

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NETFLIX, INC. AND HULU, LLC,

Petitioners,

v.

DIVX, LLC,

Patent Owner.

Case No. IPR2020-00558

U.S. Patent 10,225,588

Declaration of Patrick McDaniel, Ph.D.

TABLE OF CONTENTS

I.	BACKGROUND AND QUALIFICATIONS	5
B.	Compensation	9
C.	Materials and Other Information Considered	9
II.	UNDERSTANDING OF THE LAW	9
A.	Legal Standard for Prior Art	9
B.	Anticipation and Prior Art.....	10
C.	Obviousness.....	12
D.	Legal Standard for Claim Construction.....	17
III.	LEVEL OF ORDINARY SKILL IN THE ART	22
IV.	OVERVIEW OF THE TECHNOLOGY	25
A.	Adaptive Bitrate Streaming.....	25
B.	DRM.....	31
1.	Key Management.....	32
2.	Partial Encryption	33
V.	THE '588 PATENT	35
A.	Background Art Admitted by the '588 Patent	35
B.	Summary of the Alleged Invention	39
C.	The Alleged Invention According to Patent Owner.....	40
D.	Priority Date	40
E.	Challenged Claims	41
VI.	CLAIM CONSTRUCTION	41
VII.	SUMMARY OF THE PRIOR ART.....	42

A.	Invalidity Grounds	42
B.	U.S. Patent Pub. No. 2011/0096828 (“Chen”) (Exhibit 1006).....	43
C.	U.S. Patent Pub. No. 2007/0083467 (“Lindahl”) (Exhibit 1007)	45
D.	U.S. Patent 8,683,066 (“Hurst”) (Exhibit 1008)	45
VIII.	INVALIDITY	46
A.	Ground 1: Chen in View of Lindahl and Hurst.....	46
1.	Motivation to Combine	46
2.	Claim 1	59
3.	Claim 2	123
4.	Claim 3	129
5.	Claim 4	132
6.	Claim 5	132
7.	Claim 6	133
8.	Claim 7	134
9.	Claim 8	138
10.	Claim 9	140
11.	Claim 10	143
12.	Claim 11	145
13.	Claim 12	150
14.	Claim 13	152
15.	Claim 14	152
16.	Claim 15	153
17.	Claim 16	153

18.	Claim 17	153
19.	Claim 18	154
20.	Claim 19	155
21.	Claim 20	155
22.	Claim 21	156
23.	Claim 22	156
24.	Claim 23	157
25.	Claim 24	160
IX.	RESERVATION OF RIGHTS.....	161

I, Patrick McDaniel, declare as follows:

1. My name is Patrick McDaniel. I am a Professor of Computer Science at Penn State University and Director of the Institute for Network and Security Research. I have prepared this report as an expert witness retained by Netflix, Inc. and Hulu, LLC. In this report I give my opinions as to whether certain claims of U.S. Patent No. 10,225,588 (“the ’588 patent”) are invalid. I provide technical bases for these opinions as appropriate.

2. This report contains statements of my opinions formed to date and the bases and reasons for those opinions. I may offer additional opinions based on further review of materials in this case, including opinions and/or testimony of other expert witnesses. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters contained herein.

I. BACKGROUND AND QUALIFICATIONS

3. This section contains a summary of my educational background, career history, publications, and other relevant qualifications. My full curriculum vitae is attached as Appendix 1 to this declaration.

4. Since 2017, I have been the William L. Weiss Professor of Information and Communications Technology in the School of Electrical Engineering and Computer Science at Pennsylvania State University in University

Park, PA. I am also the director of the Institute for Network and Security Research, and founder and co-director of the Systems and Internet Infrastructure Security Laboratory, a research laboratory focused on the study of security in diverse network and computer environments. My research efforts primarily involve cryptography, key management, computer systems, network, management, authentication, systems security, and technical public policy.

5. Before my current position, I was an Assistant Professor (2004-2007), Associate Professor (2007-2011), Full Professor (2011-2015), and Distinguished Professor of Computer Science and Engineering at Pennsylvania State University (2015-2017). Since 2004, I have taught several courses in the field of computer systems, systems programming, networks, and network and computer security at both the undergraduate and graduate level. I created and continue to maintain several of these courses for Penn State.

6. From 2003-2009, I was also an Adjunct Professor at the Stern School of Business at New York University in New York, NY. At the Stern School of Business, I taught courses in computer and network security and online privacy.

7. My qualifications for forming the opinions in this report are summarized here. I earned a Ph.D. in Computer Science and Engineering from University of Michigan, Ann Arbor in 2001. I earned a Bachelor of Science

degree in Computer Science from Ohio University in 1989 and a Master of Science degree, also in Computer Science, from Ball State University in 1991.

