Conference , Yellow Mountain, China, 2004*, 511 pages, Lecture Notes in Computer Science (Book 3089), 2004.
- [9] M. Jakobsson, “Permissions and Privacy,” in *IEEE Security & Privacy*, vol. 18, no. 2, pp. 46-55, March-April 2020
- [10] M. Jakobsson, “The Rising Threat of Launchpad Attacks,” in *IEEE Security & Privacy*, vol. 17, no. 5, pp. 68-72, Sept.-Oct. 2019
- [11] J Koven, C Felix, H Siadati, M Jakobsson, E Bertini, “Lessons learned developing a visual analytics solution for investigative analysis of scamming activities,” *IEEE transactions on visualization and computer graphics* 25 (1), 225-234
- [12] M Jakobsson, “Two-factor inauthentication?the rise in SMS phishing attacks” *Computer Fraud & Security* 2018 (6), 6-8
- [13] M. Dhiman, M. Jakobsson, T.-F. Yen, “Breaking and fixing content-based filtering,” 2017 APWG Symposium on Electronic Crime Research (eCrime), 52-56
- [14] M. Jakobsson, “Addressing sophisticated email attacks,” 2017 International Conference on Financial Cryptography and Data Security, 310-317
- [15] M. Jakobsson, “User trust assessment: a new approach to combat deception,” *Proceedings of the 6th Workshop on Socio-Technical Aspects in Security and Trust*, 2016, pages 73-78
- [16] H. Siadati, T. Nguyen, P. Gupta, M. Jakobsson, “Mind your SMSes: Mitigating social engineering in second factor authentication,” *Computers and Security*, 2016

- [17] M. Jakobsson, W. Leddy, “Could you fall for a scam? Spam filters are passe. What we need is software that unmasks fraudsters,” *IEEE Spectrum* 53 (5), 2016, 40-55
- [18] N. Sae-Bae, M. Jakobsson, *Hand Authentication on Multi-Touch Tablets*, HotMobile 2014
- [19] Y. Park, J. Jones, D. McCoy, E. Shi, M. Jakobsson, *Scambaiter: Understanding Targeted Nigerian Scams on Craigslist*, NDSS 2014
- [20] D. Balfanz, R. Chow, O. Eisen, M. Jakobsson, S. Kirsch, S. Matsumoto, J. Molina, and P. van Oorschot, “The future of authentication,” *Security & Privacy, IEEE*, 10(1), 22-27, 2012.
- [21] M. Jakobsson, and H. Siadati, *Improved Visual Preference Authentication: Socio-Technical Aspects in Security and Trust*, (STAST), 2012 Workshop on IEEE, 27–34, 2012.
- [22] M. Jakobsson, R. I. Chow, and J. Molina, “Authentication-Are We Doing Well Enough?[Guest Editors’ Introduction]” *Security & Privacy, IEEE*, 10(1), 19-21, 2012.
- [23] E. Shi, Y. Niu, M. Jakobsson, and R. Chow, “Implicit authentication through learning user behavior,” *Information Security*, 99-113, Springer Berlin Heidelberg, 2011.
- [24] M. Jakobsson and K. Johansson, “Practical and Secure Software-Based Attestation,” *Lightweight Security & Privacy: Devices, Protocols and Applications (LightSec)*, 1–9, 2011.
- [25] A. Juels, D. Catalano, and M. Jakobsson, *Coercion-resistant electronic elections: Towards Trustworthy Elections*, 37–63, Springer Berlin Heidelberg, 2010.
- [26] M. Jakobsson and F. Menczer, “Web Forms and Untraceable DDoS Attacks,” in *Network Security*, Huang, S., MacCallum, D., and Du, D. Z., Eds., 77–95, Springer, 2010.
- [27] R. Chow, M. Jakobsson, R. Masuoka, J. Molina, Y. Niu, E. Shi, and Z. Song, “Authentication in the Clouds: A Framework and its Application to Mobile Users,” 2010.
- [28] X. Wang, P. Golle, M. Jakobsson, and A. Tsow, “Deterring voluntary trace disclosure in re-encryption mix-networks,” *ACM Trans. Inf. Syst. Secur.*, 13(2), 1-24, 2010.
- [29] X. Wang, P. Golle, M. Jakobsson, A. Tsow, “Deterring voluntary trace disclosure in re-encryption mix-networks,” *ACM Trans. Inf. Syst. Secur.* 13(2): (2010)

