

UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE PATENT TRIAL AND APPEAL BOARD

ZIMPERIUM, INC.,
Petitioner,
v.
TAASERA LICENSING LLC,
Patent Owner.

U.S. Patent No. 9,071,518
Case No. IPR2024-00007

DECLARATION OF MARKUS JAKOBSSON, PH.D.

TABLE OF CONTENTS

I.	INTRODUCTION	5
II.	EDUCATION BACKGROUND, PROFESSIONAL EXPERIENCE, AND OTHER QUALIFICATIONS.....	5
III.	ASSIGNMENT AND MATERIALS CONSIDERED	11
IV.	UNDERSTANDING OF THE LAW	12
V.	LEVEL OF SKILL IN THE ART	16
VI.	THE '518 PATENT	18
A.	Effective Filing Date	18
B.	Overview	18
C.	Prosecution History	20
D.	Claims.....	22
VII.	CLAIM CONSTRUCTION.....	27
A.	“Substantially real time” and “Substantially real-time data”	28
B.	“Initiating ... at least one action” and “initiate an action”	29
VIII.	UNPATENTABILITY ANALYSIS	30
A.	Overview of the Prior Art.....	30
1.	Broch	30
2.	Mahaffey	33
3.	Stricklen	36
B.	Ground 1: Claims 1-10, 12, 14-17, and 20-23 Are Obvious Over Broch and Stricklen.....	37
1.	Independent Claim Analysis	37
a.	Claim 1.....	37

b.	Claim 10.....	56
c.	Claim 17.....	63
2.	Dependent Claim Analysis	67
a.	Claim 2.....	67
b.	Claim 3.....	67
c.	Claim 4.....	68
d.	Claim 5.....	68
e.	Claims 6, 14, and 21	69
f.	Claims 7, 12, and 20	70
g.	Claims 8, 15, and 22	72
h.	Claims 9, 16, and 23	72
3.	Motivation to Combine	73
C.	Ground 2: Claims 1-10, 12, 14-17, and 20-23 Are Obvious Over Mahaffey.....	80
1.	Independent Claim Analysis	81
a.	Claim 1.....	81
b.	Claim 10.....	101
c.	Claim 17.....	107
2.	Dependent Claim Analysis	111
a.	Claim 2.....	111
b.	Claim 3.....	112
c.	Claim 4.....	112
d.	Claim 5.....	113

e.	Claims 6, 14, and 21	114
f.	Claims 7, 12, and 20	115
g.	Claims 8, 15, and 22	116
h.	Claims 9, 16, and 23	116
3.	Motivation to Combine	117
D.	Secondary Considerations of Non-Obviousness.....	121

I, Markus Jakobsson, Ph.D., declare as follows:

I. INTRODUCTION

1. I have been retained by Zimperium, Inc. (“Zimperium”) as an independent expert consultant in this proceeding before the United States Patent and Trademark Office (“PTO”). I am not an employee of Zimperium or any affiliate or subsidiary of Zimperium.

2. I have been asked to consider whether certain references teach or suggest the features recited in the claims of U.S. Patent No. 9,071,518, which I refer to herein as the ’518 patent.

3. My opinions and the bases for my opinions are set forth below.

4. I am being compensated at my standard consulting rate for my time. My compensation is based solely on the amount of time that I devote to activity related to this case and is in no way contingent on the nature of my findings, the presentation of my findings in testimony, or the outcome of this or any other proceeding. I have no other financial interest in this proceeding.

II. EDUCATION BACKGROUND, PROFESSIONAL EXPERIENCE, AND OTHER QUALIFICATIONS

5. A detailed record of my professional qualifications, including a list of professional publications and presentations are set forth in my current Curriculum Vitae, a copy of which is attached as Appendix A. This Curriculum Vitae lists many of the publications that I have authored or co-authored.

6. I received a Ph.D. in Computer Science/Cryptography in 1997 from the University of California San Diego and a M.Sc. in Computer Engineering in 1994 from the Lund Institute of Technology, Sweden. The title of my PhD thesis was “Privacy vs. Authenticity,” and the focus was on secure digital commerce.

7. My professional background is in computer security, mobile security, malware detection, quantitative and qualitative fraud analysis, and development of disruptive security technologies.

8. I am currently Chief Scientist at Artema Labs, a crypto startup. My work there involves identifying new threats and developing new solutions related to crypto payments, non-fungible tokens and wallet technologies.

9. 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.

10. I have been materially involved in the fields of computer and network security, digital rights management, trust establishment, cryptography, socio-technical fraud, malware detection, detection of malicious emails, user interfaces, authentication, and fraud detection for over twenty years.

11. I previously was Principal Scientist at ByteDance Ltd. My work there involved identifying new threats, developing new solutions, and guiding research in areas relating to cryptography and Internet security.

12. Prior to that, I was the Chief of Security and Data Analytics at Amber Solutions, Inc. My work there was to develop and evaluate algorithms to process sensor outputs from distributed and mobile sensor networks, which generate predicates related to the underlying actions of users in the proximity of the sensor networks; to develop and evaluate algorithms to protect the privacy of such users; to develop and evaluate algorithms to protect the security of such users; and to develop and evaluate algorithms that assess the risk of attacks against the system.

13. I have been Chief Scientist at Agari, an email security company addressing Business Email Compromise, and other related forms of abuse.

14. Between 2013 and 2015, I served as a Senior Director at Qualcomm, where my work focused on mobile security systems and mobile malware detection. I joined Qualcomm as a result of them acquiring FatSkunk Inc, a mobile malware startup I co-founded. My work there involved developing remote attestation techniques for use in mobile environments.

15. 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.

16. I have authored or co-authored more than 100 publications including technical papers, textbooks, textbook chapters, and articles, and have delivered numerous presentations including keynotes at several international conferences and workshops, as well as the USPTO. These publications of topics including, but not limited to: crimeware, phishing, electronic identity theft, email security, endpoint security, network security, digital fraud, software-based attestation, remote transaction security and malware detection. My peer reviewed publications have been cited at least 19,854 times, according to Google Scholar. I hold more than 100 patents.

17. Examples of the 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);
- “Mobile Authentication: Problems and Solutions” (Springer, 2012);

- “Understanding Social Engineering Based Scams” (Springer, 2016);
- “Towards Trustworthy Elections: New Directions in Electronic Voting” (Springer, 2010); and
- “Mobile Authentication: Problems and Solutions” (Springer, 2012).

Books translated to Chinese and Korean have separate titles and years.

18. I have authored numerous publications regarding mobile security.

Relevant examples of such publications include:

- “Implicit authentication for mobile devices,” in *The 4th Usenix Workshop on Hot Topics in Security (HotSec’09)*, Montreal, Canada, 2009;
- “Controlling data in the cloud: outsourcing computation without outsourcing control,” in *CCSW ’09: Proceedings of the 2009 ACM Workshop on Cloud Computing Security*, New York, NY, USA, 2009, pp. 85-90;
- “Quantifying security in hybrid cellular networks,” in *Applied Cryptography and Network Security*, New York, NY, 2005;
- “Security weaknesses in Bluetooth,” in *CT-RSA 2001: Proceedings of the 2001 Conference on Topics in Cryptology*, London, UK, 2001, pp. 176-191.

19. I have authored numerous publications regarding user interfaces:

- “Love and authentication,” in CHI ’08: Proceeding of the twenty-sixth annual SigChi conference on Human Factors in Computing Systems, New York, NY, USA, 2008, pp. 197-200; “Bootstrapping mobile PINs using passwords” W2SP, 2011;
- “Designing ethical phishing experiments: a study of (ROT13) rOnl query features,” in WWW ’06: *Proceedings of the 15th International Conference on World Wide Web*, New York, NY, USA, 2006.

20. I have published extensively on software-based attestation, which is a method to protect devices, and especially mobile devices, against malware. Those publications include:

- “Assured detection of malware with applications to mobile platforms,” published in 2010 (co-authored with Karl-Anders Johansson);
- “Retroactive detection of malware with applications to mobile platforms,” in HotSec 2010 (co-authored with Karl-Anders Johansson);
- “Mobile malware: why the traditional AV paradigm is doomed, and how to use physics to detect undesirable routines,” in Blackhat 2013 (co-authored with Guy Stewart).

21. Related patent publications include U.S. Patent Nos. 8,370,935, 8,375,442, 8,544,089, 9,202,051, and 9,665,711. These patents were assigned to

Qualcomm when it acquired FatSkunk, Inc, a startup I co-founded to address the problem of enhancing trust in transactions, especially remote transactions.

22. Much of my other work is also related to applied computer security in general, as described in my CV.

III. ASSIGNMENT AND MATERIALS CONSIDERED

23. I have been asked to provide analysis and explain the subject matter of the '518 patent, including the state of the art when the application underlying the '518 patent was filed. I have also been asked to consider, analyze, and explain certain prior art to the '518 patent including how that art relates to the challenged claims of the '518 patent and to provide my opinions regarding whether that art invalidates the claimed subject matter.

24. The opinions expressed in this declaration are not exhaustive of opinions I may offer in the future regarding the unpatentability of the claims of the '518 patent. Therefore, the fact that I do not address a particular point should not be understood to indicate an agreement on my part that any claim complies with the requirements of any applicable patent or other rule.

25. I reserve the right to amend and supplement this declaration in light of additional evidence, arguments, or testimony presented during this IPR or related proceedings on the '518 patent.

26. In forming the opinions set forth in this declaration, I have considered and relied upon my education, knowledge of the relevant field, knowledge of scientific and engineering principles, and my experience. I have also reviewed and considered the '518 patent (Exhibit 1001), its prosecution history (Exhibit 1002), and the following additional materials:

Exhibit	Description
1004	U.S. Pub 2012/0131685 A1 to Broch et al.
1005	U.S. Pub. 2011/0047597 A1 to Mahaffey et al.
1006	U.S. Pub. 2008/0070495 A1 to Stricklen et al.
1007	U.S. Pub. 2010/0100591 A1 to Mahaffey et al.
1008	U.S. Patent No. 7,519,609 B2 to McConnell
1009	U.S. Patent No. 6,785,673 B1 to Fernandez et al.
1010	U.S. Pub. 2010/0217774 A1 to Marshall
1011	U.S. Patent No. 6,757,706 B1 to Dong et al.
1012	Joint Claim Construction Chart from <i>In re: Taasera Licensing LLC Patent Litigation</i> , Case No. 2:22-MD-03042-JRG (E.D. Tex.)

IV. UNDERSTANDING OF THE LAW

27. I am not an attorney, but have been instructed in and applied the law as described in this section.

28. I understand that the first step in comparing an asserted claim to the prior art is for the claim to be properly construed. I address how a person of ordinary

skill in the art (POSITA) would have understood the claims of the alleged invention in Section VII below.

29. I have been further instructed and understand that a patent claim is unpatentable and invalid as obvious if the subject matter of the claim as a whole would have been obvious to a POSITA of the claimed subject matter as of the time of the invention at issue. I understand that when assessing the obviousness of claimed subject matter, the following factors are evaluated: (1) the scope and content of the prior art; (2) the difference or differences between each claim of the patent and the prior art; and (3) the level of ordinary skill in the art at the time the patent was filed.

30. I understand that claimed subject matter may be obvious in view of more than one item of prior art. I understand, however, that it is not enough to show simply that all the limitations of the claimed subject matter are spread throughout the prior art. Instead, for claimed subject matter to be obvious over multiple references, there must be some reason or motivation for a POSITA to combine the prior art references to arrive at the claimed subject matter.

31. I have been informed that, in seeking to determine whether an invention that is a combination of known elements would have been obvious to a POSITA at the time of the invention, one must consider the references in their entirety to

ascertain whether the disclosures in those references render the combination obvious to such a person.

32. I have been informed and understand that, while not required, the prior art references themselves may provide a teaching, suggestion, motivation, or reason to combine, but other times the motivation linking two or more prior art references is common sense to a POSITA at the time of the invention.

33. I understand that a particular combination may be proven obvious by showing that it was obvious to try the combination. I have been informed that, if a technique has been used to improve one device, and a POSITA would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill.

34. I further understand that an obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references also may be supplied by the direction of the marketplace. For example, when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a POSITA has good reason to pursue the known options within his or her technical grasp because the result is likely the product not of innovation but of ordinary skill and common sense.

35. I have been informed that the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. Thus, where all of the elements of a claim are used in substantially the same manner, in devices in the same field of endeavor, the claim is likely obvious.

36. Additionally, I understand that a patent is likely to be invalid for obviousness if a POSITA can implement a predictable variation or if there existed at the time of the invention a known problem for which there was an obvious solution encompassed by the patent's claims. Therefore, when a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one.

37. I further understand that combining embodiments related to each other in a single prior art reference would not ordinarily require a leap of inventiveness.

38. I also understand that a POSITA must have had a reasonable expectation of success when combining references for claimed subject matter to be obvious.

39. I have been informed and I understand that factors referred to as "objective indicia of non-obviousness" or "secondary considerations" are also to be considered when assessing obviousness when such evidence is available. I understand that these factors can include: (1) commercial success; (2) long-felt but

unresolved needs; (3) copying of the invention by others in the field; (4) initial expressions of disbelief by experts in the field; (5) failure of others to solve the problem the claimed subject matter solved; and (6) unexpected results.

40. I also understand that evidence of objective indicia of non-obviousness must be commensurate in scope with the claimed subject matter. I further understand that there must be a relationship, sometimes referred to as a “nexus,” between any such secondary indicia and the claimed invention.

41. Finally, I have been informed that one cannot use hindsight to determine that an invention was obvious.

42. I provide my opinions in this declaration based on the guidelines set forth above.

V. LEVEL OF SKILL IN THE ART

43. I have been informed and understand that the level of ordinary skill in the relevant art at the time of the invention is relevant to inquiries such as the meaning of claim terms, the meaning of disclosures found in the prior art, and the reasons a POSITA may have for combining references.

44. I have been informed and understand that factors that may be considered in determining the level of ordinary skill include: (1) the education of the inventor; (2) the type of problems encountered in the art; (3) prior art solutions to those problems; (4) rapidity with which innovations are made; (5) sophistication of

the technology; and (6) education level of active workers in the relevant field. I have been further informed and understand that a person of ordinary skill in the art (POSITA) is also a person of ordinary creativity.

45. In my opinion, a POSITA in the field of the '518 patent would have had a degree in computer science, computer engineering, electrical engineering, or a similar discipline, along with 2 years of professional experience working with systems for managing mobile device security, or an equivalent level of skill, knowledge, and experience. In my opinion, this POSITA would have been aware of and generally knowledgeable about the different types of mobile devices available to users and technology used to ensure device security

46. In view of my educational background (including two masters degrees and one doctoral degree) and decades of experience working as a researcher, studying, among other matters: cryptography, Internet security, mobile security, and technical and social solutions to fraud, including phishing, malware and the detection and measuring of anomalous behavior, as discussed in Section II, I was a person of more than the ordinary level of skill in the art as of July 2011. In forming my opinion, I have drawn on my academic background and professional experiences. My opinions herein, however, were formed taking into account the perspective of an ordinarily skilled artisan.

VI. THE '518 PATENT

A. Effective Filing Date

47. I understand that the application leading to the '518 patent was filed on July 2, 2012.

48. Based on my review of the '518 patent, I note that it also references on its face a U.S. provisional application, filed July 1, 2011.

49. For purposes of this declaration, I have been instructed to use July 1, 2011 as the effective filing date of the '518 patent. My opinions in this declaration were formed from the perspective of a POSITA as of July 1, 2011, including both the knowledge of a POSITA at that time as well as how a POSITA would have understood the prior art. My opinions, however, would be the same even if the patent is only entitled to a July 2012 filing date.

B. Overview

50. The '518 patent relates to “methods and systems for ... providing remote management of mobile devices at a server using rules based actions.” (Ex. 1001, 1:15-18.)

51. The patent explains that “[t]he general trend in corporate network management is the addition of mobility of users of the corporate network.” (*Id.*, 1:23-24.)

52. But, mobile devices “may use several different platforms for accessing resources within a network.” (*Id.*, 1:33-38.) This purportedly “make[s] it difficult

to ensure security when users access computing resources within an organization's network[.]” (*Id.*, 1:39-42.)

53. The 518 patent also notes that users “utilize a single device for both business and pleasure.” (*Id.*, 1:42-44.) This also, according to the patent, creates security issues because these “devices include[e] their own personal preferences, data, and applications” that “may be counter to the security or management goals of an organization.” (*Id.*, 1:45-48.)

54. The patent then goes on to discuss a “server-based rules-based action framework.” (*Id.*, 2:6-8.)

55. The server “gather[s] status and configuration information about each of a plurality of mobile devices” employing “different mobile platforms.” (*Id.*, 2:8-12.)

56. Then, “[a]t the server, software processes monitor status information and respond automatically to changes, causing appropriate rules to be evaluated to determine if one is triggered.” (*Id.*, 2:12-15.)

57. The '518 patent explains that these rules are intended to “work across multiple platforms” thereby “simplifying the administrative process.” (*Id.*, 2:16-23.)

58. The rules can also specify an “action” to be taken. The patent provides examples of these actions, like “blocking access ... to a corporate resource” or “wiping” the device.” (*Id.*, 2:36-43.)

59. Examples attributes to be gathered at the server include the device's "operating system," "available ... storage," "battery condition," whether the device is "jailbroken," the device's "password quality," and the device's "policy compliance status." (*See id.*, 20:61-23:40.)