8. I am a Fellow of the Association for Computing Machinery (the leading professional association for computer science) and the Institute for Electrical and Electronics Engineering (the leading professional association for computer engineering).

9. I was also the Program Manager (PM) and lead scientist for the Cyber Security Collaborative Research Alliance (CRA) from 2013 to 2018. The CRA is led by Penn State University and includes faculty and researchers from the Army Research Laboratory, Carnegie Mellon University, Indiana University, the University of California-Davis, and the University of California-Riverside. This initiative is a major research project aimed at developing a new science of cyber-security for military networks, computers, and installations.

10. I have served as an advisor to several Ph.D. and master's degree candidates, several of whom have gone on to become professors at various institutions such as North Carolina State University, the University of Oregon, and the Georgia Institute of Technology. I am currently an advisor to two Ph.D. candidates and a number of master's students.

11. Before joining Pennsylvania State University as a professor, I was a software developer and project manager for companies in the networking industry

including Applied Innovation, Inc. and Primary Access Corporation. I was also a senior researcher at AT&T Research-Labs. As part of my duties in these industrial positions, I informed, reviewed and formed corporate policies and practices relating to the deployment and subsequent management of software systems.

12. I have published extensively in the field of network and security management, computer systems, authentication, systems security, applied cryptography and network security. I have published well over 200 papers which have been cited over 26,000 times. In addition to writing several articles for industry journals and conferences, I have authored portions of numerous books related to computer systems, applied cryptography and network security. I have served on the editorial boards of several peer-reviewed journals including ACM Transactions on Internet Technology, for which I was the Editor-in-Chief. I was also an Associate Editor for ACM Transactions on Information and System Security and IEEE Transactions of Software Engineering, two highly-regarded journals in the field. A complete list of my publications in the last 10 years, as well as a list of editorial positions can be found in my curriculum vitae, which is attached as Appendix 1.

13. Over the last 20 years I have frequently taught classes in security and software systems including, among others, introduction to computer and network security, advanced network security, and mobile system security. The courses

covered topics in basic security and cryptography that relate to many of the issues relevant to this case including the selection and use of ciphers, securing media, and access control. I have also discussed broadcast systems such as streaming video in course of my teaching, and explored related issues in the course of my professional research.

B. Compensation

14. For my efforts in connection with the preparation of this declaration I have been compensated at my standard rate of \$600 per hour for this type of consulting activity. My compensation is in no way contingent on the results of these or any other proceedings relating to the above-captioned patent.

C. Materials and Other Information Considered

15. I have considered information from various sources in forming my opinions. I have reviewed and considered each of the exhibits listed in the attached Appendix 2 (Materials Considered) in forming my opinions.

II. UNDERSTANDING OF THE LAW

16. I have applied the following legal principles provided to me by counsel in arriving at the opinions set forth in this report.

A. Legal Standard for Prior Art

17. I am not an attorney. I have been informed by attorneys of the relevant legal principles and have applied them to arrive at the opinions set forth in this declaration.

18. I understand that the petitioner for inter partes review may request the cancellation of one or more claims of a patent based on grounds available under 35 U.S.C. § 102 and 35 U.S.C. § 103 using prior art that consists of patents and printed publications.

B. Anticipation and Prior Art

19. I understand that § 102 specifies when a challenged claim is invalid for lacking novelty over the prior art, and that this concept is also known as “anticipation.” I understand that a prior art reference anticipates a challenged claim, and thus renders it invalid by anticipation, if all elements of the challenged claim are disclosed in the prior art reference. I understand the disclosure in the prior art reference can be either explicit or inherent, meaning it is necessarily present or implied. I understand that the prior art reference does not have to use the same words as the challenged claim, but all of the requirements of the claim must be disclosed so that a person of ordinary skill in the art could make and use the claimed subject-matter.

20. In addition, I understand that § 102 also defines what is available for use as a prior art reference to a challenged claim.

21. Under § 102(a), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country before the challenged claim’s date of invention.

22. Under § 102(b), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country more than one year prior to the challenged patent's filing date.

23. Under § 102(e), a challenged claim is anticipated if it was described in published patent application that was filed by another in the United States before the challenge claim's date of invention, or was described in a patent granted to another that was filed in the United States before the challenged claim's date of invention.

24. I understand that a challenged claim's date of invention is presumed to be the challenged patent's filing date. I also understand that the patent owner may establish an earlier invention date and "swear behind" prior art defined by § 102(a) or § 102(e) by proving (with corroborated evidence) the actual date on which the named inventors conceived of the subject matter of the challenged claim and proving that the inventors were diligent in reducing the subject matter to practice.

25. I understand that the filing date of patent is generally the filing date of the application filed in the United States that issued as the patent. However, I understand that a patent may be granted an earlier effective filing date if the patent owner properly claimed priority to an earlier patent application.