- [30] M. Jakobsson, and C. Soghoian, "Social Engineering in Phishing," Information Assurance, Security and Privacy Services, 4, 2009.
- [31] M. Jakobsson, C. Soghoian and S. Stamm, "Phishing," Handbook of Financial Cryptography (CRC press, 2008)
- [32] M. Jakobsson and A. Tsow, "Identity Theft," In John R. Vacca, Editor, "Computer And Information Security Handbook" (Morgan Kaufmann, 2008)
- [33] S. Srikwan and M. Jakobsson, "Using Cartoons to Teach Internet Security," Cryptologia, vol. 32, no. 2, 2008
- [34] M. Jakobsson, N. Johnson and P. Finn, "Why and How to Perform Fraud Experiments," IEEE Security and Privacy, March/April 2008 (Vol. 6, No. 2) pp. 66-68
- [35] M. Jakobsson and S. Myers, "Delayed Password Disclosure," International Journal of Applied Cryptography, 2008, pp. 47-59.
- [36] M. Jakobsson and S. Stamm, "Web Camouflage: Protecting Your Clients from Browser Sniffing Attacks," IEEE Security & Privacy Magazine. November/December 2007
- [37] P. Finn and M. Jakobsson, "Designing and Conducting Phishing Experiments," IEEE Technology and Society Magazine, Special Issue on Usability and Security
- [38] T. Jagatic, N. Johnson, M. Jakobsson and F. Menczer. "Social Phishing," The Communications of the ACM, October 2007
- [39] A. Tsow, M. Jakobsson, L. Yang and S. Wetzel, "Warkitting: the Drive-by Subversion of Wireless Home Routers," Anti-Phishing and Online Fraud, Part II Journal of Digital Forensic Practice, Volume 1, Special Issue 3, November 2006
- [40] M. Gandhi, M. Jakobsson and J. Ratkiewicz, "Badvertisements: Stealthy Click-Fraud with Unwitting Accessories," Anti-Phishing and Online Fraud, Part I Journal of Digital Forensic Practice, Volume 1, Special Issue 2, November 2006
- [41] N. Ben Salem, J.-P. Hubaux and M. Jakobsson. "Reputation-based Wi-Fi Deployment," Mobile Computing and Communications Review, Volume 9, Number 3 (Best papers of WMASH 2004)
- [42] N. Ben Salem, J. P. Hubaux, and M. Jakobsson. "Node Cooperation in Hybrid Ad hoc Networks," IEEE Transactions on Mobile Computing, Vol. 5, No. 4, April 2006.
- [43] P. MacKenzie, T. Shrimpton, and M. Jakobsson. "Threshold Password-Authenticated Key Exchange," Journal of Cryptology, 2005

- [44] A. Juels, M. Jakobsson, E. Shriver, and B. Hillyer. “How To Turn Loaded Dice Into Fair Coins.” *IEEE Transactions on Information Theory*, vol. 46(3). May 2000. pp. 911–921.
- [45] M. Jakobsson, P. MacKenzie, and J.P. Stern. “Secure and Lightweight Advertising on the Web,” *Journal of Computer Networks*, vol. 31, issue 11–16, Elsevier North-Holland, Inc., 1999. pp. 1101–1109.
- [46] M. Jakobsson, “Cryptographic Protocols,” Chapter from *The Handbook of Information Security*. Hossein Bidgoli, Editor-in-Chief. Copyright John Wiley & Sons, Inc., 2005, Hoboken, N.J.
- [47] M. Jakobsson, “Cryptographic Privacy Protection Techniques,” Chapter from *The Handbook of Information Security*. Hossein Bidgoli, Editor-in-Chief. Copyright John Wiley & Sons, Inc., 2005, Hoboken, N.J.
- [48] M. Jakobsson, E. Shi, P. Golle, R. Chow, “Implicit authentication for mobile devices,” 4th USENIX Workshop on Hot Topics in Security (HotSec ’09); 2009 August 11; Montreal, Canada.
- [49] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, J. Molina, “Controlling data in the cloud: outsourcing computation without outsourcing control,” *Proceedings of the 2009 ACM Workshop on Cloud Computing Security (CCSW 2009)*; 2009 November 13; Chicago, IL. NY: ACM; 2009; pp. 85–90.
- [50] M. Jakobsson, H. Siadati, M. Dhiman, “Liar Buyer Fraud, and How to Curb It,” NDSS, 2015
- [51] M. Jakobsson, T.-F. Yen, “How Vulnerable Are We To Scams?,” BlackHat, 2015
- [52] M. Jakobsson, “How to Wear Your Password,” BlackHat, 2014
- [53] M. Jakobsson and G. Stewart, “Mobile Malware: Why the Traditional AV Paradigm is Doomed, and How to Use Physics to Detect Undesirable Routines,” in BlackHat, 2013.
- [54] M. Jakobsson, and H. Siadati, “SpoofKiller: You Can Teach People How to Pay, but Not How to Pay Attention” in *Socio-Technical Aspects in Security and Trust (STAST)*, 2012 Workshop on, 3-10, 2012.
- [55] M. Jakobsson, and M. Dhiman, “The benefits of understanding passwords,” in *Proceedings of the 7th USENIX conference on Hot Topics in Security*, Berkeley, CA, USA, 2012.
- [56] M. Jakobsson, and S. Taveau, “The Case for Replacing Passwords with Biometrics,” *Mobile Security Technologies*, 2012.
- [57] M. Jakobsson and D. Liu, “Bootstrapping mobile PINs using passwords,” W2SP, 2011.