60. The patent also lists many example "rules," such as a "minimum operating system version," "applications compliance," "restrict[ions]" on the device being "jailbroken," and "passcode" requirements. (*Id.*, 19:30-60.)

C. Prosecution History

61. As noted above, I have reviewed the '518 patent's prosecution history.

62. The '518 patent's originally filed claims were rejected in a December 2013 office action. (*See Ex. 1002 at 166.*)

63. Here, in addition to other issues highlighted by the Examiner, the claims were rejected as anticipated by U.S. Pub. 2010/0242088 ("Thomas") or obvious over Thomas in combination with various secondary references. (*Id.* at 170-183.)

64. The Applicant responded with an amendment in May 2014.

65. As part of this, claim 1 was amended to require evaluating mobile device compliance "at the server" using "administrator" defined rules. (*See id.* at 331.)

66. The Applicant then argued that Thomas differs from the claimed subject matter because in Thomas, “enforcement is generally performed locally on clients.” (*Id.* at 337.)

67. The Applicant explained that due to this local enforcement, Thomas purportedly did not teach “that status information about mobile devices is gathered and stored” in a server accessible database and “evaluating, by a server, a compliance state of mobile devices.” (*Id.*)

68. The claims were rejected again in September 2014. (*See id.* at 352.)

69. In this rejection, the Examiner renewed the prior anticipation and obviousness rejections employing Thomas. (*See id.* at 353-372.)

70. Applicant responded with another amendment in February 2015.

71. In this amendment, the Applicant interjected limitations into the claims requiring gathering status information “in a substantially real time manner,” “formatting the status information” to allow rules to apply to devices with “a plurality of different mobile operating systems,” and initiating actions “at the server.” (*Id.* at 390.)

72. Then, the Applicant went on to reiterate its argument that Thomas “does not handle mobile status information at a server in the claimed manner.” (*Id.* at 395.) Instead, the Applicant characterized Thomas as using “client facilities” “to implement and/or participate in the application of ... policies.” (*Id.*)

73. Because of this operation, the Applicant argued that Thomas does not “gather” status information “at a server” as claimed. Instead, data was gathered elsewhere. (*Id.* at 396.) Thomas also did not, according to Applicant, teach gathering device data from a “plurality” of sources. (*Id.*) And, because data is not sent to a server, the Applicant argued that Thomas did not teach “formatting” that data as claimed to allow the server to apply a uniform set of rules. (*Id.*)

74. After this response, the claims were allowed. (*See id.* at 465-470.)

D. Claims

75. For reference, claims 1-4, 6, 8, 10, 14, 15, 17, and 21-22 of the '518 patent are reproduced below. Claims 1, 10, and 17 are independent. The remainder of the claims are dependent:

1. A method for providing device management services comprising steps of:

gathering, at a server, from a network, status information related to an operating state of each of a plurality of mobile devices, each mobile device utilizing one of a plurality of different mobile operating systems, wherein the status information for each mobile device is gathered from a plurality of sources including each mobile device in a substantially real time manner;

formatting the status information for each of the mobile devices, such that rules can be substantially

uniformly applied to the status information across the plurality of different mobile operating systems; storing the status information in an electronic database accessible to the server;

evaluating, at the server, a first compliance state of each of the mobile devices from the status information using a plurality of administrator-defined rules, wherein each of the rules applies to mobile devices regardless of mobile operating system; and

initiating, at the server, at least one action defined by the administrator-defined rules in response to the step of evaluating,

wherein the step of evaluating is performed by the server automatically, in response to changes in the status information in the electronic database.

2. The method of claim 1, wherein the at least one action is selectable from a plurality of actions, comprising blocking access by the at least one of the mobile devices to a corporate resource.

3. The method of claim 1, wherein the at least one action is selectable from a plurality of actions, comprising blocking e-mail access for the at least one of the mobile devices.

4. The method of claim 1, wherein the at least one action is selectable from a plurality of actions, comprising wiping information from the at least one of the mobile devices.

6. The method of claim 1, wherein the step of gathering status information, for the at least one mobile device, comprises gathering information from that mobile device and at least one other source on the network.

8. The method of claim 1, wherein the plurality of different mobile operating systems comprise at least operating systems compliant with Android and iOS specifications.

10. A method for providing device management services comprising steps of:

receiving, at a server, a plurality of attributes related to an operating state of each of a plurality of mobile devices, each mobile device utilizing one of a plurality of different mobile operating systems, wherein the attributes for each mobile device are gathered from a plurality of sources including each mobile device;

making a determination that an attribute of a mobile device has changed;

evaluating the attribute, by a processor at the server in response to the determination, to determine if the attribute change triggers at least one of a plurality of administrator-defined rules, which indicates that the attribute is out of compliance, wherein each of the rules applies to mobile devices regardless of mobile operating system; and

automatically initiating, at the server, an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,

wherein the attribute is stored in a data store accessible to the server, that includes substantially real-time data, gathered from the plurality of sources on a network and pertaining to a plurality of attributes of a plurality of mobile devices, wherein the substantially real-time data is formatted, such that the rules can be substantially uniformly applied to the plurality of attributes across the plurality of different mobile operating systems.

14. The method of claim **10**, further comprising the step of gathering the plurality of attributes for at least one of the plurality of mobile device by gathering attribute information from that mobile device and at least one other source on a network.

15. The method of claim 10, wherein the plurality of different mobile operating systems comprise at least operating systems compliant with Android and iOS specifications.

17. A system for providing device management services comprising:

a data store configured to be accessible by a server;

a CPU of the server coupled to a memory configured to execute computer readable instructions;

computer readable instructions for execution on the server comprising instructions configured to:

automatically make a determination that an attribute of a mobile device stored in the data store has changed;

evaluate the attribute, in response to the determination, to determine if the attribute change triggers at least one of a plurality of administrator-defined rules, which indicates that the attribute is out of compliance, wherein each of the rules applies to mobile devices regardless of mobile operating system; and

automatically initiate an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,

wherein the data store is configured to gather substantially real-time data pertaining to a plurality of attributes related to an operating state of each of the plurality of mobile devices, the attributes for each mobile device being gathered from a plurality of sources including each mobile device, wherein each mobile device utilizes one of a plurality of different mobile operating systems, and the substantially real-time data is formatted, such that the rules can be substantially uniformly applied to the plurality of attributes across the plurality of different mobile operating systems.

21. The system of claim 17, wherein the computer readable instructions further comprise instructions configured to gather the plurality of attributes for at least one of the plurality of mobile device by gathering attribute information from that mobile device and at least one other source on a network.

22. The system of claim 17, wherein the plurality of different mobile operating systems comprise at least operating systems compliant with Android and iOS specifications.

VII. CLAIM CONSTRUCTION

76. I am not a legal expert or an attorney and offer no opinions on the law.

I understand that claim construction is a matter of law. However, I have been

informed by counsel of the legal standards that apply to claim construction in an *inter partes* review, and I have applied them in forming my opinions. I understand that during an *inter partes* review, the same claim construction standard used in federal district court is applied when addressing the meaning of claim terms.

77. I have been informed that the words of a claim are generally given the ordinary and customary meaning that the term or phrase would have to POSITA at the time of the invention in view of the surrounding claim language, the specification and the file history (collectively, the “intrinsic evidence” or “intrinsic record”).

78. I also understand that courts may consider extrinsic evidence, such as expert and inventor testimony, dictionaries, and learned treatises (collectively, the “extrinsic evidence” or the “extrinsic record”), but that such extrinsic evidence should be given less weight than the intrinsic evidence.

79. In my opinion, the terms of the ’518 patent’s claims can be afforded their plain and ordinary meaning for purposes of the analysis of the prior art that I am conducting in this declaration. While I do not believe explicit construction is required, the below paragraphs provide further analysis of certain terms’ plain and ordinary meanings.

A. “Substantially real time” and “Substantially real-time data”

80. Claim 1 of the ’518 patent references “substantially real time” data gathering. Claim 10 similarly references “substantially real-time data.”

81. I understand that in a co-pending district court action involving the '518 patent, Patent Owner proposed that “substantially real time” be construed to mean “without intentional delay, given the processing limitations of the system.” (Ex. 1012 at 168.)

82. I understand that the defendants in the same action argued that this term is indefinite. (*See id.*)

83. For purposes of this declaration, I have been instructed to accept and has applied Patent Owner’s construction.

84. Additionally, based on my review of the '518 patent, while the patent does not define the term “real time,” it does note that it includes data gathering “in response to device attribute changes.” (Ex. 1001, 12:1-2.)

85. In my opinion, a POSITA would have recognized that the plain and ordinary meaning of “substantially real time” embraces at least this: data gathering that occurs immediately when a device attribute changes.

B. “Initiating ... at least one action” and “initiate an action”

86. Claims 1 and 10 both require “initiating ... at least one action.” Claim 17 similarly requires the claimed system to “initiate an action.”

87. I understand that Patent Owner has argued in a co-pending district court action that “initiating ... at least one action” should be afforded its plain and ordinary meaning. (Ex. 1012 at 184.)

88. I also understand that the defendants in this action argued that the term “initiating” means “automatically causing to begin in real time.” (*Id.*)

89. For purposes of this declaration, I have accepted Patent Owner’s proposal.

90. Regardless, in my opinion, even if Defendants’ construction is applied, the prior art would still render the claims obvious. As I explain in greater detail below, both Broch and Mahaffey teach systems where a server automatically triggers responsive action after the completion of an evaluation/analysis.

VIII. UNPATENTABILITY ANALYSIS

91. For the reasons I explain in greater detail below, it is my opinion that as of the effective filing date of the ’518 patent, a POSITA would have considered claims 1-4, 6, 8, 10, 14, 15, 17, and 21-22 of the ’518 patent to be unpatentable as obvious over the prior art.

A. Overview of the Prior Art

92. The following paragraphs provide an overview of the main prior art references I discuss throughout this declaration.

1. Broch

93. Broch indicates on its face that it published May 24, 2012 and was filed November 19, 2010. (*See* Ex. 1004, Cover.) In view of this date of filing, which predates the ’518 patent’s own July 2011 filing date, I understand that Broch is prior art under § 102(e).

94. Broch is not cited by the '518 patent and was not considered during prosecution. (See generally Exs. 1001, 1002.)

95. Broch relates to relates to a “mobile device management system” that “allows network administrators to control the distribution and publication of applications to mobile device users in an enterprise network.” (*Id.*, ¶ [0016].)

96. As shown in Figure 1, which is reproduced below for reference, a “[d]evice management system 102 selectively logs (e.g., tracks and/or stores) data from” multiple “mobile devices 104” via “network 106” “based on the data logging policies associated with each particular mobile device.” (*Id.*, ¶ [0056].)

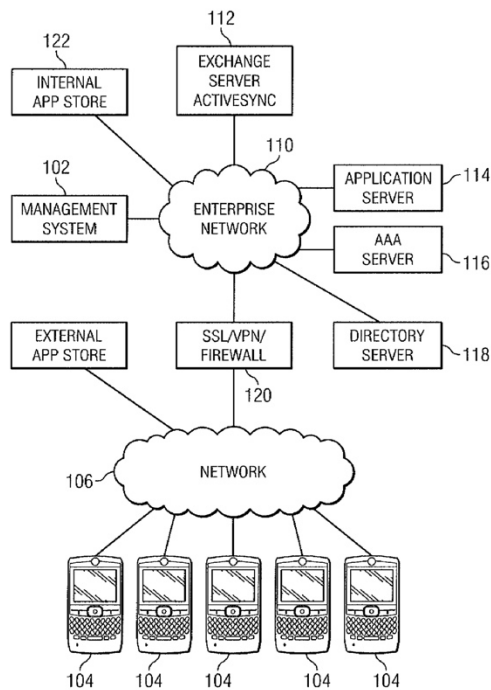


FIG. 1

(*Id.*, Fig. 1.)

97. The devices can run different operating systems. This can include the “Android” or “Windows Mobile operating system[s].” (*Id.*, ¶¶ [0023], [0038].)

98. According to Broch, information gathered from the different devices includes “application usage data,” the identity of “applications installed,” “SIM ... card changes,” “configuration information,” “large file uploads,” “changes to the operating system files or behavior,” or “other activities that may be relevant to a security profile.” (*Id.*, ¶¶ [0038], [0078].)

99. Broch explains that this data can be sent from the devices to the management system 102 in “real time.” (*Id.*, ¶¶ [0043], [0079].)

100. Moreover, Broch explains that the data is “hash[ed]” prior to transmission, and then synchronized upon receipt at the management system 102 with a stored “mobile device profile object.” (*Id.*, ¶¶ [0038], [0047].)

101. Broch’s system also allows “[a] network administrator” to “create a number of policies.” (*Id.*, ¶ [0081].)

102. The management system 102 uses the policies to evaluate the received data and manage the mobile devices. According to Broch, this can entail determining whether a mobile device improperly includes a blacklisted application or potential virus, includes required applications or upgrades, or is configured in a way that makes it unsecure (like employing an unauthorized SIM card or being jailbroken/rooted). (*See id.*, ¶¶ [0078]-[0079], [0081]-[0082].)

103. Based on this evaluation, the server can trigger a response. Example responses include sending a notification, blacklisting a device, or wiping a managed device. (*See id.*, ¶¶ [0081]-[0082].)

104. Figure 11, which is reproduced below, provides a basic example where the system/server receives an application install event from a device, applies a policy to this event, and then triggers a response:

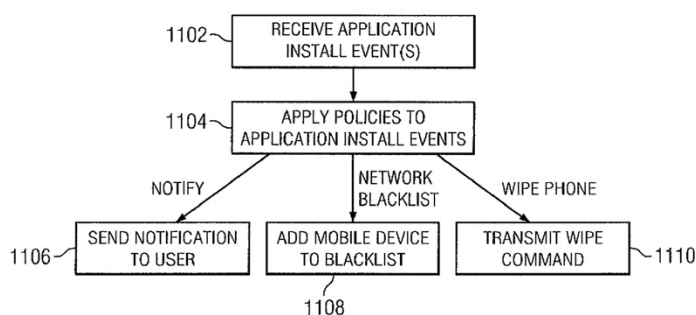


FIG. 11

(*Id.*, Fig. 11, ¶ [0082].)

2. Mahaffey

105. Mahaffey indicates on its face that it published February 2011 and was filed August 25, 2010. (Ex. 1005, Cover.) In view of these dates of publication and filing, both which predate the '518 patent's own July 2011 filing, I understand that Mahaffey is prior art under at least §§ 102(a) and (e).

106. While Mahaffey is not cited by the '518 patent, a related publication is. This publication, however, was not specifically assessed or applied by the Examiner to reject the claims during prosecution. (*See generally* Exs. 1001, 1002.)

107. Mahaffey relates to a system for “detecting and preventing data from adversely affecting ... mobile communication devices.” (Ex. 1005, ¶ [0002].)

108. Mahaffey’s system employs a “server” to monitor, “provide protection from[,] and remov[e] ... undesired applications or other data that may affect” the “devices....” (*Id.*, ¶ [0021].)

109. Multiple “mobile communication devices 101” send data to “servers 151” “over a ... network 121.” (*Id.*, ¶¶ [0025], [0055].) Figure 1, which is reproduced below, shows one such device communicating with a server:

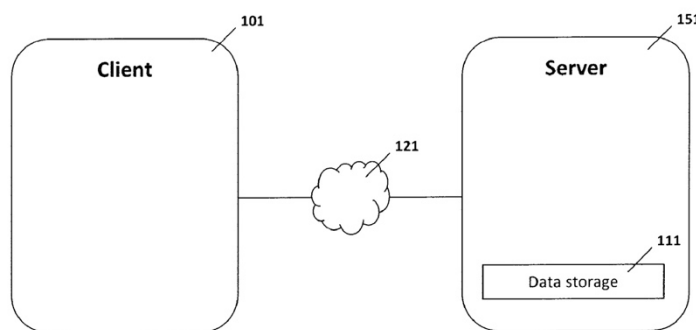


Figure 1

(*Id.*, Fig. 1.)

110. The mobile devices can run various operating systems, like “Apple iOS™” or “Android™ OS....” (*Id.*, ¶ [0022].)

111. Mahaffey explains that the data transmitted by its devices includes “device data” (operating system, firmware, and hardware information) and “application data” (like the presence of malware, software with coding flaws,

software behavioral data, and metadata like file location and characteristics). (*Id.*, ¶¶ [0030]-[0033].)

112. Mahaffey explains that its system employs a “consistent form of information between” mobile device “platforms” to allow the “server 151” to “include[e] some or all of the same fields in assessments” for all managed devices. (*Id.*, ¶ [0167].)

113. In this regard, a “hashing algorithm” can be applied to data before it is sent to a server. (*Id.*, ¶ [0037].)

114. A “cross-platform system” can also facilitate data communication between “a variety of mobile communication device platforms” and the server. (*Id.*, ¶ [0028].)

115. According to Mahaffey, each mobile device can send data to the server “whenever it encounters an undesirable application....” (*Id.*, ¶ [0135].)

116. Next, Mahaffey explains that its system allows an “administrator to set policy” for managed devices. (*Id.*, ¶ [0182].)

117. For example, “[a] blacklist” of “applications ... explicitly denied from running” on the devices can be specified by the administrator. (*Id.*) Additionally, “policies can be set for a specific application, even if the application is available on multiple platforms” (like a device running an Apple, Android, or some other operating system). (*Id.*, ¶ [0199].)