26. I understand that when a challenged claim covers several structures, either generically or as alternatives, the claim is deemed anticipated if any of the structures within the scope of the claim is found in the prior art reference.

27. I understand that when a challenged claim requires selection of an element from a list of alternatives, the prior art teaches the element if one of the alternatives is taught by the prior art.

C. Obviousness

28. I understand that even if a challenged claim is not anticipated, it is still invalid if the differences between the claimed subject matter and the prior art are such that the claimed subject matter would have been obvious to a person of ordinary skill in the pertinent art at the time the alleged invention.

29. I understand that obviousness must be determined with respect to the challenged claim as a whole.

30. I understand that one cannot rely on hindsight in deciding whether a claim is obvious.

31. I also understand that an obviousness analysis includes the consideration of factors such as (1) the scope and content of the prior art, (2) the differences between the prior art and the challenged claim, (3) the level of ordinary skill in the pertinent art, and (4) “secondary” or “objective” evidence of non-obviousness.

32. Secondary or objective evidence of non-obviousness includes evidence of: (1) a long felt but unmet need in the prior art that was satisfied by the claimed invention; (2) commercial success or the lack of commercial success of the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise of the claimed invention by others skilled in the art; (5) taking of licenses under the patent by others; (6) deliberate copying of the claimed invention; and (7) contemporaneous and independent invention by others. However, I understand that there must be a relationship between any secondary evidence of non-obviousness and the claimed invention.

33. I understand that a challenged claim can be invalid for obviousness over a combination of prior art references if a reason existed (at the time of the alleged invention) that would have prompted a person of ordinary skill in the art to combine elements of the prior art in the manner required by the challenged claim. I understand that this requirement is also referred to as a “motivation to combine,” “suggestion to combine,” or “reason to combine,” and that there are several rationales that meet this requirement.

34. I understand that the prior art references themselves may provide a motivation to combine, but other times simple common sense can link two or more prior art references. I further understand that obviousness analysis recognizes that

market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may come from market forces.

35. I understand obviousness to include, for instance, scenarios where known techniques are simply applied to other devices, systems, or processes to improve them in an expected or known way. I also understand that practical and common-sense considerations should be applied a proper obviousness analysis. For instance, familiar items may have obvious uses beyond their primary purposes.

36. I understand that the combination of familiar elements according to known methods is obvious when it yields predictable results. For instance, obviousness bars patentability of a predictable variation of a technique even if the technique originated in another field of endeavor. This is because design incentives and other market forces can prompt variations of it, and predictable variations are not the product of innovation, but rather ordinary skill and common sense.

37. I understand that a particular combination may be obvious if it was obvious to try the combination. 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 person of ordinary skill has good reason to pursue the known options within his or her technical grasp. This would result in something obvious because the result is the product not of innovation but of ordinary skill and

common sense. However, I understand that it may not be obvious to try a combination when it involves unpredictable technologies.

38. It is further my understanding that a proper obviousness analysis focuses on what was known or obvious to a person of ordinary skill in the art, not just the patentee. Accordingly, I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

39. It is my understanding that the Manual of Patent Examining Procedure §2143 sets forth the following as exemplary rationales that support a conclusion of obviousness:

40. Combining prior art elements according to known methods to yield predictable results;

41. Simple substitution of one known element for another to obtain predictable results;

42. Use of known technique to improve similar devices (methods, or products) in the same way;

43. Applying a known technique to a known device (method, or product) ready for improvement to yield predictable results;

44. Choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;

45. Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art;

46. Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

47. A person of ordinary skill in the art looking to overcome a problem will often use the teachings of multiple publications together like pieces of a puzzle, even though the prior art does not necessarily fit perfectly together. Therefore, I understand that references for obviousness need not fit perfectly together like puzzle pieces. Instead, I understand that obviousness analysis takes into account inferences, creative steps, common sense, and practical logic and applications that a person of ordinary skill in the art would employ under the circumstances.

48. I understand that a claim can be obvious in light of a single reference, if the elements of the challenged claim that are not explicitly or inherently disclosed in the reference can be supplied by the common sense of one of skill in the art.

49. I understand that obviousness also bars the patentability of applying known or obvious design choices to the prior art. One cannot patent merely

substituting one prior art element for another if the substitution can be made with predictable results. Likewise, combining prior art techniques that are interoperable with respect to one another is generally obvious and not patentable.

50. In sum, my understanding is that obviousness invalidates claims that merely recite combinations of, or obvious variations of, prior art teachings using understanding and knowledge of one of skill in the art at the time and motivated by the general problem facing the inventor at the time. Under this analysis, the prior art references themselves, or any need or problem known in the field of endeavor at the time of the invention, can provide a reason for combining the elements of or attempting obvious variations on prior art references in the claimed manner.

D. Legal Standard for Claim Construction

51. I understand that before any invalidity analysis can be properly performed, the scope and meaning of the challenged claims must be determined by claim construction.