- [58] M. Jakobsson and R. Akavipat, “Rethinking passwords to adapt to constrained keyboards,” 2011.
- [59] Y. Niu, E. Shi, R. Chow, P. Golle, and M. Jakobsson, “One Experience Collecting Sensitive Mobile Data,” In USER Workshop of SOUPS, 2010.
- [60] E. Shi, Y. Niu, M. Jakobsson, and R. Chow, “Implicit Authentication through Learning User Behavior,” 2010.
- [61] M. Jakobsson and K. Johansson, Assured Detection of Malware With Applications to Mobile Platforms, 2010.
- [62] M. Jakobsson and K. Johansson, “Retroactive Detection of Malware With Applications to Mobile Platforms,” in HotSec 2010, Washington, DC, 2010.
- [63] M. Jakobsson, A Central Nervous System for Automatically Detecting Malware, 2009.
- [64] R. Chow, P. Golle, M. Jakobsson, R. Masuoka, J. Molina, E. Shi, and J. Staddon, “Controlling data in the cloud: outsourcing computation without outsourcing control,” ACM workshop on Cloud computing security (CCSW), 2009.
- [65] M. Jakobsson and A. Juels, “Server-Side Detection of Malware Infection,” in New Security Paradigms Workshop (NSPW), Oxford, UK, 2009.
- [66] M. Jakobsson, “Captcha-free throttling,” Proceedings of the 2nd ACM workshop on Security and artificial intelligence, 15–22, 2009.
- [67] M. Jakobsson, E. Shi, P. Golle, and R. Chow, “Implicit authentication for mobile devices,” Proceedings of the 4th USENIX conference on Hot topics in security, 9–9, 2009.
- [68] C. Soghoian, O. Friedrichs and M. Jakobsson, “The Threat of Political Phishing,” International Symposium on Human Aspects of Information Security & Assurance (HAISA 2008)
- [69] R. Chow, P. Golle, M. Jakobsson, L. Wang and X. Wang, “Making CAPTCHAs Clickable,” In proc. of HotMobile 2008.
- [70] M. Jakobsson, A. Juels, and J. Ratkiewicz, “Privacy-Preserving History Mining for Web Browsers,” Web 2.0 Security and Privacy, 2008.
- [71] M. Jakobsson, E. Stolterman, S. Wetzel, L. Yang, “Love and Authentication,” (Notes) ACM Computer/Human Interaction Conference (CHI), 2008. Also see www.I-forgot-my-password.com
- [72] M. Jakobsson and S. Myers, “Delayed Password Disclosure,” Proceedings of the 2007 ACM workshop on Digital Identity Management
- [73] M. Jakobsson, S. Stamm, Z. Ramzan, “JavaScript Breaks Free,” W2SP ’07

- [74] A. Juels, S. Stamm, M. Jakobsson, “Combatting Click Fraud via Premium Clicks,” USENIX Security 2007
- [75] R. Chow, P. Golle, M. Jakobsson, X. Wang, “Clickable CAPTCHAs,” Ad-Fraud ’07 Workshop; 2007 September 14; Stanford, CA, USA
- [76] S. Stamm, Z. Ramzan, and M. Jakobsson, “Drive-by Pharming,” In Proceedings of Information and Communications Security, 9th International Conference, ICICS 2007
- [77] M. Jakobsson, A. Tsow, A. Shah, E. Blevis, Y.-K. Lim, “What Instills Trust? A Qualitative Study of Phishing,” USEC ’07.
- [78] R. Akavipat, V. Anandpara, A. Dingman, C. Liu, D. Liu, K. Pongsanon, H. Roinestad and M. Jakobsson, “Phishing IQ Tests Measure Fear, not Ability,” USEC ’07.
- [79] M. Jakobsson, “The Human Factor in Phishing,” American Conference Institute’s Forum on Privacy & Security of Consumer Information, 2007
- [80] S. Srikwan, M. Jakobsson, A. Albrecht and M. Dalkilic, “Trust Establishment in Data Sharing: An Incentive Model for Biodiversity Information Systems,” TrustCol 2006
- [81] J.Y. Choi, P. Golle, M. Jakobsson, “Tamper-Evident Digital Signatures: Protecting Certification Authorities Against Malware,” DACS ’06
- [82] L. Yang, M. Jakobsson, S. Wetzel, “Discount Anonymous On Demand Routing for Mobile Ad hoc Networks,” SECURECOMM ’06
- [83] P. Golle, X. Wang, M. Jakobsson, A. Tsow, “Deterring Voluntary Trace Disclosure in Re-encryption Mix Networks.” IEEE S&P ’06
- [84] M. Jakobsson, A. Juels, T. Jagatic, “Cache Cookies for Browser Authentication (Extended Abstract),” IEEE S&P ’06
- [85] M. Jakobsson and J. Ratkiewicz, “Designing Ethical Phishing Experiments: A study of (ROT13) rOnl auction query features.”, WWW ’06
- [86] M. Jakobsson and S. Stamm. “Invasive Browser Sniffing and Countermeasures,” WWW ’06
- [87] J.Y. Choi, P. Golle and M. Jakobsson. “Auditable Privacy: On Tamper-Evident Mix Networks,” Financial Crypto ’06
- [88] A. Juels, D. Catalano and M. Jakobsson. “Coercion-Resistant Electronic Elections,” WPES ’05
- [89] V. Griffith and M. Jakobsson. “Messin’ with Texas, Deriving Mother’s Maiden Names Using Public Records,” ACNS ’05, 2005.