118. Upon receipt of data from the mobile devices, Mahaffey’s “server
151 ... will analyze this data and provide a security related assessment, response
and/or other action” using the policies. (*Id.*, ¶ [0027]; *see also id.*, Fig. 3.)

119. In response to this analysis, the server “notif[ies] the ... device” or takes
other “remediation actions.” (*Id.*, ¶¶ [0040], [0137], [0195].)

3. Stricklen

120. Stricklen indicates on its face that it published March 20, 2008. (*See*
Ex. 1006, Cover.) In view of this date of publication, which predates the ’518
patent’s own July 2011 filing date by more than a year, I understand that Stricklen
is prior art under § 102(b).

121. Stricklen is not cited by the ’518 patent and was not considered during
prosecution. (*See generally* Exs. 1001, 1002.)

122. Stricklen relates to an “an enterprise level device management tool for
various mobile device platforms (e.g., Microsoft™, Symbian™, and RIM™).” (*Id.*,
¶ [0006].)

123. Stricklen explains that in its system, a single server “uniformly
manage[s] a plurality of mobile devices of different types” without “know[ing] the
manner and format or other platform-specific information” of the devices. (*Id.*, ¶
[0062].)

124. Because the system operates in this way, “an administrative user” is only required to “define a management task once for multiple mobile devices 119 of different types....” (*Id.*, ¶ [0071].)

B. Ground 1: Claims 1-10, 12, 14-17, and 20-23 Are Obvious Over Broch and Stricklen

125. In my opinion, a POSITA would have considered Broch and Stricklen to be within the scope of the prior art and highly relevant to the patentability of the ’518 patent’s claims.

126. In my opinion, when viewed and considered in light of the knowledge possessed by a POSITA, Broch and Stricklen together disclose all the limitations of and render claims 1-10, 12, 14-17, and 20-23 of the ’518 patent obvious.

127. In addition, in my opinion, a POSITA have been motivated to combine and collectively consider the teachings of these references and would have had a reasonable expectation of success when doing so.

1. Independent Claim Analysis

a. Claim 1

[1-p] “1. A method for providing device management services comprising steps of:”

128. To the extent this preamble language limiting, in my opinion it is taught by Broch.

129. More particularly, Broch relates to a method and “mobile device management system” that “allows network administrators to control” various

aspects of the devices employed by “mobile device users in an enterprise network.”

(Ex. 1004, ¶¶ [0001], [0016].)

[1-1(a)] “gathering, at a server, from a network, status information related to an operating state of each of a plurality of mobile devices,”

130. This, in my opinion, is taught by Broch.

131. More particularly, Broch’s system gathers **“status information related to an operating state of ... mobile devices.”**

132. The information gathered by Broch’s system includes “application usage data” like “activity data (e.g., data regarding calls, messages, and email), content data (e.g., the text within the messages or email body), and/or context data (e.g., timestamps and location data, etc.)” (Ex. 1004, ¶ [0038].)

133. Other data, like “man-machine interface (MMI) data, file system commands, and ... data characterizing usage of, and/or the actions performed on, the mobile device,” can also be collected. (*See id.*)

134. The system also gathers “data that is relevant to a security profile” of managed devices. (*Id.*, ¶ [0078].) This includes “applications installed,” “configuration information,” “SIM ... card changes,” “large file uploads to a memory card or a remote host,” “changes to the operating system files or behavior, excessive or uncharacteristic out-of-network roaming, or other activities that may be relevant to a security profile.” (*Id.*)

135. In my opinion, a POSITA would have considered all ***“status information related to”*** a device’s ***“operating state.”***

136. Application and device usage data relates to the status and operation of a mobile device. Similarly, device configuration change information—like the SIM card the device is currently using, operating system changes, amount of roaming, and security profile related data—also relates to the status and operation of the device.

137. Next, in my opinion, the data Broch collects from a ***“plurality of mobile devices”*** is ***“gather[ed]”*** at a ***“server”*** over a ***“network.”***

138. Broch explains that in its system, “a plurality of mobile devices 105 ... communicate with device management system 102 via one or more network links 106.” (Ex. 1004, ¶ [0018].)

139. Further, not only is information collected from multiple devices, but the “[m]anagement system 102” can be “one or more computer servers or systems....” (*Id.*, ¶ [0021].)

140. In operation, “[d]evice management system 102 selectively logs (e.g., tracks and/or stores) data from the mobile devices....” (*Id.*, ¶ [0056].) According to Broch, this allows the management system 102 to “maintain an accurate image of the data storage device(s) of the mobile device 104, including the applications installed and the files stored on the mobile device.” (*Id.*, ¶ [0043].) Likewise, Broch

explains that its system operates to “synchronize the state of the mobile device” with the management system 102’s “database” over a “network link.” (*Id.*, ¶ [0017].)

141. An example of this is shown in the below annotated version of Figure 1. As shown, multiple mobile devices 104 (highlighted blue) upload data (highlighted orange) over network 106 to management system 102 (highlighted red). Again, per Broch, the management system 102 can be a server:

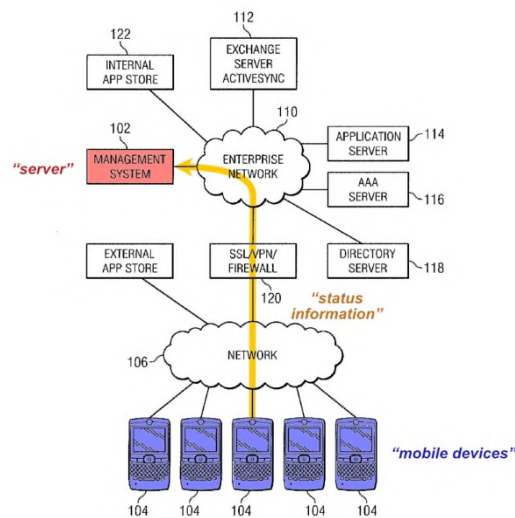


FIG. 1

(*Id.*, Fig. 1 (annotated).)

[1-1(b)] “each mobile device utilizing one of a plurality of different mobile operating systems,”

142. This, in my opinion, is taught by Broch.

143. Broch’s system gathers data from devices running **“a plurality of different mobile operating systems.”**

144. Broch explains that each “mobile device 104” managed by its system can be “a smartphone (e.g., the iPhone manufactured by Apple Inc. of Cupertino,

Calif., the BlackBerry manufactured by Research in Motion (RIM), the G1 based on the Android operating system, or Samsung BlackJack based on the Windows Mobile operating system), tablet (e.g. the iPad manufactured by Apple Inc. of Cupertino Calif.), feature phone, basic cellular phone, personal digital assistant, or other multimedia device.” (Ex. 1004, ¶ [0023].)

145. Similarly, according to Broch, each mobile device’s “operating system may be Windows-based, Mac-based, or Unix or Linux-based, or Symbian-based, among other suitable operating systems.” (*Id.*, ¶ [0038].)

146. Broch also explains that its managing server collects “operating system type” information from mobile devices. (*Id.*, ¶ [0046].) There would be no reason for the system to collect such information if the system was not able to simultaneously monitor and manage devices running different operating systems.

147. In my opinion, Stricklen serves to further confirm that a system like that taught by Broch should obviously be able to accommodate mobile devices running different operating systems.

148. In this regard, Stricklen explains that “mobile devices utilized by employees or associates of an organization vary in their type, supporting platform, or vary in other ways.” (Ex. 1006, ¶ [0003].)

149. Among other things, the devices routinely employ different “operating system.” (*Id.*, ¶ [0069].)

150. A POSITA would have understood that Broch teaches this type of system: it is meant to manage—via a server—many different devices (running different operating systems) used by users associated with an organization or enterprise.

[1-1(c)] “wherein the status information for each mobile device is gathered from a plurality of sources including each mobile device in a substantially real time manner;”

151. This, in my opinion, is taught by Broch.

152. Broch gathers status information ***“from a plurality of sources including each mobile device.”***

153. As I discussed above in connection with limitation [1-1(a)], Broch’s system gathers a variety of status and operating state information from mobile devices.

154. Further, Broch’s system also obtains information from other network sources besides the mobile devices.

155. For instance, Broch explains that the “mobile device management system 102 may include a component that interacts with an ActiveSync/Exchange server 112” to obtain “visibility into all devices attempting to connect to corporate emails.” (Ex. 1004, ¶ [0084].) According to Broch, a “plug-in or other code module ... on ActiveSync/Exchange server 112” “may be configured to transmit information on mobile devices 104 accessing the server.” (*Id.*)

156. As shown by Figure 1, this server 112 is connected to the same network as device management system 102. (*See id.*, Fig. 1.)

157. The server can be provided with other information as well.

158. For example, Broch's management system 102 is also connected to "authentication (AAA) servers 116." (*Id.*, ¶ [0020]; Fig. 1.)

159. Broch explains that the mobile device "profile object" stored and analyzed by its server / system 102 can include "authentication information" relating to mobile device users. (*Id.*, ¶ [0047].)

160. In my opinion, a POSITA would have understood that this information would obviously come (at least in part) from the authentication servers. For example, a POSITA would have recognized that the authentication servers would provide commonly and routinely collect security information such as login attempts and the like.

161. Next, in my opinion, Broch's system also allows for "***substantially real time***" data gathering.

162. According to Broch, a "control client application 308" running on each mobile device is responsible for collecting device data. (*Id.*, ¶ [0038].)

163. Broch goes on to explain that this "application 308" can "operate to provide ... data" to the server "in real-time...." (*Id.*, ¶ [0043].)

164. Similarly, according to Broch “control client 308 can ... transmit state information and event-related data to device management system on a ... as-needed basis” including “when new applications are installed...” (*Id.*, ¶ [0079].) As another example, as I have noted, the system gathers information providing “visibility into all devices attempting to connect to corporate emails.” (Ex. 1004, ¶ [0084].) In other words, the system determines what devices are in the process of doing (and not merely what they have done in the past). Moreover, a POSITA would have recognized that the benefit of Broch’s system would have been dramatically reduced if this type of information was not gathered in real-time, as that would mean that sensitive information (emails) may be provided to a hostile device.

165. In sum, in Broch, data is sent to the management system / server as it is generated and without delay. (*See id.*)

[1-2] “formatting the status information for each of the mobile devices, such that rules can be substantially uniformly applied to the status information across the plurality of different mobile operating systems;”

166. This, in my opinion, is taught by Broch.

167. In particular, Broch’s system “***format[s] the status information***” sent from mobile devices to its management system server.

168. Broch explains that before transferring data from mobile devices to the management system / server, the “control client 308” running on the mobile devices

“creates a hash for the particular file and causes mobile device 104 to transmit the hash to device management system 102.” (Ex. 1004, ¶ [0057].)

169. Then, when device data arrives at the management system server, it is “synchronize[d] ... with a corresponding mobile device profile object stored” by that server. (*Id.*, ¶ [0038].)

170. In my opinion, by synchronizing data with a standardized server object, Broch “**format[s]**” the data as this limit requires. The format of the data has changed: it has been hashed, and then synchronized with the server-stored data object. Further, because the data is stored as a standardized object on the server (as opposed to employing a device-specific data format), Broch’s system allows for the uniform application of rules.

171. Next, in my opinion, Broch employs “**rules**” that are “**applied to the status information**” collected from mobile devices.

172. Broch’s “[d]evice management system 102 can be configured to apply a wide variety of policies.” (*Id.*, ¶ [0082].)

173. According to Broch, these policies “manage[]” the “mobile devices 104” that transmit data to Broch’s system. (*Id.*, ¶ [0058].)

174. Broch provides examples of policies. For instance, the policies may determine if a “newly installed application has been placed on a blacklist” or is a “potential virus.” (*Id.*, ¶ [0082].) They can also determine “whether required

applications (and upgrades thereto) have been installed” by the monitored mobile devices. (*Id.*, ¶ [0078].) Similarly, relayed device “configuration” information can be used to “determine whether any changes have occurred that “indicate a security issue” (such as “a SIM card change” or the device “being jailbroken or rooted”). (*Id.*, ¶ [0079]; *see also id.*, ¶ [0081]-[0082].)

175. In my opinion, a POSITA would also have understood that Broch’s rules are intended to be “***substantially uniformly applied ... across the plurality of different mobile operating systems.***”

176. Again, as I have explained, Broch’s system is meant to monitor and apply policies to a “plurality of mobile devices 105.” (*Id.*, ¶ [0018].) And, these devices can run a variety of different operating systems. (*See id.*, ¶¶ [0023], [0032].)

177. As I have also noted, Broch stores data from mobile devices on its server as a “mobile device profile object.” (*Id.*, ¶ [0038].) This allows for uniform rule application.

178. Moreover, Figure 13 of Broch “illustrates an example application management interface ... that allows a network administrator to configure one or more policies....” (*Id.*, ¶ [0086].)

179. As shown, Broch allows an administrator to designate a policy as applicable to “All-Smartphones” (as opposed to just “Android” or “iPhone” devices):

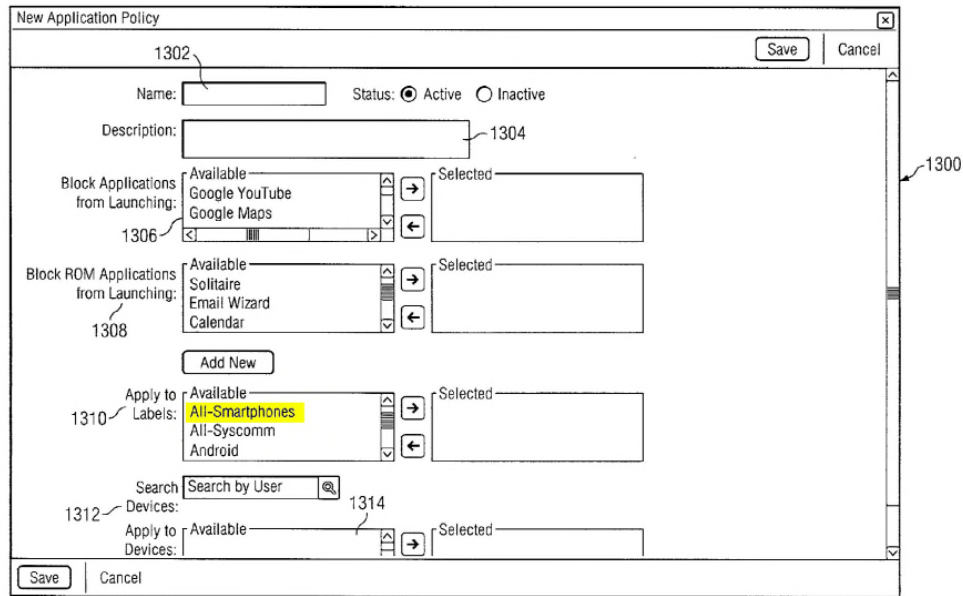


FIG. 13

(*Id.*, Fig. 13 (annotated); ¶ [0092].)

180. Stricklen confirms that server-applied policies like those taught by Broch are designed to be applied uniformly regardless of device operating system (and that data received and stored by Broch’s server is formatted to allow for this).

181. Stricklen explains that it “provides enterprises ... with an enterprise level device management tool for various mobile device platforms (e.g., Microsoft™, Symbian™, and RIM™). (Ex. 1006, ¶ [0006].)

182. According to Stricklen, to allow these different mobile device platforms running different operating systems to communicate with a single server, “agents of modules native to the platforms themselves ... provide interaction with individual mobile devices.” (*Id.*, ¶ [0007].) An “abstraction layer” then allows “management tasks or other functions relating to mobile devices of different types” to be “generally

defined and translated for application to mobile devices supported by different platforms.” (*Id.*, ¶ [0007].)

183. Stricklen explains that due to its system’s use of the native agent and the abstraction layer, the managing server “need not know the manner and format or other platform-specific information to uniformly manage a plurality of mobile devices of different types.” (*Id.*, ¶ [0062].)

184. Put another way, by using intermediate layers that translate communications and transmitted data, a central server can “manage mobile devices of varying types and platforms, such that an administrative user need only define a management task once for multiple mobile devices 119 of different types....” (*Id.*, ¶ [0071].)

185. In my opinion, this is how a POSITA would have understood that this Broch’s system operates similarly.

186. More particularly, Broch data would be processed through an abstraction layer and then synchronized with a stored device profile object that is structured to allow Broch’s management system server to apply a single, unified set of policies to all managed devices.

187. Because device data is formatted, passed through this abstraction layer, and stored on Broch’s server as a standardized profile object rather than in the

device's native format, policies can be applied by the server to all the devices regardless of operating system.

188. Again, this is functionality Broch's system is already intended to provide: Broch teaches that its system can simultaneously collect data from multiple devices running different operating systems. It also teaches that administrators can designate a policy as applicable to all these devices regardless of the operating system running on those devices.

[1-3] “storing the status information in an electronic database accessible to the server;”

189. This, in my opinion, is taught by Broch.

190. Broch's “**server**” (the device management system 102) includes an “**electronic database**” that “**stor[es] the status information**” gathered from devices.

191. Broch explains that “device management system 102 is configured to selectively log data from each of the mobile devices 104 of an enterprise.” (Ex. 1004, ¶ [0040].)

192. Per Broch, the server logged data is stored in a “database within or connected with device management system 102....” (*Id.*, ¶ [0057].)