- [90] M. Jakobsson and L. Yang. “Quantifying Security in Hybrid Cellular Networks,” ACNS ’05, 2005
- [91] Y.-C. Hu, M. Jakobsson, and A. Perrig. “Efficient Constructions for One-way Hash Chains,” ACNS ’05, 2005
- [92] M. Jakobsson. “Modeling and Preventing Phishing Attacks,” Phishing Panel in Financial Cryptography ’05. 2005, abstract in proceedings.
- [93] N. Ben Salem, J.-P. Hubaux, and M. Jakobsson. “Reputation-based Wi-Fi Deployment Protocols and Security Analysis,” In WMASH ’04. ACM Press, 2004. pp. 29–40.
- [94] M. Jakobsson and S. Wetzel. “Efficient Attribute Authentication with Applications to Ad Hoc Networks,” In VANET ’04. ACM Press, 2004. pp. 38–46.
- [95] M. Jakobsson, X. Wang, and S. Wetzel. “Stealth Attacks in Vehicular Technologies,” Invited paper. In Proceedings of IEEE Vehicular Technology Conference 2004 Fall (VTC-Fall 2004). IEEE, 2004.
- [96] A. Ambainis, H. Lipmaa, and M. Jakobsson. “Cryptographic Randomized Response Technique,” In PKC ’04. LNCS 2947. Springer-Verlag, 2004. pp. 425–438.
- [97] P. Golle, M. Jakobsson, A. Juels, and P. Syverson. “Universal Re-encryption for Mixnets,” In CT-RSA ’04. LNCS 2964. Springer-Verlag, 2004. pp. 163–178.
- [98] P. Golle and M. Jakobsson. “Reusable Anonymous Return Channels,” In WPES ’03. ACM Press, 2003. pp. 94–100.
- [99] M. Jakobsson, S. Wetzel, B. Yener. “Stealth Attacks on Ad-Hoc Wireless Networks,” In IEEE VTC ’03, 2003.
- [100] N. Ben Salem, L. Buttyan, J.-P. Hubaux, and M. Jakobsson. “A Charging and Rewarding Scheme for Packet Forwarding in Multi-hop Cellular Networks,” In ACM MobiHoc ’03. ACM Press, 2003. pp. 13–24.
- [101] M. Jakobsson, J.-P. Hubaux and L. Buttyan. “A Micro-Payment Scheme Encouraging Collaboration in Multi-Hop Cellular Networks,” In FC ’03. LNCS 2742. Springer-Verlag, 2003. pp. 15–33.
- [102] M. Jakobsson, T. Leighton, S. Micali and M. Szydlo. “Fractal Merkle Tree Representation and Traversal,” In RSA-CT ’03 2003.
- [103] A. Boldyreva and M. Jakobsson. “Theft protected proprietary certificates,” In DRM ’02. LNCS 2696, 2002. pp. 208–220.

- [104] P. Golle, S. Zhong, M. Jakobsson, A. Juels, and D. Boneh. “Optimistic Mixing for Exit-Polls,” In *Asiacrypt '02*. LNCS 2501. Springer-Verlag, 2002. pp. 451–465.
- [105] P. MacKenzie, T. Shrimpton, and M. Jakobsson. “Threshold Password-Authenticated Key Exchange,” In *CRYPTO '02*. LNCS 2442. Springer-Verlag, 2002. pp. 385–400.
- [106] M. Jakobsson. “Fractal Hash Sequence Representation and Traversal,” In *Proceedings of the 2002 IEEE International Symposium on Information Theory (ISIT '02)*. 2002. pp. 437–444.
- [107] M. Jakobsson, A. Juels, and R. Rivest. “Making Mix Nets Robust For Electronic Voting By Randomized Partial Checking,” In *Proceedings of the 11th USENIX Security Symposium*. USENIX Association, 2002. pp. 339–353.
- [108] D. Coppersmith and M. Jakobsson. “Almost Optimal Hash Sequence Traversal,” In *Financial Crypto '02*. 2002.
- [109] M. Jakobsson. “Financial Instruments in Recommendation Mechanisms,” In *Financial Crypto '02*. 2002.
- [110] J. Garay, and M. Jakobsson. “Timed Release of Standard Digital Signatures,” In *Financial Crypto '02*. 2002.
- [111] F. Menczer, N. Street, N. Vishwakarma, A. Monge, and M. Jakobsson. “Intellishopper: A Proactive, Personal, Private Shopping Assistant,” In *AAMAS '02*. ACM Press, 2002. pp. 1001–1008.
- [112] M. Jakobsson, A. Juels, and P. Nguyen. “Proprietary Certificates,” In *CT-RSA '02*. LNCS 2271. Springer-Verlag, 2002. pp. 164–181.
- [113] M. Jakobsson and A. Juels. “An Optimally Robust Hybrid Mix Network,” In *PODC '01*. ACM Press. 2001. pp. 284–292.
- [114] M. Jakobsson and M. Reiter. “Discouraging Software Piracy Using Software Aging,” In *DRM '01*. LNCS 2320. Springer-Verlag, 2002. pp. 1–12.
- [115] M. Jakobsson and S. Wetzel. “Security Weaknesses in Bluetooth,” In *CT-RSA '01*. LNCS 2020. Springer-Verlag, 2001. pp. 176–191.
- [116] M. Jakobsson and D. Pointcheval. “Mutual Authentication for Low-Power Mobile Devices,” In *Financial Crypto '01*. LNCS 2339. Springer-Verlag, 2001. pp. 178–195.
- [117] M. Jakobsson, D. Pointcheval, and A. Young. “Secure Mobile Gambling,” In *CT-RSA '01*. LNCS 2020. Springer-Verlag, 2001. pp. 110–125.
- [118] M. Jakobsson and S. Wetzel. “Secure Server-Aided Signature Generation,” In *PKC '01*. LNCS 1992. Springer-Verlag, 2001. pp. 383–401.

- [119] M. Jakobsson and A. Juels. “Addition of ElGamal Plaintexts,” In T. Okamoto, ed., ASIACRYPT ’00. LNCS 1976. Springer-Verlag, 2000. pp. 346–358.
- [120] M. Jakobsson, and A. Juels. “Mix and Match: Secure Function Evaluation via Ciphertexts,” In ASIACRYPT ’00. LNCS 1976. Springer-Verlag, 2000. pp. 162–177.
- [121] R. Arlein, B. Jai, M. Jakobsson, F. Monrose, and M. Reiter. “Privacy-Preserving Global Customization,” In ACM E-Commerce ’00. ACM Press, 2000. pp. 176–184.
- [122] C.-P. Schnorr and M. Jakobsson. “Security of Signed ElGamal Encryption,” In ASIACRYPT ’00. LNCS 1976. Springer-Verlag, 2000. pp. 73–89.
- [123] P. Bohannon, M. Jakobsson, and S. Srikwan. “Cryptographic Approaches to Privacy in Forensic DNA Databases,” In Public Key Cryptography ’00. LNCS 1751. Springer-Verlag, 2000, pp. 373–390.
- [124] J. Garay, M. Jakobsson, and P. MacKenzie. “Abuse-free Optimistic Contract Signing,” In CRYPTO ’99. LNCS 1666. Springer-Verlag, 1999. pp. 449–466.
- [125] M. Jakobsson. “Flash Mixing,” In PODC ’99. ACM Press, 1999. pp. 83–89.
- [126] G. Di Crescenzo, N. Ferguson, R. Impagliazzo, and M. Jakobsson. “How To Forget a Secret,” In STACS ’99. LNCS 1563. Springer-Verlag, 1999. pp. 500–509.
- [127] M. Jakobsson, D. M’Raihi, Y. Tsiounis, and M. Yung. “Electronic Payments: Where Do We Go from Here?,” In CQRE (Secure) ’99. LNCS 1740. Springer-Verlag, 1999. pp. 43–63.
- [128] C.P. Schnorr and M. Jakobsson. “Security Of Discrete Log Cryptosystems in the Random Oracle + Generic Model,” In Conference on The Mathematics of Public-Key Cryptography. 1999.
- [129] M. Jakobsson and A. Juels “Proofs of Work and Breadpudding Protocols,” In CMS ’99. IFIP Conference Proceedings, Vol. 152. Kluwer, B.V., 1999. pp. 252 – 272.
- [130] M. Jakobsson and C-P Schnorr. “Efficient Oblivious Proofs of Correct Exponentiation,” In CMS ’99. IFIP Conference Proceedings, Vol. 152. Kluwer, B.V., 1999. pp. 71–86.
- [131] M. Jakobsson, P. MacKenzie, and J.P. Stern. “Secure and Lightweight Advertising on the Web,” In World Wide Web ’99

- [132] M. Jakobsson, J.P. Stern, and M. Yung. “Scramble All, Encrypt Small,” In *Fast Software Encryption '99*. LNCS 1636. Springer-Verlag, 1999. pp. 95–111.
- [133] M. Jakobsson and J. Mueller. “Improved Magic Ink Signatures Using Hints,” In *Financial Cryptography '99*. LNCS 1648. Springer-Verlag, 1999. pp. 253–268.
- [134] M. Jakobsson. “Mini-Cash: A Minimalistic Approach to E-Commerce,” In *Public Key Cryptography '99*. LNCS 1560. Springer-Verlag, 1999. pp. 122–135.
- [135] M. Jakobsson. “On Quorum Controlled Asymmetric Proxy Re-encryption,” In *Public Key Cryptography '99*. LNCS 1560. Springer-Verlag, 1999. pp. 112–121.
- [136] M. Jakobsson and A. Juels. “X-Cash: Executable Digital Cash,” In *Financial Cryptography '98*. LNCS 1465. Springer-Verlag, 1998. pp. 16–27.
- [137] M. Jakobsson and D. M'Raihi. “Mix-based Electronic Payments,” In *Proceedings of the Selected Areas in Cryptography*. LNCS 1556. Springer-Verlag, 1998. pp. 157–173.
- [138] M. Jakobsson, E. Shriver, B. Hillyer, and A. Juels. “A Practical Secure Physical Random Bit Generator,” In *CCS '98: Proceedings of the 5th ACM conference on Computer and communications security*. ACM Press, 1998. pp. 103–111.
- [139] M. Jakobsson. “A Practical Mix,” In *Advances in Cryptology – EuroCrypt '98*. LNCS 1403. Springer-Verlag, 1998. pp. 448–461.
- [140] M. Jakobsson and M. Yung. “On Assurance Structures for WWW Commerce,” In *Financial Cryptography '98*. LNCS 1465. Springer-Verlag, 1998. pp. 141–157.
- [141] E. Gabber, M. Jakobsson, Y. Matias, and A. Mayer. “Curbing Junk E-Mail via Secure Classification,” In *Financial Cryptography '98*. LNCS 1465. Springer-Verlag, 1998. pp. 198–213.
- [142] M. Jakobsson and M. Yung. “Distributed ‘Magic Ink’ Signatures,” In *Advances in Cryptology – EuroCrypt '97*. LNCS 1233. Springer-Verlag, 1997. pp. 450–464.
- [143] M. Jakobsson and M. Yung. “Applying Anti-Trust Policies to Increase Trust in a Versatile E-Money System,” In *Financial Cryptography '97*. LNCS 1318. Springer-Verlag, 1997. pp. 217–238.
- [144] A. Herzberg, M. Jakobsson, S. Jarecki, H. Krawczyk, and M. Yung. “Proactive public-key and signature schemes,” In *Proceedings of the 4th Annual Conference on Computer Communications Security*. ACM Press, 1997. pp. 100–110.