[1-4] “evaluating, at the server, a first compliance state of each of the mobile devices from the status information using a plurality of administrator-defined rules, wherein each of the rules applies to mobile devices regardless of mobile operating system; and”

193. This, in my opinion, is taught by Broch.

194. To begin, Broch’s system employs **“a plurality of administrator-defined rules.”**

195. Here, Broch explains that “[a] network administrator may ... create a number of policies.” (*Id.*, ¶ [0081].)

196. As I explained above when discussing limitation [1-2], these administrator-created policies can evaluate whether a mobile device improperly includes a blacklisted application or potential virus, includes required applications or upgrades, or is configured in a way (*i.e.*, has an unauthorized SIM card or is jailbroken/rooted) that makes it unsecure. (*See id.*, ¶¶ [0078], [0079], [0081]-[0082].)

197. Moreover, as I have explained, Figure 13 of Broch shows that a policy can be designated as applicable to all devices. (*See id.*, Fig. 13.) Likewise, Stricklen teaches that a system like Broch’s should be designed to allow broad rule applicability regardless of mobile operating system. (*See Ex. 1006*, ¶¶ [0007], [0062], [0071].) In view of this, in my opinion a POSITA would have understood that Broch’s policies are intended to be applied to mobile devices **“regardless of mobile operating system.”** (*See also supra* discussion of limitation [1-2].)

198. Next, in my opinion, Broch’s system uses administrator-defined rules to **“evaluat[e], at the server, a first compliance state of each of the mobile devices from the status information.”**

199. Broch's server engages in analysis: Broch's server (device management system 102) "analyze[s]" "security or device state information" "to detect one or more device state changes." (Ex. 1004, ¶ [0079].)

200. Similarly, Broch explains that its management system / server "analyze[s] the security-related information" provided by mobile devices. (*Id.*)

201. Further, Broch explains that as part of this analysis, the management system / server uses the administrator-defined policies to "identify which applications have been installed on the managed mobile devices 104, including whether required applications (and upgrades thereto) have been installed." (*Id.*, ¶ [0078].)

202. The management system / server also "determine[s] whether any changes have occurred ... that, for example, might indicate a security issue." (*Id.*, ¶ [0079].)

203. Similarly, "[d]evice management system 102 may also determine from the device configuration whether certain functions are enabled or disabled," or whether "the policy configuration of the mobile device 104 is up-to-date, such as minimum acceptable password attributes and the like. ..." (*Id.*)

204. And, the management / system server can evaluate mobile device data to determine "whether a mobile device lacks a required application or fails to have updated policies installed...." (*Id.*, ¶ [0081].)

205. Per Broch, this “analy[sis]” of “security related information” by the management system / server occurs in a “posture-based profile of respective mobile devices.” (*Id.*) In other words, the server uses the administrator defined policies to evaluate the collected device security information.

206. Figure 11, which is reproduced in annotated form below, provides one example of an evaluation performed by server / system 102.

207. As shown, the server first “receive[s]” data from monitored mobile devices indicating “application install event(s).” (*Id.*, Fig. 11.)

208. Then, the “[d]evice management system 102 ... applies one or more policies to the application install event” for purposes of evaluating this received data (yellow highlighting). (*Id.*, ¶ [0082].)

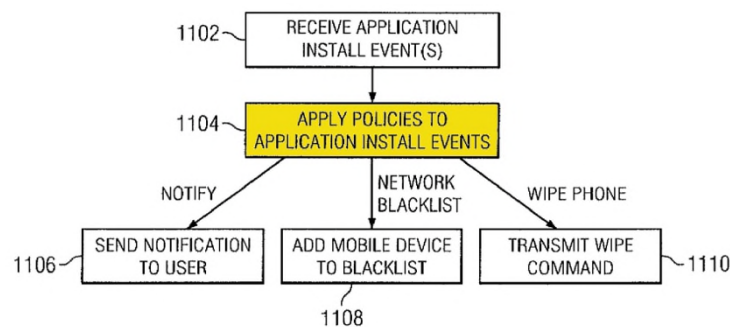


FIG. 11

(*Id.*, Fig. 11 (annotated).)

[1-5] “initiating, at the server, at least one action defined by the administrator-defined rules in response to the step of evaluating,”

209. This, in my opinion, is taught by Broch.

210. Broch’s “**server**” (its device management system) can “***initiate ... at least one action defined by the administrator defined rules***” in response to its evaluation of mobile device data.

211. More particularly, Broch explains that “[a] network administrator can configure policies to respond to application install events in a variety of manners.” (Ex. 1004, ¶ [0082].

212. There are a variety of “possible actions that could be triggered” after analysis. (*Id.*, ¶ [0081].)

213. For example, if evaluation determines that an application has been installed on a device that is “unauthorized, but do[es] not present security issues,” the “mobile device management system 102 may be configured to transmit a notification or warning to the user of the managed mobile device 104....” (*Id.*)

214. Alternatively, if more severe “security issues” are detected like the presence of “a known virus or other malicious application,” “[a]nother policy can be configured [to] cause the mobile device management system 102 to transmit a wipe command” to a mobile device. (*Id.*)

215. As can be seen, it is the mobile device management system 102 / server that initiates these actions: the management system sends a notification or wipe command.

216. Broch also explains that policies can specify other actions to be initiated at the management system / server, such as “lock[ing] down one or more of the input/output devices of the mobile device,” “push[ing] new policy configurations to the mobile device,” or “block[ing] access to” email or other network resources. (*Id.*, ¶¶ [0039], [0082]-[0083].)

217. Figure 11, again reproduced below, provides an example.

218. As shown, after the management system / server “appl[ies] policies to application install events,” it can trigger various actions including “send[ing] notification to user,” “add[ing] mobile device to blacklist,” or “transmit[ting] wipe command” (all highlighted yellow):

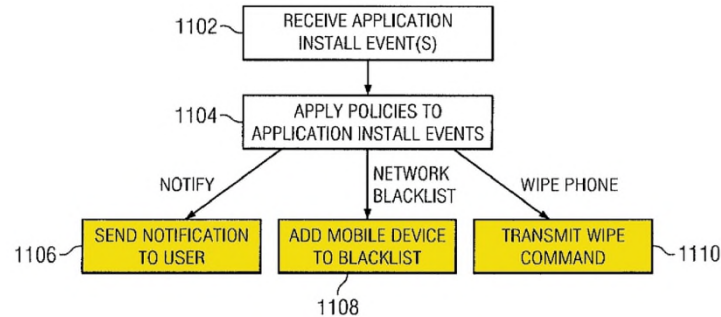


FIG. 11

(*Id.*, Fig. 11 (annotated).)

219. Next, in Broch, the server-initiated actions occur automatically.

220. This is shown in Figure 11: as shown, the responses are automatically triggered by and occur immediately after the policy evaluation performed by the server without the need for manual or other similar intervention. (*See id.*)

221. Consistent with this, Broch explains that server actions are “triggered” by evaluations. (*Id.*, ¶ [0081].) It also explains that policy evaluation causes the server to “respond.” (*Id.*, ¶ [0082].)

[1-6] “wherein the step of evaluating is performed by the server automatically, in response to changes in the status information in the electronic database.”

222. This, in my opinion, is taught by Broch.

223. Again, as explained above in connection with limitation [1-1(a)], Broch teaches that mobile devices can transmit a variety of different types of data to its management system / server for storage and evaluation, including data indicating application installs or security change events.

224. Broch also explains that the evaluation process performed by “device management system 102 ... may be initiated in response to individual events....” (Ex. 1004, ¶ [0082].)

225. Figure 11 provides an overview. As shown, receipt of data from a mobile device at step 1102 results in the performance of evaluation step 1104:

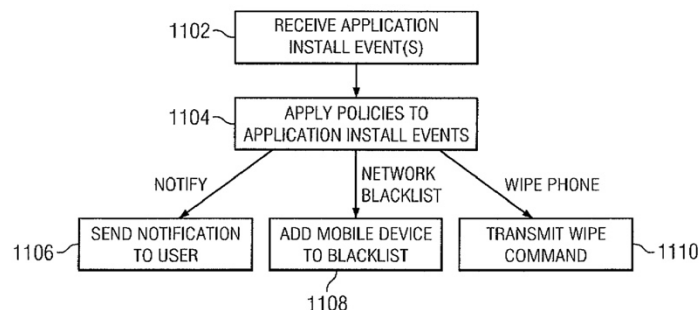


FIG. 11

(*Id.*, Fig. 11.)

226. This, in my opinion, teaches what this limitation requires.

227. Because Broch (1) teaches that evaluation can be “initiated” when events occur and data associated with these events is received and stored at the management system / server, and (2) describes and depicts a process that begins this evaluation immediately upon data receipt without any intervening steps or events, a POSITA would understand that Broch’s server-based evaluation process is ***“performed ... automatically” “in response to changes in the status information in the electronic database.”***

228. Additionally, a POSITA would have recognized that it would make little sense for Broch’s system to engage in “real-time” data gathering (*id.*, ¶ [0043]) if its management system / server did not also immediately and automatically evaluate and recognize any changes in that data upon receipt.

b. Claim 10

229. Claim 10 is another independent claim.

[10-p] “10. A method for providing device management services comprising steps of:”

230. To the extent limiting, this is in my opinion taught by Broch.

231. As discussed in connection with element [1-p] (claim 1’s preamble), Broch teaches a server-based system for managing mobile devices.

[10-1(a)] “receiving, at a server, a plurality of attributes related to an operating state of each of a plurality of mobile devices,”

232. As explained when discussing limitation [1-1(a)], Broch’s system gathers various ***“attributes related to an operating state”*** of mobile devices at its management system / server.

233. This gathered data includes application usage data, device usage data, the identity of installed applications, the update state of applications, configuration information, SIM card change data, data indicating unusual operating system or network behavior, and whether the device is jailbroken/rooted. (*See, e.g.*, Ex. 1004, ¶¶ [0038], [0078]-[0079], [0081]-0082].)

234. In my opinion, all of this constitutes “attributes” relating to device “operating state.” The applications being run and used, a device’s update state, a device’s configuration, the SIM card a device uses, whether or not the device is jailbroken, and the other data discussed above all characterizes the operation and configuration of a device and constitutes a device attribute.

[10-1(b)] “each mobile device utilizing one of a plurality of different mobile operating systems,”

235. This, in my opinion, is taught by Broch.

236. As discussed in connection with limitation [1-1(b)], Broch’s system is designed to gather and analyze data from mobile devices running different operating systems.

[10-1(c)] “wherein the attributes for each mobile device are gathered from a plurality of sources including each mobile device;”

237. This, in my opinion, is taught by Broch.

238. As discussed in connection with limitation [1-1(c)], Broch’s system gathers data from and manages a plurality of mobile devices.

239. Moreover, in addition to gathering data from devices themselves, the system also gathers data regarding the devices from other sources, including an ActiveSync/Exchange server 112 and authentication (AAA) servers 116. (*See supra* discussion of limitation [1-1(c)].)

[10-2] “making a determination that an attribute of a mobile device has changed;”

240. This, in my opinion, is taught by Broch.

241. In particular, Broch “**determin[es]**” that device “**attribute[s]**” have “**changed.**”

242. **According to Broch**, after the “control client 308” running on a mobile device “transmit[s] state information and event-related data to the device management system[,]” the management system “may analyze this data to detect one or more device state changes.” (Ex. 1004, ¶ [0079].)

243. Broch provides examples of change determinations.

244. For instance, the system can detect “that a new application has been installed.” (*Id.*, ¶ [0082].)

245. The system can also determine that “SIM ... card changes” have occurred. (*Id.*, ¶ [0078].)

246. The system can also detect “changes to the operating system files or behavior” and relay these relayed to the management system. (*Id.*; see also *id.*, ¶¶ [0081], [0083] (similarly discussing detected device “changes”).)

[10-3] “evaluating the attribute, by a processor at the server in response to the determination, to determine if the attribute change triggers at least one of a plurality of administrator-defined rules, which indicates that the attribute is out of compliance, wherein each of the rules applies to mobile devices regardless of mobile operating system; and”

247. This, in my opinion, is taught by Broch and Stricklen.

248. As explained when addressing limitation [1-4], Broch and Stricklen together teach a system that employs a ***“plurality of administrator-defined rules”*** that apply to monitored devices ***“regardless of mobile operating system.”***

249. Example policies can prohibit certain applications, require application upgrades, prevent jailbreaking/rooting, or requiring security compliance. Broch’s Figure 13 provides an example of an interface that can be used by an administrator to design a policy. The administrator can specify that the policy applies to all devices regardless of operating system. (See Ex. 1004, Fig. 13.) Stricklen confirms that a system that manages an enterprise’s various devices should allow for a uniform set of rules regardless of device operating system. (See *supra* limitation [1-4].)

250. In my opinion, Broch’s system also ***“determine[s] if the attribute change triggers at least one of a plurality of administrator-defined rules”*** because ***“the attribute is out of compliance.”***

251. In particular, Broch’s management system monitors for and analyzes changes in mobile device data using administrator-specified policies. (*E.g.*, Ex. 1004, ¶¶ [0078]-[0079], [0081]-[0083].)

252. Applying these policies, Broch’s system “manage[s]” mobile devices by determining if the device complies with the policy. (*Id.* ¶ [0058].)

253. According to Broch, this can entail determining if a device attribute change, like an “application install,” runs afoul of a policy. (*Id.*, ¶ [0081]-[0082]; Fig. 11.)

254. Other changes that violate security policies (and thus indicate that the device does not comply with the policy) can also be determined.

255. For instance, Broch can determine if “certain functions are enabled or disabled,” that a “SIM card change” has occurred, that the device has been “jailbroken or rooted,” or that a newly installed application is a “potential virus.” (*Id.*, ¶¶ [0079], [0081]-[0082].)

256. All of these are examples of attributes triggering a policy due to lack of compliance: the device has run afoul of a policy that certain functions be

present/absent, a policy that the SIM card not be changed, a policy that the device not be jailbroken, or the like.

257. As explained when discussing limitation [1-6], Broch teaches that it automatically engages in evaluation of policy compliance using received device data.

[10-4] “automatically initiating, at the server, an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,”

258. This, in my opinion, is taught by Broch.

259. In particular, as discussed in connection with both limitations [1-5] and [1-6], Broch’s system automatically initiates a response at the server in response to the performed analysis. Example responses include a notification, device blacklisting, or wiping.

[10-5(a)] “wherein the attribute is stored in a data store accessible to the server, that includes substantially real-time data, gathered from the plurality of sources on a network and pertaining to a plurality of attributes of a plurality of mobile devices,”

260. This, in my opinion, is taught by Broch.

261. Broch’s system includes the claimed **“data store accessible to the server”**

262. In this regard, and as discussed in connection with limitation [1-3], Broch’s “device management system 102” includes a “database” that stores mobile device data. (Ex. 1004, ¶ [0057].)

263. Further, Broch teaches that stored data is “***substantially real-time data.***”

264. As explained when discussing limitation [1-1(c)], Broch explains that mobile devices send data to Broch’s management system 102 / server “in real time.” (Ex. 1004, ¶ [0043]; *see also id.*, ¶ [0079];

265. Broch also, in my opinion, teaches both “***gather[ing]***” data “***from the plurality of sources***” and data “***pertaining to a plurality of attributes of a plurality of mobile devices.***”

266. As discussed when addressing limitation [1-1(c)], Broch’s system gathers data from many different mobile devices. It also gathers data from various external sources like ActiveSync/Exchange and authentication servers.

[10-5(b)] “wherein the substantially real-time data is formatted, such that the rules can be substantially uniformly applied to the plurality of attributes across the plurality of different mobile operating systems.”

267. This, in my opinion, is taught by Broch and Stricklen.

268. In particular, Broch “***format[s]***” collected device data by hashing and then synchronizing it with a “mobile device profile object” stored on its management system. (Ex. 1004, ¶¶ [0038], [0057].)

269. Broch and Stricklen together teach a system where rules can be “***substantially uniformly applied***” to data collected from devices running “***different mobile operating systems.***”

270. Broch's systems can manage mobile devices running different operating systems. (See *id.*, ¶¶ [0023], [0038].) It also allows rules to be defined that apply to all these devices regardless of operating system. (See, e.g., Fig. 13.)

271. As I explain when discussing limitation [1-2], Stricklen confirms that a system like that taught by Broch should be designed such that rules "manage mobile devices of varying types and platforms." (Ex. 1006, ¶ [0071].)

c. Claim 17

272. Claim 17 is another independent claim.

[17-p] "17. A system for providing device management services comprising:"

273. This, in my opinion, is taught by Broch for the same reasons I discussed for preambles [1-p] and [10-p] of claims 1 and 10.

[17-1] "a data store configured to be accessible by a server;"

274. This, in my opinion, is taught by Broch for the reasons discussed in connection with limitations [1-3] and [10-4(a)]: Broch's management system 102 / server includes a database.

[17-2] "a CPU of the server coupled to a memory configured to execute computer readable instructions;"

[17-3] "computer readable instructions for execution on the server comprising instructions configured to:"

275. This, in my opinion, is taught and rendered obvious by Broch.

276. Broch explains that its "management system 102" may be "one or more computer servers or systems." (Ex. 1004, ¶ [0021].)

277. “FIG. 2 illustrates an example” of one server/computer system. (*Id.*)

278. As shown, the system includes both “[p]rocessors 201” and “memory 203.” (*Id.*, ¶ [0026].)

279. Per Broch, the memory can include “[s]oftware” for “implementing” Broch’s various functions. (*Id.*, ¶¶ [0023], [0026].)

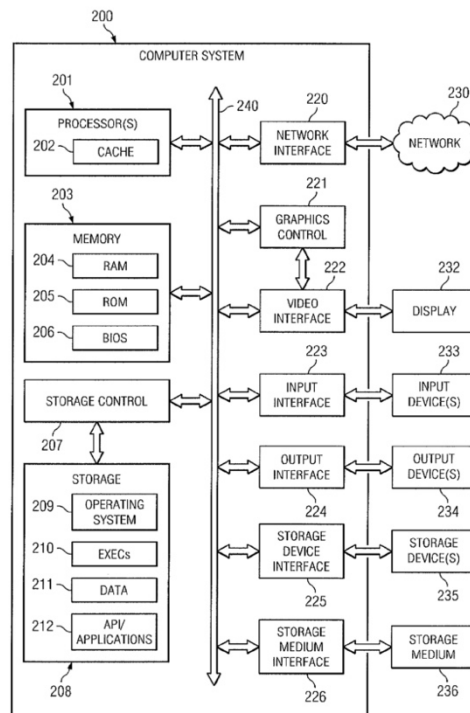


FIG. 2

(*Id.*, Fig. 2.)

[17-3(a)] “automatically make a determination that an attribute of a mobile device stored in the data store has changed;”

280. This, in my opinion, is taught by Broch for the reasons discussed in connection with limitations [1-1(c)], [1-6], and [10-2]: Broch’s system is automated, and monitors for mobile device attribute changes.

[17-3(b)] “evaluate the attribute, in response to the determination, to determine if the attribute change triggers at least one of a plurality of administrator-defined rules, which indicates that the attribute is out of compliance, wherein each of the rules applies to mobile devices regardless of mobile operating system; and”

281. This, in my opinion, is taught by Broch for the reasons explained for limitations [1-3] and [10-4]: Broch’s management system 102 / server evaluates device data using administrator-defined policies to identify, among other things, device policy compliance and violations (such as improper applications, viruses, SIM card changes, jailbreaking, etc.)

[17-3(c)] “automatically initiate an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,”

282. This, in my opinion, is taught by Broch for the reasons I discussed in connection with limitations [1-5] and [1-6]: Broch’s system automatically initiates a responsive action (such as a device notification, device blacklisting, or device wiping) at the device management system / server in response to the analysis performed by that system / server.

[17-3(d)] “wherein the data store is configured to gather substantially real-time data pertaining to a plurality of attributes related to an operating state of each of the plurality of mobile devices, the attributes for each mobile device being gathered from a plurality of sources including each mobile device, wherein each mobile device utilizes one of a plurality of different mobile operating systems, and the substantially real-time data is formatted, such that the rules can be substantially uniformly applied to the plurality of attributes across the plurality of different mobile operating systems.”

283. This, in my opinion, is all taught by Broch.

284. To begin, Broch's system is able to gather data in "real-time" (i.e., "**substantially real-time**" as the claim here requires). (Ex. 1004, ¶¶ [0038], [0043], [0079].)

285. Next, the gathered "**attributes**" come from "**a plurality of sources including each mobile device.**" In particular, as explained when discussing limitation [1-1(c)], Broch collects both device data, and data from other external sources.

286. Broch's system, as explained in more detail when discussing limitation [1-1(b)], is also designed to work with different mobile devices running "**different mobile operating systems.**" (See, e.g., Ex. 1004, ¶¶ [0023], [0038].)

287. Further, in my opinion, and as explained when discussing limitation [1-2], Broch "**format[s]**" data, and Broch and Stricklen both teach systems allowing "**rules**" to "**be substantially uniformly applied.**"

288. Again, as I explained, Broch hashes collected data and synchronizes it with a standardized object stored on its server.

289. As shown by Figure 13, it also allows an administrator to designate policies as applicable to all devices (regardless of operating system).

290. As I also explained, Stricklen likewise teaches that a system (like Broch's) that monitors devices running different operating systems should be designed to allow for uniformly and generally applicable rules and policies to reduce

the burden on the administrator and make management easier. (*See supra* limitation [1-2].)

2. Dependent Claim Analysis

a. Claim 2

291. Claims 2 depends on claims 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising blocking access by the at least one of the mobile devices to a corporate resource”

292. As I explain above, Broch and Stricklen teach everything claim 1 requires. Broch teaches the additional limitation required by this dependent claim.

293. In particular, after evaluating device data, Broch’s server can “trigger[]” various “possible actions.” (Ex. 1004, ¶ [0081].) Figure 11, for example, shows multiple potential responsive actions that can be initiated by a server. (*See id.*, Fig. 11.)

294. Among these possible actions, the server can “[b]lock[] access” to “[o]ne or more network application services...” (*Id.*, ¶ [0081].)

295. According to Broch, this can include “[e]nterprise network applications, such as an email application....” (*Id.*, ¶ [0016].)

b. Claim 3

296. Claims 3 depends on claims 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising blocking e-mail access for the at least one of the mobile devices.”

297. As I explain above, Broch and Stricklen teach everything claim 1 requires. Broch teaches the additional limitation required by this dependent claim.

298. Again, Broch’s server can take various responsive actions after evaluating device data. This includes blocking a device from accessing “network application services” like an “email application.” (Ex. 1004, ¶¶ [0016], [0081]; Fig. 11.)

c. Claim 4

299. Claims 4 depends on claims 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising wiping information from the at least one of the mobile devices.”

300. As I explain above, Broch and Stricklen teach everything claim 1 requires. Broch teaches the additional limitation required by this dependent claim.

301. Broch explains that another of the actions that can be “initiate[d]” by its server is “a wipe action” on a mobile device if it is sufficiently “compromised.” (Ex. 1004, ¶ [0081]; Fig. 11.)

d. Claim 5

302. Claims 5 depends on claims 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising warning a user of the at least one of the mobile devices that further action may be taken.”

303. As I explain above, Broch and Stricklen teach everything claim 1 requires. Broch teaches the additional limitation required by this dependent claim.

304. Again, as discussed above in connection with claims 3 and 4, Broch teaches that its server can initiate a device wipe or block access to network resources.

305. Moreover, Broch’s server can also transmit “warning notifications to the user ... if an unauthorized application is detected on a mobile device.” (Ex. 1004 ¶ [0081]; Fig. 11.)

306. In my opinion, a POSITA would have understood that this “warning” would obviously include a notification of an impending wipe/resource block if the unauthorized application is not deleted. In fact, this is the entire purpose of “warning”: to inform a device user of impending consequences if action (like application removal) is not taken.

e. Claims 6, 14, and 21

307. Claims 6, 14, and 21 respectively depend on claims 1, 10, and 17. They all similarly require:

“gathering [attribute] information from that mobile device and at least one other source on the network”

308. As I explain above, Broch and Stricklen teach everything claims 1, 10 and 17 require. Broch teaches the additional limitations required by these dependent claims

309. As discussed in connection with limitation [1-1(c)], Broch’s system gathers a variety of status information from mobile devices and ***“other source[s] on the network”*** like an ActiveSync/Exchange server 112” and “authentication (AAA) servers 116.”

f. Claims 7, 12, and 20

310. Claims 7, 12, and 20 respectively depend on claims 1, 10, and 17 and further require:

“the step of subsequently evaluating a second compliance state of at least one of the mobile devices to determine if the at least one action should be revoked” (claim 7)

“[the step of]/[wherein the computer readable instructions further comprise instructions configured to] automatically [revoking]/[revoke] the action if the attribute is brought back into compliance” (claims 12, 20)

311. As I explain above, Broch and Stricklen teach everything claims 1, 10 and 17 require. Broch teaches the additional limitations required by these dependent claims

312. Broch’s system “monitor[s] the security state” of mobile devices and blocks policy violating devices from accessing “network application services.” (Ex. 1004, ¶ [0016], [0081]; Fig. 11.)

313. To do so, the management system “signal[s] the security state of a managed mobile device ... to ... enterprise services.” (*Id.*, ¶ [0083]; *see also id.*, Abstract, [0016], [0081].)

314. A device can be both “admit[ted] or den[ied] access.” (*Id.*)

315. Again, Broch evaluates and triggers responsive actions automatically. (*See id.*, ¶ [0082]; Fig. 11; *supra* limitation [1-6].)

316. This, in my opinion, renders what this claim requires obvious. Because Broch continuously monitors device security and provides enterprise services with both deny and admit access signals, a POSITA would have understood that Broch not only denies access when a device initially falls out of compliance, but also re-admits that same device when it is later determined to come back into compliance.

317. Indeed, it would make little sense for Broch’s system to permanently prohibit a device from accessing network resources whenever that device is determined to be non-compliant. Instead, as Broch itself teaches, the access decision would be continuously re-evaluated and the device would be re-allowed access once it again complies with all applicable policies.

g. Claims 8, 15, and 22

318. Claims 8, 15, and 22 respectively depend on claims 1, 10, and 17. They all further requires that:

“the plurality of different mobile operating systems comprise at least operating systems compliant with Android and iOS specifications”

319. As I explain above, Broch and Stricklen teach everything claims 1, 10 and 17 require. Broch teaches the additional limitations required by these dependent claims.

320. More particularly, Broch teaches that its system can monitor and manage mobile devices running different operating systems. This includes “the iPhone manufactured by Apple Inc. of Cupertino, Calif.” (which runs the iOS operating system) and “the G1 based on the Android operating system....” (Ex. 1004, ¶ [0030].)

h. Claims 9, 16, and 23

321. Claims 9, 16, and 23 respectively depend on claims 1, 10, and 17 and further relatedly require:

“wherein the [status information / attribute] comprises location information about [each / the] mobile device”

322. As I explain above, Broch and Stricklen teach everything claims 1, 10 and 17 require. Broch teaches the additional limitation required by these dependent claims.

323. Broch explains that each mobile device can include “Global Positioning System modules.” (Ex. 1004, ¶ [0036].)

324. Because of this, Broch also explains that the “application data” collected by Broch’s system can include “context data” like “timestamps and location data....” (*Id.*, ¶ [0038].)

325. Further confirming that Broch’s server receives device location information, Broch explains that its system allows “locating” a user “device (if GPS enabled).” (*Id.*, ¶ [0049].)

3. Motivation to Combine

326. In my opinion, a POSITA would have considered it obvious and would have been motivated to combine the teachings of Broch and Stricklen to arrive at the subject matter required by the ’518 patent’s claims.

327. As initial matter, in my opinion, a POSITA would have been motivated to implement Broch to employ real-time data collection, to gather various different types of device data, and to allow for a range of different policies.

328. Broch does contemplate that its system can employ different data collection frequencies, including collection on a “periodic basis.” (*E.g.*, Ex. 1004, ¶¶ [0079], [0082].)

329. It also, however, states that its system allows for “real-time” collection “when” events occur on managed mobile devices. (*Id.*, ¶ [0043], [0079].)

330. In my opinion, a POSITA would have understood that the particular frequency of data collection employed by a management system like that taught by Broch entails nothing more than a routine, straightforward balancing of system accuracy and resource consumption. A system that engages in real-time, more frequent collection is more likely to timely identify security issue. But more frequent collection consumes more device, storage, network, and other resources. While a system that engages in less frequent collection will consume fewer resources, it is less likely to timely identify security issues.

331. In my opinion, a POSITA would have been readily able to engage in this balancing, and would have considered the use of real-time data collection to be nothing more than a straightforward design choice.

332. Further, in my opinion, a POSITA would have recognized that certain types of management—such as the detection of malware—would be rendered far less effective if not supported by real-time data gathering. A POSITA would have considered increased resource use resulting from real-time data gathering acceptable given the need to quickly identify malware on managed devices.

333. Moreover, in my opinion, a POSITA would have recognized that others of the types of data Broch collects, such as application installs and SIM card changes, occur less frequently.

334. In my opinion, a POSITA would have recognized that because of the relative infrequency of these events, real-time monitoring would be more practicable and would not impose a significant burden on either the managed devices or the managing server. Moreover, due to the significant security impact of such events, a POSITA would have recognized the importance of a real-time monitoring of such events.

335. Next, in my opinion, a POSITA would also have been motivated to design Broch's system such that it collects and evaluates various different types of device data.

336. Some examples in Broch more limited in scope. For instance, Fig. 11 monitors and assesses "application install event(s)" (Ex. 1004, Fig. 11.)

337. In my opinion, a POSITA would have recognized that Figure 11 (and Broch's other examples) are merely meant to exemplify the operation of its system.

338. Rather than being limited in scope, a POSITA would have recognized that Broch's system is intended to collect and assess a range of status, operation, and attribute information from devices.

339. Consistent with this, Broch itself explains that its system is broadly intended to "allow network administrators" to "manag[e] and control[]" various aspects of "mobile devices in an enterprise environment." (Ex. 1004, ¶¶ [0015]-

[0016].) This includes not only “the deployment of applications,” but also device “security state.” (*Id.*)

340. In my opinion, a POSITA would have understood that collecting more data and allowing greater policy flexibility would further these goals by allowing Broch’s system to detect and address a greater range of potential device security issues.

341. Additionally, by collecting a greater range of data, allowing additional rule flexibility, and allowing a greater range of responsive actions, Broch’s system would be able to be used in a greater range of circumstances and address a broader range of network administrator needs. This would render it more commercially viable, and enable it to detect, classify and address a wider range of security risks.

342. Next, in my opinion, a POSITA would have been motivated to look to Stricklen when implementing Broch.

343. As I have repeatedly explained, Broch teaches a system intended to accommodate and manage devices running different operating systems. (Ex. 1004, ¶¶ [0030], [0032].)

344. It also allows an administrator to designate a policy as applicable to all devices regardless of operating system. (*See, e.g., id.*, Fig. 13.)

345. But, Broch does not fully detail how this functionality is achieved.

346. Stricklen provides additional detail regarding how a single server can communicate and interact with devices running different operating systems.

347. Stricklen explains that its system operates to “enable[] device-independent consolidated asset inventory, configuration management, policy compliance, and security management through a single management application.” (Ex. 1006, ¶ [0006].)

348. To accomplish this, Stricklen’s system employs both an “agent[] or module” that is “native to the platforms themselves” and an “abstraction layer.” (*Id.*, ¶ [0007].)

349. According to Stricklen, the native agent/module allows for “interaction with individual mobile devices” while the abstraction layer allows “management tasks or other functions relating to mobile devices of different types” to be “generally defined and translated...” (*Id.*; *see also id.*, ¶ [0027].)

350. In my opinion, a POSITA would have understood that Stricklen operates just in the way Broch requires.

351. In particular, Stricklen’s native agent/module and abstraction layer allow devices running multiple operating systems to communicate with a single management server.

352. In my opinion, this similarity in purpose and operation would have provided a POSITA with a strong motivation to consider and use Stricklen when

implementing Broch. A POSITA would have recognized that Stricklen provides a straightforward means for achieving the functionality taught by Broch: the various devices managed by Broch's management system / server would be able to communicate with that server regardless of what operating system they run.

353. In my opinion, a POSITA also would have recognized that Stricklen's system provides numerous other benefits and advantages. This also would have motivated its application to Broch.

354. For example, Stricklen explains that its system allows for performance of an initial "[b]ootstrapping" process. (Ex. 1006, ¶ [0059].)

355. More specifically, administrators can "push[] an initial policy or set of instructions to a mobile device" associated with an enterprise. (*Id.*)

356. In my opinion, a POSITA would have recognized that this would have improved Broch's system. Not only would Broch's management system be able to monitor and manage mobile devices over time, but it would be able to dictate their initial configuration via the bootstrapping process. This, in turn, would have both improved the system's security (by dictating initial, more secure device configurations) and ensured a greater uniformity between mobile device configurations easing later management (as only deviations from the original configuration would need to be monitored). Moreover, a protection of a device starting from the initial configuration helps address the problem of potential device

corruption during the setup process performed by the user as he or she sets up the device, e.g., connecting the device to potentially untrusted WiFi networks.

357. Next, Stricklen explains that its can be integrated with an “existing organizational structure” like an “enterprise directory.” (Ex. 1006, ¶ [0016].)

358. In my opinion, a POSITA would have understood that this would also have improved Broch. Existing enterprise information would be leveraged to simplify the process of defining management policies. For instance, using directory/organizational information, policies would be tailored to apply to devices owned by particular parts (as opposed to all) of an organization.

359. Stricklen also explains that its system allows for the use of various “standards” compliant “components.” (Ex. 1006, ¶ [0006].)

360. In my opinion, a POSITA would have recognized that designing Broch’s system to be similarly standard compliant would have simplified its design and rendered it more compatible with a wider range of devices and networks. Moreover, using standard complaint components would allow Broch’s system to be more readily repaired and upgraded (as there would be a greater base of information and existing knowledge regarding how the system’s components function). In the context of security, it is commonly understood that use of standardized components (such as standardized encryption algorithms) strengthens security, both by the additional scrutiny of the algorithms and their implementations.

361. In my opinion, a POSITA would also have had a more than reasonable expectation of success.

362. As I have explained, Broch is already open to and designed to allow for real-time data gathering, the collection of many different types of device data, and the application of different policies.

363. Given this system flexibility, POSITA would have a more than reasonable expectation that Broch's system would be successfully implemented to include this collection of features. Indeed, a POSITA would have recognized that Broch's system is designed to do so.

364. Moreover, given the similarities between Broch's and Stricklen's systems—both are meant to monitor and manage devices running different operating systems—in my opinion, a POSITA would have every reason to believe and no reason to doubt that the two references' systems would be successfully combined.