- [145] M. Bellare, M. Jakobsson, and M. Yung. “Round-Optimal Zero-Knowledge Arguments Based on any One-Way Function,” In *Advances in Cryptology – EuroCrypt ’97*. LNCS 1233. Springer-Verlag, 1997. pp. 280–305.
- [146] M. Jakobsson and M. Yung. “Proving Without Knowing,” In *Crypto ’96*. LNCS 1109. Springer-Verlag, 1996. pp. 186–200.
- [147] M. Jakobsson, K. Sako, and R. Impagliazzo. “Designated Verifier Proofs and Their Applications,” In *Advances in Cryptology – EuroCrypt ’96*. LNCS 1070. Springer-Verlag, 1996. pp. 143–154.
- [148] M. Jakobsson and M. Yung. “Revokable and Versatile Electronic Money,” In *CCS ’96: Proceedings of the 3rd ACM conference on Computer and communications security*. ACM Press, 1996. pp. 76–87.
- [149] M. Jakobsson. “Ripping Coins for a Fair Exchange,” In *Advances in Cryptology – EuroCrypt ’95*. LNCS 921. Springer-Verlag, 1995. pp. 220–230.
- [150] M. Jakobsson. “Blackmailing using Undeniable Signatures,” In *Advances in Cryptology EuroCrypt ’94*. LNCS 950. Springer-Verlag, 1994. pp. 425–427.
- [151] M. Jakobsson. “Reducing costs in identification protocols,” *Crypto ’92*, 1992.
- [152] M. Jakobsson. “Machine-Generated Music with Themes,” In *International Conference on Artificial Neural Networks ’92*. Vol 2. Amsterdam: Elsevier, 1992. pp. 1645–1646
- [153] M. Jakobsson, “Social Engineering 2.0: What’s Next,” *McAfee Security Journal*, Fall 2008
- [154] M. Jakobsson and S. Myers, “Delayed Password Disclosure,” *ACM SIGACT News archive*, Volume 38, Issue 3 (September 2007), pp. 56 - 75
- [155] V. Griffith and M. Jakobsson. “Messin’ with Texas, Deriving Mother’s Maiden Names Using Public Records,” *CryptoBytes*, 2007.
- [156] M. Jakobsson, A. Juels, J. Ratkiewicz, “Remote-Harm Detection,” Beta-version available at rhd.ravenwhitedevelopment.com/
- [157] S. Stamm, M. Jakobsson, “Social Malware,” Experimental results available at [www.indiana.edu/ phishing/verybigad/](http://www.indiana.edu/phishing/verybigad/)
- [158] M. Jakobsson. “Privacy vs. Authenticity,” Ph.D. Thesis, University of California at San Diego, 1997
- [159] Markus Jakobsson, Automatic PIN creation using passwords, US20130125214 A1, 2012.

- [160] Markus Jakobsson, Systems and methods for creating a user credential and authentication using the created user credential, US 20130111571 A1, 2012.
- [161] Markus Jakobsson, Password check by decomposing password, US 20120284783 A1, 2012.
- [162] Markus Jakobsson, William Leddy, System and methods for protecting users from malicious content, US 20120192277 A1, 2011.
- [163] Markus Jakobsson, Karl-Anders R. Johansson, Auditing a device, US8370935 B1, 2011.
- [164] Markus Jakobsson, Methods and Apparatus for Efficient Computation of One-Way Chains in Cryptographic Applications, US20120303969 A1, 2011.
- [165] Markus Jakobsson, Automatic PIN creation using password, US 20120110634 A1, 2011.
- [166] Markus Jakobsson, Jim Roy Palmer, Gustavo Maldonado, Interactive CAPTCHA, US20130007875 A1, 2011.
- [167] Markus Jakobsson, System access determination based on classification of stimuli, US 20110314559 A1, 2011.
- [168] Markus Jakobsson, System access determination based on classification of stimuli, WO 2011159356 A1, 2011.
- [169] Markus Jakobsson, Method, medium, and system for reducing fraud by increasing guilt during a purchase transaction, US8458041 B1, 2011.
- [170] Markus Jakobsson, Visualization of Access Information, US 20120233314 A1, 2011.
- [171] Markus Jakobsson, Richard Chow,Runting Shi, Implicit authentication, US20120137340 A1, 2010.
- [172] Markus Jakobsson, Karl-Anders R. Johansson, Auditing a device, EP2467793 A1, 2010.
- [173] Markus Jakobsson, Event log authentication using secure components, US 20110314297 A1, 2010.
- [174] Markus Jakobsson, Philippe J.P. Golle, Risk-based alerts, US 20110314426 A1, 2010.
- [175] Markus Jakobsson, Karl-Anders R. Johansson, Auditing a device, US 20110041178 A1, 2010.
- [176] M. Jakobsson, A. Juels, J. Kaliski Jr, S. Burton and others, Identity authentication system and method, US Patent 7,502,933, 2009.