C. Ground 2: Claims 1-10, 12, 14-17, and 20-23 Are Obvious Over Mahaffey

365. In my opinion, a POSITA would have considered Mahaffey to be within the scope of the prior art and highly relevant to the patentability of the '518 patent's claims.

366. In my opinion, when viewed and considered in light of the knowledge possessed by a POSITA, Mahaffey discloses all the limitations of and renders claims 1-10, 12, 14-17, and 20-23 of the '518 patent obvious.

367. In addition, in my opinion, a POSITA have been motivated to combine and collectively consider the teachings of Mahaffey and would have had a reasonable expectation of success when doing so.

1. Independent Claim Analysis

a. Claim 1

[1-p] “1. A method for providing device management services comprising steps of:”

368. To the extent this preamble language is limiting, Mahaffey teaches the claimed method for providing **“device management services.”**

369. Mahaffey explains that it is directed to a system and method for “detecting and preventing data from adversely affecting ... mobile communication devices.” (Ex. 1005, ¶ [0002].)

370. A “server” “provide[s] protection from and removal of undesired applications or other data that may affect” the “devices....” (*Id.*, ¶ [0021].)

[1-1(a)] “gathering, at a server, from a network, status information related to an operating state of each of a plurality of mobile devices,”

371. This, in my opinion, is taught by Mahaffey.

372. In particular, Mahaffey’s system **“gather[s] ... status information related to an operating state ... of mobile devices.”**

373. Here, Mahaffey explains that its system operates to collect both “device data” and “application data.” (Ex. 1005, ¶¶ [0030]-[0031].)

374. Further, per Mahaffey, the “device data” can be “linked to the application data” to later allow “assessments, categorizations, and characterizations” to take “into account the source of the data.” (*Id.*, ¶ [0076].)

375. Mahaffey elaborates explaining that “device data” “pertains to the state capabilities, operating system, firmware version, memory capacity, available communication ports, battery limitations, hardware characteristics and other ‘baseline’ information” of a particular “mobile communication device.” (*Id.*, ¶ [0030].)

376. And, “[a]pplication data includes data objects that are malware or spyware, and thereby can negatively affect a mobile communication device.” (*Id.*, ¶ [0031].)

377. “Application data” can also include “data objects” present on the mobile device that “have coding flaws or other issues that can negatively affect the device,” “compromise[] privacy, overtax[] a device’s battery or network connection, and/or has objectional content.” (*Id.*, ¶ [0032].)

378. Further, “application data” can include “metadata” like “the location” in a “device’s filesystem where a data object is stored” and other “characteristics of how the data object is installed on or integrates the mobile communication device’s operating system.” (*Id.*, ¶ [0033].) “[A]pplication data also includes behavioral data” like “how an application interacts with or uses” device “resources, such as

memory usage, battery usage, storage usage, CPU usages, API usage, errors and crashes, network services connected to ..., and runtime library linkage.” (*Id.*, ¶ [0034].)

379. In my opinion, this data is “***status information***” relating to mobile device “***operating state.***” In particular, the device and application data gathered and transmitted by Mahaffey together indicate applications running on the device, the impact those applications have on device operation and behavior, and the device’s various hardware characteristics.

380. Next, in my opinion, in Mahaffey information from “***each of a plurality of mobile devices***” is gathered “***at a server.***”

381. “FIG. 1” of Mahaffey, which is reproduced and annotated below, shows “mobile communication devices 101” (blue highlighting) communicating with “servers 151” (red highlighting) “over a cellular, wireless Internet or other network 121.” (Ex. 1005, ¶ [0025].)

382. Mahaffey explains that each “server 151” “may have access to a plurality of mobile communication devices....” (*Id.*, ¶ [0055].)

383. Further, each of the mobile devices “send[] application data” (orange highlighting) “to server 151....” (*Id.*, ¶ [0039].) Thus, the server engages in the “gather[ing]” of “device data” from the devices. (*Id.*, ¶¶ [0075]-[0076].)

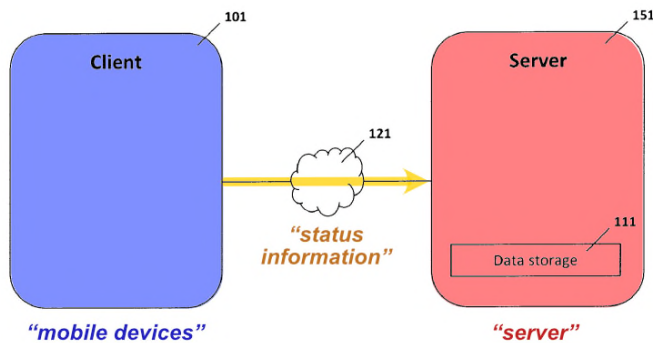


Figure 1

(Id., Fig. 1 (annotated).)

384. The gathering and storage of device data at the server is also shown by steps 305 and 307 of Figure 3, reproduced and annotated below:

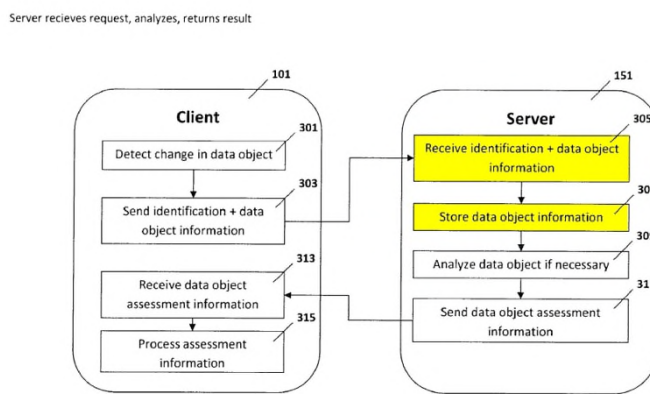


Figure 3

(Id., Fig. 3 (annotated).)

[1-1(b)] “each mobile device utilizing one of a plurality of different mobile operating systems,”

385. This, in my opinion, is taught by Mahaffey.

386. In particular, Mahaffey teaches that the “mobile communication devices 101” managed by its system employ **“a plurality of different mobile operating systems.”**

387. In particular, the managed devices can be “any computer or computing device running an operating system for use on handheld mobile devices, such as smartphones, PDAs, tablets, mobile phones and the like.” (Ex. 1005; ¶ [0022].)

388. This can include “the Apple iPhone®, the Apple iPad®, the Palm Pre, or any device running the Apple iOS™, Android™ OS, Google Chrome OS, Symbian OS®, Windows Mobile® OS, Pam OS® or Palm Web OSTM.” (*Id.*)

389. Moreover, Mahaffey explains that its system allows an “IT administrator to manage a fleet of mobile communication devices running different operating systems.” (*Id.*, ¶ [0199]; *see also* Fig. 11.)

[1-1(c)] “wherein the status information for each mobile device is gathered from a plurality of sources including each mobile device in a substantially real time manner;”

390. This, in my opinion, is taught by Mahaffey.

391. In particular, Mahaffey gathers status information **“from a plurality of sources including each mobile device.”**

392. Again, as I explained when discussing limitation [1-1(a)], Mahaffey’s system gathers a variety of status information from mobile devices.

393. Further, Mahaffey also gathers device information from other network sources besides the mobile devices themselves.

394. For instance, “server 151” can “analyze[] user-generated ratings and comments for an application, such as those gathered by application marketplace data gathering system 1005.” (Ex. 1005, ¶ [0095].)

395. Similarly, Mahaffey explains that in addition to application/device data, “multiple” other “data sources” are “available to server 151” such as “characterization data, trust data, distribution data, and user supplied data to determine if an application is malicious.” (*Id.*, ¶ [0100].)

396. Figure 10 of Mahaffey provides yet another example, depicting the “[g]athering [o]f data from disparate data sources” including the mobile devices 101, the application marketplace data gathering system 1005 and various other sources. (*Id.*, Fig. 10.)

397. Next, in my opinion, Mahaffey’s system allows device “status information” to be gathered in “***a substantially real time manner.***”

398. In particular, Mahaffey explains that the system is “continuously updated to provide constant real-time protection” of devices. (Ex. 1005, ¶ [0004].) Real-time protection can only be provided as Mahaffey teaches if data is gathered from devices in real-time as it is generated.

399. Consistent with this, Mahaffey explains that each “mobile communication device 101” can “send[] events to server 151” “whenever it encounters an undesirable application....” (*Id.*, ¶ [0135].) Likewise, it explains that its system allows for customization of “how frequently” mobile devices “gather and transmit behavioral data.” (*Id.*, ¶ [0060].) Thus, per Mahaffey, the rate at which data is transmitted can be controlled. And, data can be sent at the time a tracked event occurs. In other words, data can be transmitted in a non-delayed fashion as it is generated.

400. Moreover, Mahaffey also teaches that its system “immediately” analyzes received data. (*Id.*, ¶ [0110].) This, in my opinion, confirms that Mahaffey’s system engages in the real-time gathering of data. It would make little sense for the system to “immediately” analyze received data if that data were not also collected as it is generated in real-time. Put another way, immediate data analysis would be of little use if the data underlying that analysis is old and out-of-date.

[1-2] “formatting the status information for each of the mobile devices, such that rules can be substantially uniformly applied to the status information across the plurality of different mobile operating systems;”

401. This, in my opinion, is taught by Mahaffey.

402. To begin, Mahaffey teaches that its system employs “***rules***” that are “***applied to the status information***” collected from mobile devices.

403. Here, Mahaffey explains that its system allows an “administrator to set policy [regarding] what applications are desirable to install on a device or group of devices.” (Ex. 1005, ¶ [0182].)

404. Mahaffey provides various examples of administrator-specified policies that are to be used to assess the application and other data received from mobile devices.

405. For instance, “[a] blacklist” of “applications ... explicitly denied from running on a mobile communications device” can be specified. (*Id.*)

406. Additionally, “a policy may specify certain capabilities (e.g., sending a device’s location to the internet) that are blacklisted....” (*Id.*)

407. Next, in my opinion, Mahaffey teaches that its system’s rules are intended to be “***substantially uniformly applied ... across the plurality of different mobile operating systems.***”

408. Mahaffey explains that its system is designed to allow for mobile device protection “regardless of the make or model of the mobile communication device(s).” (*Id.*, ¶ [0021].)

409. The system can perform a “cross-platform assessment” applicable to different systems, like “BlackBerry” and “Android....” (*Id.*, ¶ [0167].)

410. Because of this, “policies can be set for a specific application, even if the application is available on multiple platforms and has multiple versions.” (*Id.*, ¶

[0199].) “[I]f a certain application is known to be undesirable on” different “operating systems” like those employed by “iPhones, BlackBerry devices and Android devices,” “then the IT administrator can block all versions of the application from installation, regardless of platform.” (*Id.*)

411. To accomplish this ability to gather and then assess data received from mobile devices employing different platforms, Mahaffey teaches that the data transmitted from devices to the server is “*formatt[ed]*.”

412. Mahaffey explains that its system operates to “provide a consistent form of information between platforms[,]” so as to allow the “server 151” to “include[e] some or all of the same fields in assessments for data objects that run on different platforms....” (*Id.*, ¶ [0167].)

413. In my opinion, this is what this limitation requires. Data from different devices running different operating systems is consistently formatted—or, as Mahaffey says, is gathered at the server in “a consistent form”—to allow Mahaffey’s server to analyze the data.

414. Next, Mahaffey also provides examples of how this consistent information form is achieved.

415. Mahaffey notes that a “hashing algorithm” can be employed to “hash” data before transmission to the server. (*Id.*, ¶ [0037].)

416. And, a “cross-platform system” can be employed to allow for use of the system “on a variety of mobile communication device platforms.” (*Id.*, ¶ [0028].)

417. Mahaffey points to (and incorporates by reference) “U.S. patent application Ser. No. 12/255,626” as providing further detail of a “cross-platform system” that can be used to facilitate the management of mobile devices running different operating systems. (*See id.*)

418. This application published as the ’591 application. (*See* Ex. 1007.)

419. The ’591 application explains that it provides a “cross-platform system” that operates “on any mobile communication device, regardless of manufacture or service provider.” (*Id.*, ¶ [0005].)

420. The system includes “a platform-independent or ‘core’ component, a platform specific component, and a lightweight abstraction layer component....” (*Id.*, ¶ [0021]; *see also id.*, Fig. 1.)

421. A “common API” “enable[s] communication” between different components “regardless of manufacture, service provider or operating system.” (*Id.*, ¶ [0024].) In other words, “[t]he common API ... facilitate[s] communication and allow[s] access or calling within the system.” (*Id.*, ¶ [0030].)

422. Among other functionality, the “common API” includes an “XML parser or generator.” (*Id.*, ¶ [0030].)

423. In sum, in my opinion, due to the incorporation of the '591 application a POSITA would have understood Mahaffey to teach the use of a cross-platform system that translates platform-specific data into a platform-independent data format (such as XML) using platform specific and platform independent components and an API.

424. A POSITA would also have recognized that XML is a data format that is not platform specific, and can be generated and used by systems running different operating systems.

425. In particular, it was well known at the time the '518 patent was filed that an “XML storage solution allows data from any arbitrary relational database schema to be loaded, rearranged and retrieved by a data aggregator.” (*E.g.*, Ex. 1008, 7:12-15.) In other words, XML allows data from various different sources to be aggregated together.

426. Likewise, it was known that “XML” serves “as a ‘lingua franca’ ... mak[ing] it possible for data to be exchanged regardless of the platform on which it is stored or the data model in which it is represented.” (*E.g.*, Ex. 1009, 1:38-43.) Thus, by formatting data as XML, data can be exchanged between a server and devices running different operating systems.

[1-3] “storing the status information in an electronic database accessible to the server;”

427. This, in my opinion, is taught by Mahaffey.

428. In particular, Mahaffey explains that its “server[] 151” has an internal “data storage 111” (in the form of a “database, data table, data structure, file system or other memory store”) for storing the information gathered from managed mobile devices. (Ex. 1005, ¶ [0025].)

429. This is shown in the below annotated version of Figure 1. For reference, data storage 111 (the claimed “*electronic database*”) in Mahaffey’s server 151 is highlighted dark red below:

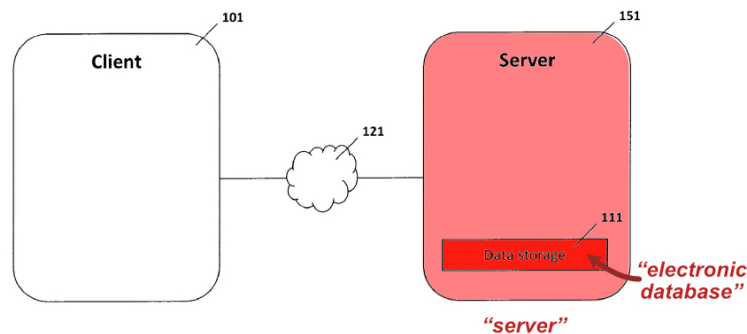


Figure 1

(*Id.*, Fig. 1 (annotated).)

430. Next, in my opinion, Mahaffey stores “*the status information*” its system previously gathered from mobile devices in this data storage 111.

431. In particular, as shown in Figure 11 below, “[i]n block 1103, application data ... is stored on server 151 or data storage 111” while gathered “device data” is similarly stored at step 1107. (*Id.*, ¶¶ [0075]-[0076].)

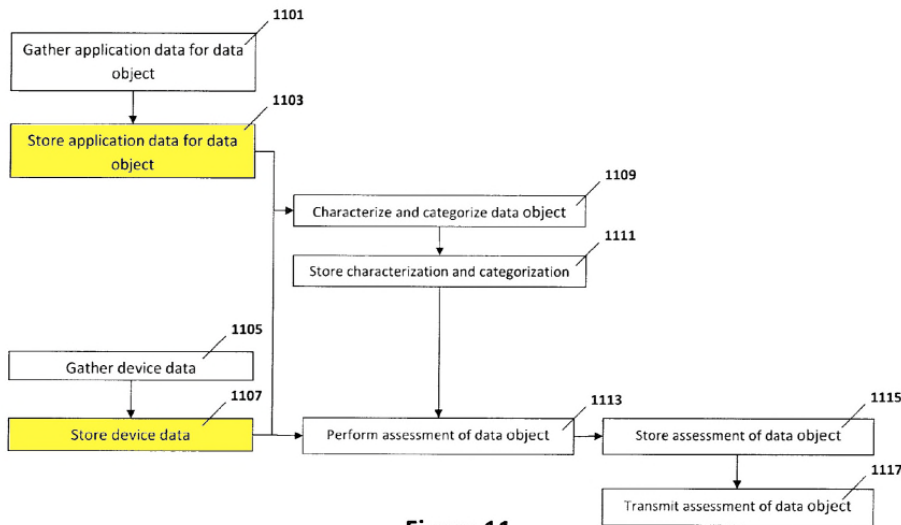


Figure 11

(Id., Fig. 11 (annotated).)

432. The step of storing data from mobile devices is also shown in Figure 3, which is reproduced in annotated form below:

Server receives request, analyzes, returns result

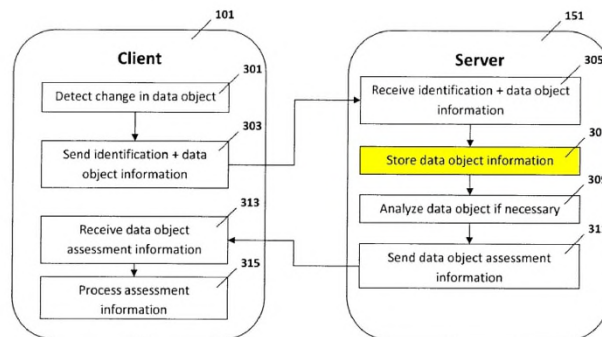


Figure 3

(Id., Fig. 3 (annotated).)