- [177] Markus Jakobsson, Method and system for facilitating throttling of interpolation-based authentication, US8219810 B2, 2009.
- [178] Markus Jakobsson, Karl-Anders R. Johansson, Auditing a device, US8375442 B2, 2009.
- [179] Markus Jakobsson, Karl-Anders R. Johansson, Auditing a device, US 20110041180 A1, 2009.
- [180] Markus Jakobsson, Pattern-based application classification, US 20110055925 A1, 2009.
- [181] Markus Jakobsson, Method and apparatus for detecting cyber threats, US8286225 B2, 2009.
- [182] Markus Jakobsson, Method and apparatus for detecting cyber threats, US20110035784 A1, 2009.
- [183] Markus Jakobsson, CAPTCHA-free throttling, US8312073 B2, 2009.
- [184] Markus Jakobsson, Captcha-free throttling, US 20110035505 A1, 2009.
- [185] Markus Jakobsson, Christopher Soghoian, Method and apparatus for throttling access using small payments, US 20100153275 A1, 2008.
- [186] Markus Jakobsson, Christopher Soghoian, Method and apparatus for mutual authentication using small payments, US 20100153274 A1, 2008.
- [187] Philippe J.P. Golle, Markus Jakobsson, Richard Chow, Resetting a forgotten password using the password itself as authentication, US 20100125906 A1, 2008.
- [188] Philippe J. P. Golle, Markus Jakobsson, Richard Chow, Authenticating users with memorable personal questions, US8161534 B2, 2008.
- [189] Richard Chow, Philippe J.P. Golle, Markus Jakobsson, Jessica N. Staddon, Authentication based on user behavior, US20100122329 A1, 2008.
- [190] Richard Chow, Philippe J.P. Golle, Markus Jakobsson, Jessica N. Staddon, Enterprise password reset, US 20100122340 A1, 2008.
- [191] Markus Jakobsson, Methods and apparatus for efficient computation of one-way chains in cryptographic applications, US8086866 B2, 2008.
- [192] Richard Chow, Philippe J. P. Golle, Markus Jakobsson, Selectable captchas, US8307407 B2, 2008.
- [193] Markus Jakobsson, Ari Juels, Sidney Louis Stamm, Method and apparatus for combatting click fraud, US 20080162227 A1, 2007.
- [194] Jakobsson, Method and apparatus for evaluating actions performed on a client device, US 20080037791 A1, 2007.

- [195] Jakobsson, Ari Juels, Method and apparatus for storing information in a browser storage area of a client device, US 20070106748 A1, 2006.
- [196] Markus Jakobsson, Steven Andrew Myers, Anti-phishing logon authentication object oriented system and method, WO 2006062838 A1, 2005.
- [197] Jakobsson, Jean-Pierre Hubaux, Levente Buttyan, Micro-payment scheme encouraging collaboration in multi-hop cellular networks, US 20050165696 A1, 2004.
- [198] Andrew Nanopoulos, Karl Ackerman, Piers Bowness, William Duane, Markus Jakobsson, Burt Kaliski, Dmitri Pal, Shane D. Rice, Less , System and method providing disconnected authentication, WO2005029746 A3, 2004.
- [199] Andrew Nanopoulos, Karl Ackerman, Piers Bowness, William Duane, Markus Jakobsson, Burt Kaliski, Dmitri Pal, Shane Rice, Ronald Rivest, Less , System and method providing disconnected authentication, US 20050166263 A1, 2004.
- [200] Andrew Nanopoulos, Karl Ackerman, Piers Bowness, William Duane, Markus Jakobsson, Burt Kaliski, Dmitri Pal, Shane D. Rice, Less , Systeme et procede d'authentification deconnectee, WO 2005029746 A2, 2004.
- [201] Markus Jakobsson, Ari Juels, Burton S. Kaliski Jr., Identity authentication system and method, WO2004051585 A3, 2003.
- [202] Markus Jakobsson, Ari Juels, Burton S. Kaliski, Jr., Identity authentication system and method, US 7502933 B2, 2003.
- [203] Markus Jakobsson, Ari Juels, Burton S. Kaliski Jr., Systeme et procede de validation d'identite, WO 2004051585 A2, 2003.
- [204] Markus Jakobsson, Burton S. Kaliski, Jr., Method and apparatus for graph-based partition of cryptographic functionality, US7730518 B2, 2003.
- [205] Markus Jakobsson, Phong Q. Nguyen, Methods and apparatus for private certificates in public key cryptography, US7404078 B2, 2002.
- [206] Jakobsson, Philip MacKenzie, Thomas Shrimpton, Method and apparatus for performing multi-server threshold password-authenticated key exchange, US20030221102 A1, 2002.
- [207] Markus Jakobsson, Philip D MacKenzie, Method and apparatus for distributing shares of a password for use in multi-server password authentication, US 7073068 B2, 2002.
- [208] Markus Jakobsson, Methods and apparatus for efficient computation of one-way chains in cryptographic applications, EP 1389376 A1 (text from WO2002084944A1), 2002.