[1-4] “evaluating, at the server, a first compliance state of each of the mobile devices from the status information using a plurality of administrator-defined rules, wherein each of the rules applies to mobile devices regardless of mobile operating system; and”

433. This, in my opinion, is taught by Mahaffey.

434. In particular, Mahaffey’s system employs **“a plurality of administrator-defined rules”** that are used to assess data from mobile devices.

435. For instance, Mahaffey explains that its “server 151 allows a user or administrator to set policy for a device or group of devices.” (Ex. 1005, ¶ [0186].) Thus, the system provides for “administrator ... set policy.” (*Id.*, ¶ [0182].)

436. “[A]n administrator may use server 151 to block all applications that are in a certain category such as social networking applications or all applications that access certain capabilities such as the ability to transmit files or other sensitive data from a device.” (*Id.*, ¶ [0187].)

437. “[S]erver 151” also “allows policy to be configured by assessment criteria....” (*Id.*)

438. Then, “[w]hen a device 101 attempts to install an application, the device sends a request to server 151 for assessment of the application. Based on policy configured on server 151, the assessment contains an indication of whether the application is allowed or disallowed....” (*Id.*, ¶ [0186].)

439. In other words, Mahaffey’s server uses the administrator-specified policies to evaluate the formatted data it receives from mobile devices.

440. Next, in my opinion, Mahaffey explains that its administrator-defined and server applied rules apply ***“regardless of mobile operating system.”***

441. Mahaffey explains that its policies allow for a “cross-platform assessment” applicable to different systems, like “BlackBerry” and “Android....” (*Id.*, ¶ [0167].)

442. Thus, “policies can be set for a specific application, even if the application is available on multiple platforms and has multiple versions.” (*Id.*, ¶ [0199].) Because of this, if a policy blocks specific applications, these applications can be blocked even if running on different “operating systems,” like those employed by “iPhones, BlackBerry devices and Android devices[.]” (*Id.*)

443. Finally, in operation, Mahaffey’s system employs these rules to ***“evaluat[e], at the server, a first compliance state of each of the mobile devices from the status information.”***

444. Again, Mahaffey explains that after each “mobile communication device 101 ... transmit[s] certain data to server 151,” the server “will analyze this data and provide a security related assessment, response and/or other action.” (*Id.*, ¶ [0027].)

445. “[B]oth device data and application data are useful for providing” this assessment. (*Id.*, ¶ [0035].)

446. The data and rules are used by the server to perform a variety of assessments: the “server may use a variety of methods of analysis.” (*Id.*, ¶¶ [0075], [0153].)

447. For example, Mahaffey explains that the server can determine if a “data object” running on a device is a “battery waster[.]” (*Id.*, ¶ [0086].)

448. Further, the server may also “perform analysis” to “determine if” a device “application is considered to be malware of spyware...” (*Id.*, ¶ [0106].)

449. Device “policy compliance checking can be performed by ... server 151....” (*Id.*, ¶ [0191].)

450. In my opinion, a POSITA would have understood this policy checking to include an assessment of whether a mobile device includes applications (or application functionality) prohibited by the administrator defined blacklists or other policies.

451. The below annotated version of Figure 11 shows performance of assessment (which again can include an assessment of whether an application is blacklisted, whether the application includes prohibited functionality, whether the application is malware/spyware, or a determination of policy compliance) by the server at step 1113:

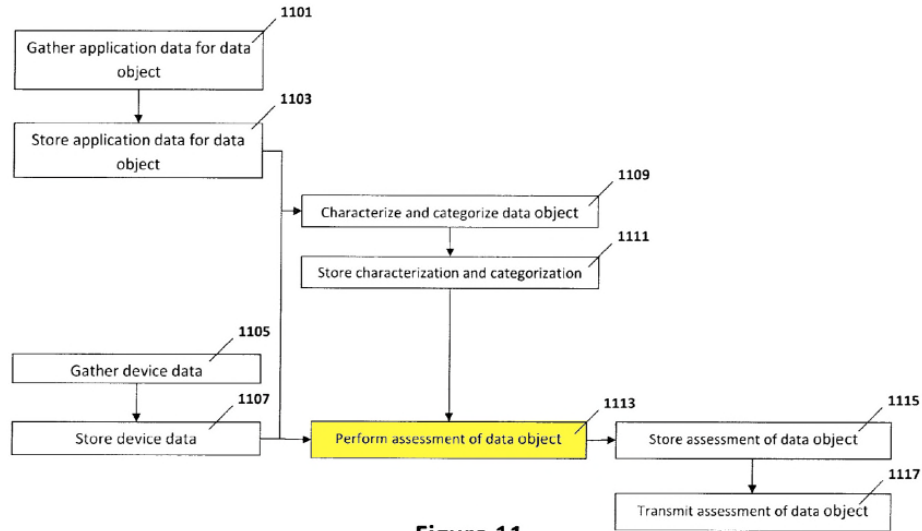


Figure 11

(Id., Fig. 11 (annotated).)

452. Performance of an evaluation/analysis by the server is also shown in step 109 of Figure 3 below:

Server receives request, analyzes, returns result

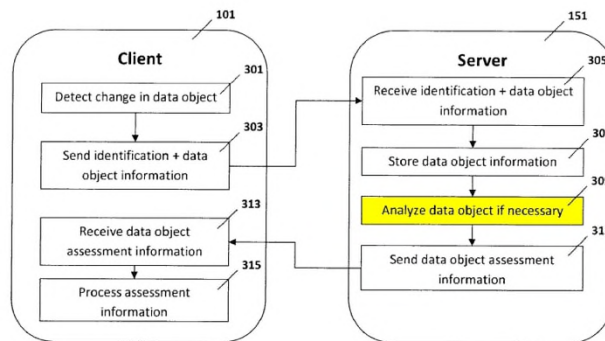


Figure 3

(Id., Fig. 3 (annotated).)

[1-5] “initiating, at the server, at least one action defined by the administrator-defined rules in response to the step of evaluating,”

453. This, in my opinion, is taught by Mahaffey.

454. In particular, after performing its evaluation, Mahaffey’s server responsively ***“initiat[es] ... at least one action defined by the administrator-defined rules.”***

455. For example, if the evaluation performed by Mahaffey server determines that there is “malicious” data on a mobile device, the server “may notify the ... device or instruct the device to reject the data object and not process it further.” (Ex. 1005, ¶ [0137].)

456. Similarly, if the server determines that a “device is ... violating policy, a variety of remediation actions are possible”: “any violating applications may have their processes ended, may be uninstalled or isolated from accessing certain system functionality (e.g., internet, private data), or may be restricted from accessing certain networks (e.g., only allowed to access Wi-Fi, not the cellular network). It may also be desirable to isolate the whole device from accessing sensitive resources such as a corporate email or VPN server while it is out of compliance to prevent information leakage.” (*Id.*, ¶ [0195].)

457. The “server 151” can “send[] a notification to mobile communication device 101” in the form of “an alert, a message, an instruction or other information” *Id.*, ¶ [0040]; *see also id.*, ¶ [0048] (“server 151 may transmit a notification, remediation instruction or the like”).)

458. This responsive, server-performed action after evaluation of device data is shown as step 1117 in Figure 11 reproduced below:

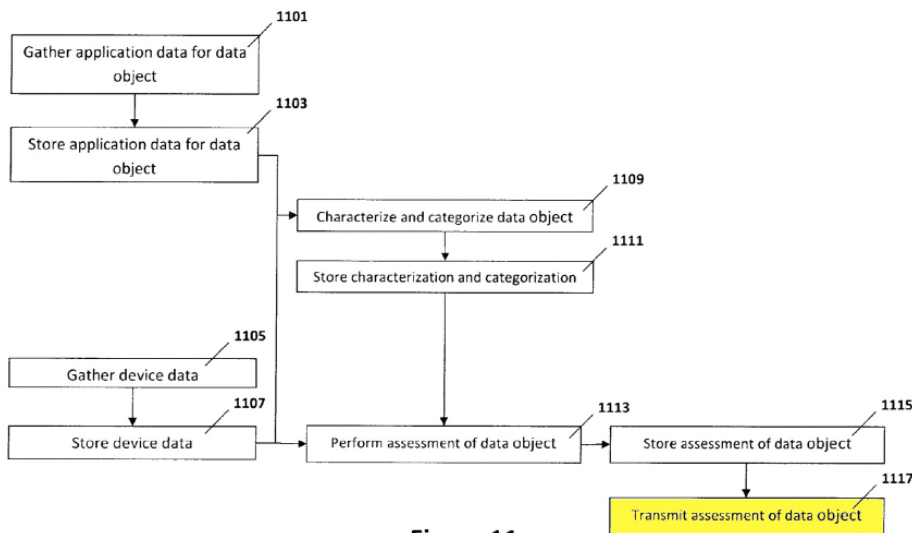


Figure 11

(Id., Fig. 11 (annotated).)

459. Figure 3, reproduced below in annotated form, also shows an example server-performed notification step after data evaluation:

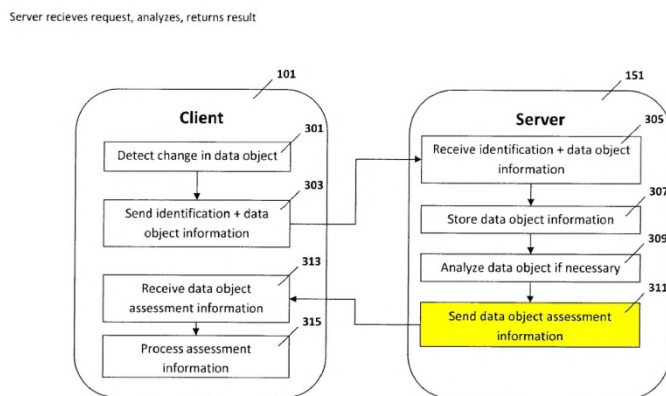


Figure 3

(Id., Fig. 3 (annotated).)

460. As shown, by both the figures above, the responsive action performed by the server occurs immediately after the server's completion of the assessment/analysis step without any intermediate steps or required administrator intervention. (*See id.*, Figs. 3, 11.)

461. Consistent with this, Mahaffey explains that its system is intended to operate such that it provides “real-time protection” of devices. (*Id.*, ¶ [0004].)

[1-6] “wherein the step of evaluating is performed by the server automatically, in response to changes in the status information in the electronic database.”

462. This, in my opinion, is taught by Mahaffey.

463. In particular, Mahaffey explains that its system allows for “automated gathering of data object information” and operates as an “automated analysis system[.]....” (Ex. 1005, ¶¶ [0073], [0104].)

464. Further, per Mahaffey, “server 151 may analyze” a received “data object immediately....” (*Id.*, ¶ [0110].)

465. Figure 11 provides an example of this, showing the performance of assessment immediately after data gathering and storage without any intermediate steps. (*See id.*, Fig. 11.)

466. Because it automatically and immediately analyzes received device data, in my opinion, Mahaffey's server engages in “***evaluating***” “***automatically, in response to changes in the status information in the electronic database***” as required.

b. Claim 10

467. Claim 10 is another independent claim.

[10-p] “10. A method for providing device management services comprising steps of:”

468. To the extent this preamble language is limiting, it is taught by Mahaffey.

469. As I discussed above in connection with element [1-p], Mahaffey teaches a server-based system for managing and protecting mobile devices.

[10-1(a)] “receiving, at a server, a plurality of attributes related to an operating state of each of a plurality of mobile devices,

470. This, in my opinion, is taught by Mahaffey.

471. Again, as I discussed when addressing limitation [1-1(a)], Mahaffey’s system gathers various **“attributes related to an operating state”** of mobile devices.

472. As explained, this includes “device data” like state capabilities, operating system, firmware version, memory capacity, available communication ports, battery limitations, and hardware characteristics.

473. It also includes “application data” like the presence of applications, malware, or spyware, along with flawed data objects, metadata, and device behavioral data. (See *supra* discussion of limitation [1-1(a)].)

[10-1(b)] “each mobile device utilizing one of a plurality of different mobile operating systems,”

474. This, in my opinion, is taught by Mahaffey.

475. As I explained when discussing limitation [1-1(b)], Mahaffey's system can analyze data from devices running various different operating systems.

[10-1(c)] “wherein the attributes for each mobile device are gathered from a plurality of sources including each mobile device;”

476. This, in my opinion, is taught by Mahaffey.

477. Again, as I explain above when discussing limitation [1-1(c)], Mahaffey gathers data from the mobile devices themselves. It also gathers data about the devices (or the applications those devices are running) from other sources, including a “marketplace data gathering system 1005,” or other external network sources providing “characterization data, trust data, distribution data, and user supplied data.”

[10-2] “making a determination that an attribute of a mobile device has changed;”

478. This, in my opinion, is taught by Mahaffey.

479. In particular, Mahaffey “**determin[es]**” that device “**attribute[s]**” have “**changed.**”

480. In this regard, in Mahaffey's system, “mobile communication device 101 detects a change in a specific data object” and communicates this to the server. (Ex. 1005, ¶ [0042].)

481. Similarly, per Mahaffey, “any time application data for a data object has not been transmitted from mobile communication device 101 and received by

server 151, it is considered to be changed and needs to be transmitted.” (*Id.*, ¶ [0043].)

482. And, Mahaffey’s system engages in the “detect[ing] that a data object has changed” such that it requires transmission of that change to the server. (*Id.*, ¶ [0050].)

483. The server in turn monitors for and then evaluates this “change in a data object stored on mobile communication device 101.” (*Id.*, ¶¶ [0042], [0050], [0053]; Figs. 5-6.)

484. According to Mahaffey, this data gathering and analysis process is “automated.” (*Id.*, ¶¶ [0073], [0104].)

485. Figure 3, which is reproduced below, provides an example of Mahaffey’s automatic detection of data changes on a mobile device, along with the relaying of this detected change to the server for analysis.

486. As shown, a data change is detected at step 301. This in turn leads to the transmission of the changed data to the server at step 303, receipt of the changed data by the server at step 305, storage of the changed data by the server at step 307, and evaluation of the changed data by the server at step 309. As shown, this entire process is triggered by a detection of a “change in data object.” And, all the steps are then applied to this changed data:

Server receives request, analyzes, returns result

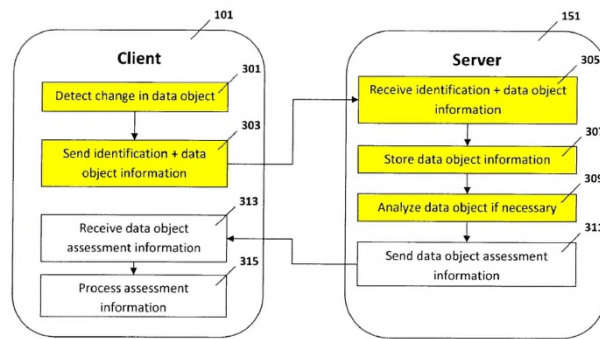


Figure 3

(Id., Fig. 3 (annotated).)

[10-3] “evaluating the attribute, by a processor at the server in response to the determination, to determine if the attribute change triggers at least one of a plurality of administrator-defined rules, which indicates that the attribute is out of compliance, wherein each of the rules applies to mobile devices regardless of mobile operating system; and”

487. This, in my opinion, is taught by Mahaffey.

488. As explained when discussing limitation [1-4], Mahaffey’s system employs a “**plurality of administrator-defined rules**” that apply to monitored devices “**regardless of mobile operating system.**” As I explained, these policies operate to prohibiting certain applications or application functionality or enforce other policy compliance.

489. Next, in my opinion, Mahaffey’s system also operates such that it “**determine[s] if the attribute change triggers at least one of a plurality of administrator-defined rules**” because “**the attribute is out of compliance.**”

490. Again, as explained when discussing limitation [1-4], Mahaffey's server monitors and analyzes changes in mobile device data. (*E.g.*, Ex. 1005, ¶¶ [0042], [0050], [0053]; Fig. 3.)

491. Further, Mahaffey's server uses administrator-defined rules to analyze changed device data received from mobile devices for purposes of "provid[ing] a security related assessment, response and/or other action." (*Id.*, ¶¶ [0027], [0035], [0075], [0153].)

492. This allows Mahaffey's server to identify the presence of policy-violating data on mobile devices.

493. This includes, for instance, blacklisted applications, applications with prohibited functionality, battery wasting applications, or malware (among other things). It also allows the server to determine that the mobile device is out of "policy compliance." (*Id.*, ¶¶ [0086], [0106], [0191].)

[10-4] "automatically initiating, at the server, an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,"

494. This, in my opinion, is taught by Mahaffey.

495. As explained above when discussing limitations [1-5] and [1-6], Mahaffey's system automatically initiates a response at the server as a result of the performed analysis. This can include, for instance, a notification or remediation action. (*See supra* discussion of limitation [1-5].) And, as I have explained,

Mahaffey's system is meant to be "automated." (Ex. 1005, ¶¶ [0073], [0104]; see also *supra* discussion of limitation [1-6].)

[10-5(a)] "wherein the attribute is stored in a data store accessible to the server, that includes substantially real-time data, gathered from the plurality of sources on a network and pertaining to a plurality of attributes of a plurality of mobile devices,

496. This, in my opinion, is taught by Mahaffey.

497. As an initial matter, Mahaffey's system includes the claimed **"data store accessible to the server"**

498. In particular, the "server[] 151" has an internal "data storage 111" that stores transmitted device information. (Ex. 1005, ¶ [0025]; Fig. 1.)

499. Next, in my opinion, Mahaffey teaches that this stored data is **"substantially real-time data."**

500. As discussed above in connection with limitation [1-1(c)], Mahaffey's system operates to "provide constant real-time protection" of devices and allows for data transmission and analysis "whenever it [is] encounter[ed]" by the system. (Ex. 1005, ¶¶ [0004]; [0135].) Once again, real-time protection obviously requires data gathered in real-time. Similarly, Mahaffey also teaches that its system "immediately" analyzes data, and transmits data "whenever it" is "encountered." (*Id.*, ¶¶ [0060], [0110], [0135].)

501. Next, in my opinion, Mahaffey teaches both “*gather[ing]*” data “*from the plurality of sources*” and data “*pertaining to a plurality of attributes of a plurality of mobile devices.*”

502. As explained when discussing limitations [1-1(c)] and [10-1(c)], Mahaffey’s system not only gathers device/application data from many different mobile devices, but it also gathers data relating to the devices from various external sources like marketplace, characterization, trust, distribution, and user supplied data. [10-5(b)] “*wherein the substantially real-time data is formatted, such that the rules can be substantially uniformly applied to the plurality of attributes across the plurality of different mobile operating systems.*”

503. This, in my opinion, is taught by Mahaffey.

504. As I discussed above when addressing limitation [1-2], Mahaffey ensures that its server is provided with “a consistent form of information.”

505. As explained, to do so Mahaffey hashes data before transmission to the server and employs a “cross-platform system” that formats data generated by devices running different operating systems by, for instance, converting it to XML. (See *supra* discussion of limitation [1-2].)

c. Claim 17

506. Claim 17 is another independent claim.

[17-p] “*17. A system for providing device management services comprising:*”

507. This, in my opinion, is taught by Mahaffey for the same reasons I discussed for preambles [1-p] and [10-p] of claims 1 and 10.

[17-1] “a data store configured to be accessible by a server;”

508. This, in my opinion, is taught by Mahaffey.

509. As I explained when discussing limitations [1-3] and [10-5(a)], Mahaffey’s server includes a data storage 111.

[17-2] “a CPU of the server coupled to a memory configured to execute computer readable instructions;”

[17-3] “computer readable instructions for execution on the server comprising instructions configured to:”

510. This, in my opinion, is taught and rendered obvious by Mahaffey.

511. In particular, Mahaffey explains that its server 151 can “operate[]” as a typical HTTP server.” (Ex. 1005, ¶ [0025].)

512. In this regard, the server can employ “computer readable storage medium” (like RAM, ROM, or a “persistent store”) that includes “computer readable instructions or computer program code.” (*Id.*, ¶¶ [0022]-[0023].)

513. Moreover, per Mahaffey, the server engages in operations such as data “analy[sis],” and can generate “a security related assessment, response and/or other action.” (*E.g.*, ¶ [0027].)

514. While Mahaffey does not use the word “CPU,” in my opinion a POSITA would nonetheless understand and consider it obvious that Mahaffey’s server 151 includes a “**CPU**” and coupled “**memory**” with “**computer readable instructions.**”

515. Again, Mahaffey explains that server 151 is typical server that engages in data storage, analysis, and assessment.

516. When the ‘518 patent was filed, it was well-known that a POSITA would have understood that “a typical server includes a processing unit 101, a memory 102, a storage unit 103 and a network interface unit 104....” (*E.g.*, Ex. 1010, ¶ [0025]; *see also id.*, Ex. 1011, 7:63-64 (“A typical server includes an IBM RISC/6000 with AIX operating system and server program”); Fig. 5 (showing a “server” with a “computer” and “application”).)

517. A POSITA would have understood that Mahaffey’s server is no different from these typical servers: it includes a processor allowing it to engage in data analysis. It also includes memory to store instructions. All servers have these components.

[17-3(a)] “automatically make a determination that an attribute of a mobile device stored in the data store has changed;”

518. This, in my opinion, is taught by Mahaffey.

519. As explained above when discussing limitation [10-2], Mahaffey’s system is automated, and monitors for mobile device attribute changes.

[17-3(b)] “evaluate the attribute, in response to the determination, to determine if the attribute change triggers at least one of a plurality of administrator-defined rules, which indicates that the attribute is out of compliance, wherein each of the rules applies to mobile devices regardless of mobile operating system; and”

520. This, in my opinion, is taught by Mahaffey.

521. As I explained above when discussing limitations [1-4] and [10-3], Mahaffey's server evaluates device and application data using administrator-defined rules. This allows the server to identify whether the devices are running blacklisted applications, employ applications with prohibited functionality, include malware, or are otherwise non-policy compliant.

[17-3(c)] "automatically initiate an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,"

522. This, in my opinion, is taught by Mahaffey.

523. In particular, as I explained when discussing limitations [1-5] and [1-6], Mahaffey's server automatically initiates a response in response to the performed analysis. This can include sending a notification to a device or taking another remediation action. (*See supra* discussion of limitation [1-5].) And, this occurs automatically. (*See supra* discussion of limitations [1-6] and [10-4].)

[17-3(d)] "wherein the data store is configured to gather substantially real-time data pertaining to a plurality of attributes related to an operating state of each of the plurality of mobile devices, the attributes for each mobile device being gathered from a plurality of sources including each mobile device, wherein each mobile device utilizes one of a plurality of different mobile operating systems, and the substantially real-time data is formatted, such that the rules can be substantially uniformly applied to the plurality of attributes across the plurality of different mobile operating systems."

524. This, in my opinion, is all taught by Mahaffey.

525. First, as explained when discussing limitation [1-1(c)], Mahaffey's system gathers data in "***substantially real-time***" as that data is generated. (Ex. 1005, ¶¶ [0004], [0060], [0135].)

526. Next, as I also explained in connection with limitation [1-1(c)], the gathered "***attributes***" come from "***a plurality of sources including each mobile device.***" Once again, Mahaffey collects not only device/application data from the mobile devices its system manages, but also data relating to the devices from other external sources.

527. Further, as explained when discussing limitation [1-1(b)], Mahaffey's system is also designed to work with different mobile devices running "***different mobile operating systems.***" (E.g., Ex. 1005, ¶ [0022].)

528. Finally, as I explained in connection with limitation [1-2], Mahaffey "***format[s]***" data to allow "***rules***" to "***be substantially uniformly applied.***"

529. Again, Mahaffey's servers are provided with "a consistent form of information." The system also employs a "cross-platform system" that can format data generated by devices running different operating systems by, for instance, hashing and converting it to XML. (See *supra* discussion of limitation [1-2].)

2. Dependent Claim Analysis

a. Claim 2

530. Claims 2 depends on claim 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising blocking access by the at least one of the mobile devices to a corporate resource”

531. As I explain above, Mahaffey teaches everything claim 1 requires. It also teaches the additional limitation required by this dependent claim.

532. In particular, Mahaffey’s server can perform “a variety of remediation actions” in view of its evaluation of device data and application of administrator-defined policies. (Ex. 1005, ¶ [0195].)

533. Per Mahaffey, these actions include “isolating the ... device from accessing ... corporate email or VPN server while it is out of compliance....” (*Id.*)

b. Claim 3

534. Claims 3 depends on claims 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising blocking e-mail access for the at least one of the mobile devices.”

535. As I explain above, Mahaffey teaches everything claim 1 requires. It also teaches the additional limitation required by this dependent claim.

536. Again, as just explained when discussing claim 2, one of the actions Mahaffey’s server can take is blocking access to “corporate email.” (Ex. 1005, ¶ [0195].)

c. Claim 4

537. Claims 4 depends on claims 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising wiping information from the at least one of the mobile devices.”

538. As I explain above, Mahaffey teaches everything claim 1 requires. It also teaches the additional limitation required by this dependent claim.

539. Another of the actions that can be performed by Mahaffey’s server after evaluation is ***“wiping information”*** as this claim requires. In particular, Mahaffey’s server can respond to analysis by causing “any violating applications” to be “uninstalled” from a managed device. (Ex. 1005, ¶ [0195].) In such circumstances, the application is ***“wip[ed]”*** from the device.

d. Claim 5

540. Claims 5 depends on claims 1 and further requires that:

“wherein the at least one action is selectable from a plurality of actions, comprising warning a user of the at least one of the mobile devices that further action may be taken.”

541. As I explain above, Mahaffey teaches everything claim 1 requires. It also teaches the additional limitation required by this dependent claim.

542. Mahaffey’s server can be configured to “transmit notifications or instructions to remediate data objects whose assessment has changed, requiring action by” a mobile device. (Ex. 1005, ¶ [0047].)

543. This notification can be in the form of an “alert, a message, [or] an instruction.” (*Id.*, ¶ [0040]; *see also id.*, ¶ [0131] (explaining that a server can provide a device with a “notification, a warning, a risk rating, or the like”).)

544. Mahaffey’s system is also able to take other actions such as “isolating” the device from network resources, or “uninstall[ing]” applications. (*Id.*, ¶ [0195].)

545. In my opinion, a POSITA would have understood that failure of a device to comply with a previously provided notification/instruction would obviously lead to Mahaffey’s server taking one of these more serious actions.

546. A POSITA would have understood that the purpose of the notification/instruction is to provide the device user with an opportunity to self-correct before the server takes a more invasive action.

e. Claims 6, 14, and 21

547. Claims 6, 14, and 21 respectively depend on claims 1, 10, and 17. They all similarly require:

“gathering [attribute] information from that mobile device and at least one other source on the network”

548. As I explain above, Mahaffey teaches everything claims 1, 10, and 17 require. It also teaches the additional limitation required by these dependent claims.

549. As discussed in connection with limitation [1-1(c)], Mahaffey gathers information not only from the mobile devices themselves, but also from other network sources a “marketplace data gathering system 1005,” or external “characterization data, trust data, distribution data, and user supplied data.” (See *supra* discussion of limitation [1-1(c)].)

f. Claims 7, 12, and 20

550. Claims 7, 12, and 20 respectively depend on claims 1, 10, and 17 and further require:

“the step of subsequently evaluating a second compliance state of at least one of the mobile devices to determine if the at least one action should be revoked” (claim 7)

“[the step of]/[wherein the computer readable instructions further comprise instructions configured to] automatically [revoking]/[revoke] the action if the attribute is brought back into compliance” (claims 12, 20)

551. As I explain above, Mahaffey teaches everything claims 1, 10, and 17 require. It also teaches the additional limitation required by these dependent claims.

552. Mahaffey’s system can “isolat[e]” a monitored device “from accessing ... corporate email or VPN server while it is out of compliance....” (Ex. 1005, ¶ [0195].)

553. This—and Mahaffey’s other responsive actions—are “automated.” (See *id.*, ¶ [0073], [0104].)

554. Because isolation occurs only when the device is “out of compliance,” a POSITA would have understood that the monitored device would obviously be automatically re-allowed to access these resources once it is determined to be compliant.

555. Additionally, it would make little sense for Mahaffey’s system to permanently prohibit a device from accessing network resources whenever that device is determined to be non-compliant. Instead, the access decision would be

continuously re-evaluated and the device would be re-allowed access once it again complies with all applicable policies.

g. Claims 8, 15, and 22

556. Claims 8, 15, and 22 respectively depend on claims 1, 10, and 17. They all further requires that:

“the plurality of different mobile operating systems comprise at least operating systems compliant with Android and iOS specifications”

557. As I explain above, Mahaffey teaches everything claims 1, 10, and 17 require. It also teaches the additional limitation required by these dependent claims.

558. Mahaffey explains that its system is meant to monitor, gather data from, and apply policies to devices running both “Apple iOS™” and “Android™ OS.” (Ex. 1005; ¶ [0022].)

h. Claims 9, 16, and 23

559. Claims 9, 16, and 23 respectively depend on claims 1, 10, and 17 and further relatedly require:

“wherein the [status information / attribute] comprises location information about [each / the] mobile device”

560. As I explain above, Mahaffey teaches everything claims 1, 10, and 17 require. It also teaches the additional limitation required by these dependent claims.

561. According to Mahaffey, its “server may use a device’s country ... or area code” when making an assessment. (Ex. 1005, ¶ [0117].)

562. In my opinion, both country and area code are examples of device “*location information.*”

3. Motivation to Combine

563. A POSITA would have considered it obvious and would have been motivated to apply Mahaffey’s teachings in the manner discussed above to arrive at the claimed subject matter. (*See* Ex. 1003, ¶¶ __-__.)

564. As an initial matter, if it is for some reason determined that Mahaffey did not properly incorporate the ’591 application by reference, in my opinion a POSITA would have still considered it obvious to apply its teachings to Mahaffey.

565. Mahaffey teaches that its server gathers and evaluates data from different devices running different operating systems. (*See, e.g.*, Ex. 1005, ¶ [0022].)

566. Mahaffey also explains that a “cross-platform system” can be used to facilitate communication between the server and the different-operating system devices. (*Id.*, ¶ [0028].)

567. Moreover, Mahaffey specifically points to the ’591 application as providing an example of one such “cross-platform system” that can be used to accomplish its system’s functionality (*See id.*)

568. In my opinion, this explicit instruction would have provided a POSITA with a strong motivation to apply the ’591 application to Mahaffey. Mahaffey’s instruction to consider, use, and apply the ’591 application would cause a POSITA

to consider and apply that application's teachings when implementing Mahaffey's system.

569. Further, I note that the '591 application itself articulates numerous benefits deriving from the use of its cross-platform system.

570. For example, the '591 application's cross-platform system is "lightweight" "without sacrificing functionality or power." (Ex. 1007, ¶ [0021].) The application goes on to explain that because of this, it allows communication between different devices without consuming excessive system resources. (*See id.*, ¶¶ [0004], [0021].)

571. Further, the '591 application states that it allows for the use of "a common API" for communication instead of "different APIs as is common in the prior art." (*Id.*, ¶ [0021].) According to the '591 application, this allows for the re-use of code and avoids the need for excessive and difficult customization. (*See id.*)

572. In my opinion, these various expressed benefits and advantages would also have motivated the application of the '591 application's cross-platform system (and data formatting procedures) to Mahaffey. And, a POSITA would have recognized that the '591 application's benefits would have translated to and improved Mahaffey.

573. For instance, Mahaffey would be improved by reducing the resources consumed when a server communicates with different types of devices.

574. Further, use of common APIs would have rendered the code underlying Mahaffey less complicated, more efficient, and less prone to software bugs. Custom APIs would not need to be developed on a device-by-device basis, which would reduce the risk for bugs and their resulting security vulnerabilities.

575. Next, Mahaffey discusses a number of more specific examples of its system and system components. This includes a system for collecting data (*e.g.*, Ex. 1005, ¶ [0069]), a server-side analysis system (*e.g.*, *id.*, ¶ [0075]), an anti-malware system (*e.g.*, *id.* ¶ [0106]), and an application assessment and advisement system (*e.g.*, *id.*, ¶ [0163]).

576. In my opinion, a POSITA would have understood that these are simply meant to be representative of the overall operation and capabilities of Mahaffey’s system.

577. A POSITA would not have viewed these examples in isolation, and instead would have understood that they are meant to exemplify—when viewed together—the range of features and functionality provided by the overall management system disclosed in and taught by Mahaffey.

578. Consistent with this, Mahaffey itself explains that its system is generally able to “provid[e] security to a plurality of mobile communication devices.” (Ex. 1005, ¶ [0021].)

579. Moreover, Mahaffey's system is designed and intended to flexibly allow for the collection of many different types of device and application data, engage in a variety of different types of analyses, and generate various different responses. (*See id.*, ¶¶ [0030], [0075], [0195].)

580. Because of this, a POSITA would have been motivated to use the features of these various example systems together.

581. In my opinion, a POSITA would also have had a more than reasonable expectation of success.

582. As I have repeatedly explained, Mahaffey itself teaches that its system is designed for and meant to use the cross-platform system of the '591 application. (*See Ex. 1005*, ¶ [0028].)

583. Moreover, Mahaffey itself notes that its various different embodiments are non-limiting, and are just examples of the overall capabilities of its system. (*See e.g., id.*, ¶ [0210].)

584. In view of this, in my opinion a POSITA would have every reason to believe—and no reason to doubt—that Mahaffey's system would be successfully designed to not only employ the '591 patent's cross-platform system, but provide all the functionality set forth in Mahaffey.

D. Secondary Considerations of Non-Obviousness

585. Based on my review of the '518 patent's prosecution history, I note that no secondary considerations of non-obviousness were identified in connection with the patent's claims.

586. I am not separately aware of any evidence of commercial success, skepticism, failure of others, industry praise, unexpected results, or the like. Instead, as discussed above, it is my opinion that the subject matter embraced by the '518 patent's claims was known in the art and would have been expected to function as a working method and system where a server gathers data from and manages mobile devices.

I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements were made with the knowledge that willfully false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1101 of Title 18 of the United States Code.

Date: 10/30/2023

Location: New York, NY

Signature: 
MARKUS JAKOBSSON, Ph.D.

ATTACHMENT 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. Kutylowski, (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 unmaskers 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/
- [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.