- [209] Markus Jakobsson, Adam Lucas Young, Method and apparatus for identification tagging documents in a computer system, US 7356845 B2, 2002.
- [210] Juan A. Garay, Markus Jakobsson, Methods and apparatus for computationally-efficient generation of secure digital signatures, US 7366911 B2, 2001.
- [211] Markus Jakobsson, Methods and apparatus for efficient computation of one-way chains in cryptographic applications, US 7404080 B2, 2001.
- [212] Markus Jakobsson, Susanne Gudrun Wetzel, Securing the identity of a bluetooth master device (bd addr) against eavesdropping by preventing the association of a detected channel access code (cac) with the identity of a particular bluetooth device, WO2002019641 A3, 2001.
- [213] Markus Jakobsson, Susanne Gudrun Wetzel, Procédé et appareil permettant d'assurer la sécurité d'utilisateurs de dispositifs à capacités bluetooth, WO 2002019641 A2, 2001.
- [214] Robert M. Arlein, Ben Jai, Markus Jakobsson, Fabian Monroe, Michael Kendrick Reiter, Less , Methods and apparatus for providing privacy-preserving global customization, US 7107269 B2, 2001.
- [215] Markus Jakobsson, Susanne Gudrun Wetzel, Secure distributed computation in cryptographic applications, US6950937 B2, 2001.
- [216] Markus Jakobsson, Susanne Gudrun Wetzel, Method and apparatus for ensuring security of users of short range wireless enable devices, US6981157 B2, 2001.
- [217] Markus Jakobsson, Susanne Gudrun Wetzel, Method and apparatus for ensuring security of users of bluetooth TM-enabled devices, US 6574455 B2, 2001.
- [218] Garay; Juan A. (West New York, NJ), Jakobsson; M. (Hoboken, NJ), Kristol; David M. (Summit, NJ), Mizikovsky; Semyon B.(Morganville, NJ), Cryptographic key processing and storage , 7023998, 2001.
- [219] Markus Jakobsson, Method, apparatus, and article of manufacture for generating secure recommendations from market-based financial instrument prices, US 6970839 B2, 2001.
- [220] Markus Jakobsson, Encryption method and apparatus with escrow guarantees, US7035403 B2, 2001.
- [221] Jakobsson, Call originator access control through user-specified pricing mechanism in a communication network, US20020099670 A1, 2001.
- [222] Markus Jakobsson, Claus Peter Schnorr, Tagged private information retrieval, US7013295 B2, 2000.

- [223] Markus Jakobsson, Secure enclosure for key exchange, US 7065655 B1, 2000.
- [224] Markus Jakobsson, Michael Kendrick Reiter, Software aging method and apparatus for discouraging software piracy, US 7003110 B1, 2000.
- [225] Markus Jakobsson, Probabilistic theft deterrence, US6501380 B1, 2000.
- [226] Markus Jakobsson, Michael Kendrick Reiter, Abraham Silberschatz, Anonymous and secure electronic commerce, EP 1150227 A1, 2000.
- [227] Markus Jakobsson, Ari Juels, Proofs of work and bread pudding protocols, US 7356696 B1, 2000.
- [228] Markus Jakobsson, Joy Colette Mueller, Methods of protecting against spam electronic mail, US7644274 B1, 2000.
- [229] Philip L. Bohannon, Markus Jakobsson, Fabian Monroe, Michael Kendrick Reiter, Susanne Gudrun Wetzel, Less , Generation of repeatable cryptographic key based on varying parameters, EP1043862 B1, 2000.
- [230] Markus Jakobsson, Ari Juels, Mix and match: a new approach to secure multiparty computation, US6772339 B1, 2000.
- [231] Markus Jakobsson, Ari Juels, Mixing in small batches, US6813354 B1, 2000.
- [232] Philip L. Bohannon, Markus Jakobsson, Fabian Monroe, Michael Kendrick Reiter, Susanne Gudrun Wetzel, Less , Generation of repeatable cryptographic key based on varying parameters, US 6901145 B1, 36566
- [233] Markus Jakobsson, Claus Peter Schnorr, Non malleable encryption method and apparatus using key-encryption keys and digital signature, US6931126 B1, 2000.
- [234] Markus Jakobsson, Claus Peter Schnorr, Non malleable encryption method and apparatus using key-encryption keys and digital signature, US 6931126 B1, 2000.
- [235] Markus Jakobsson, Flash mixing apparatus and method, US 6598163 B1, 1999.
- [236] Markus Jakobsson, Minimalistic electronic commerce system, US 6529884 B1, 1999.
- [237] Markus Jakobsson, Method and system for providing translation certificates, US 6687822 B1, 1999.
- [238] Markus Jakobsson, Verification of correct exponentiation or other operations in cryptographic applications, US6978372 B1, 1999.

- [239] Markus Jakobsson, Non malleable encryption apparatus and method, US 6507656 B1, 1999.
- [240] Markus Jakobsson, Method and system for quorum controlled asymmetric proxy encryption, US 6587946 B1, 1998.
- [241] Markus Jakobsson, Practical mix-based election scheme, US 6317833 B1, 1998.
- [242] Markus Jakobsson, Ari Juels, Method and apparatus for extracting unbiased random bits from a potentially biased source of randomness, US 6393447 B1, 1998.
- [243] Markus Jakobsson, Ari Juels, Executable digital cash for electronic commerce, US6157920 A, 1998.
- [244] Bruce Kenneth Hillyer, Markus Jakobsson, Elizabeth Shriver, Storage device random bit generator, US 6317499 B1, 1998.
- [245] Markus Jakobsson, Method and apparatus for encrypting, decrypting, and providing privacy for data values, US 6049613 A, 1998.