52. I understand that a patent may include two types of claims, independent claims and dependent claims. I understand that an independent claim stands alone and includes only the limitations it recites. I understand that a dependent claim depends from an independent claim or another dependent claim. I understand that a dependent claim includes all the limitations that it recites in addition to the limitations recited in the claim (or claims) from which it depends.

53. In comparing the challenged claims to the prior art, I have carefully considered the patent and its file history in light of the understanding of a person of skill at the time of the alleged invention.

54. I understand that to determine how a person of ordinary skill would have understood a claim term, one should look to sources available at the time of the alleged invention that show what a person of skill in the art would have understood disputed claim language to mean. It is my understanding that this may include what is called “intrinsic” evidence as well as “extrinsic” evidence.

55. I understand that, in construing a claim term, one should primarily rely on intrinsic patent evidence, which includes the words of the claims themselves, the remainder of the patent specification, and the prosecution history. I understand that extrinsic evidence, which is evidence external to the patent and the prosecution history, may also be useful in interpreting patent claims when the intrinsic evidence itself is insufficient. I understand that extrinsic evidence may include principles, concepts, terms, and other resources available to those of skill in the art at the time of the invention.

56. I understand that words or terms should be given their ordinary and accepted meaning unless it appears that the inventors were using them to mean something else or something more specific. I understand that to determine whether a term has special meaning, the claims, the patent specification, and the

prosecution history are particularly important, and may show that the inventor gave a term a particular definition or intentionally disclaimed, disavowed, or surrendered claim scope.

57. I understand that the claims of a patent define the scope of the rights conferred by the patent. I understand that because the claims point out and distinctly claim the subject matter which the inventors regard as their invention, claim construction analysis must begin with and is focused on the claim language itself. I understand that the context of the term within the claim as well as other claims of the patent can inform the meaning of a claim term. For example, because claim terms are normally used consistently throughout the patent, how a term is used in one claim can often inform the meaning of the same term in other claims. Differences among claims or claim terms can also be a useful guide in understanding the meaning of particular claim terms.

58. I understand that a claim term should be construed not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the entire specification. I understand that because the specification is a primary basis for construing the claims, a correct construction must align with the specification.

59. I understand that the prosecution history of the patent as well as art incorporated by reference or otherwise cited during the prosecution history are also

highly relevant in construing claim terms. For instance, art cited by or incorporated by reference may indicate how the inventor and others of skill in the art at the time of the invention understood certain terms and concepts.

Additionally, the prosecution history may show that the inventors disclaimed or disavowed claim scope or further explained the meaning of a claim term.

60. With regard to extrinsic evidence, I understand that all evidence external to the patent and prosecution history, including expert and inventor testimony, dictionaries, and learned treatises, can also be considered. For example, technical dictionaries may indicate how one of skill in the art used or understood the claim terms. However, I understand that extrinsic evidence is considered to be less reliable than intrinsic evidence, and for that reason is generally given less weight than intrinsic evidence.

61. I understand that in general, a term or phrase found in the introductory words or preamble of the claim, should be construed as a limitation if it recites essential structure or steps, or is necessary to give meaning to the claim. For instance, I understand preamble language may limit claim scope: (i) if dependence on a preamble phrase for antecedent basis indicates a reliance on both the preamble and claim body to define the claimed invention; (ii) if reference to the preamble is necessary to understand limitations or terms in the claim body; or (iii) if the

preamble recites additional structure or steps that the specification identifies as important.

62. On the other hand, I understand that a preamble term or phrase is not limiting where a challenged claim defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention. I understand that to make this determination, one should review the entire patent to gain an understanding of what the inventors claim they invented and intended to encompass in the claims.

63. I understand that 35 U.S.C. § 112 ¶ 6 created an exception to the general rule of claim construction called a “means plus function” limitation. These types of terms and limitations should be interpreted to cover only the corresponding structure described in the specification, and equivalents thereof. I also understand that a limitation is presumed to be a means plus function limitation if (a) the claim limitation uses the phrase “means for”; (b) the “means for” is modified by functional language; and (c) the phrase “means for” is not modified by sufficient structure for achieving the specified function.

64. I understand that a structure is considered structurally equivalent to the corresponding structure identified in the specification only if the difference between them are insubstantial. For instance, if the structure performs the same function in substantially the same way to achieve substantially the same result. I

further understand that a structural equivalent must have been available at the time of the issuance of the claim.

III. LEVEL OF ORDINARY SKILL IN THE ART

65. In forming my opinions on obviousness and determining the characteristics of a hypothetical person of ordinary skill in the art (“POSITA”) of the ’588 patent at the time of the claimed invention, I considered the level of ordinary skill in the art from September 1, 2011, the date of the provisional application of the ’588 patent, through December 29, 2011, the date of the earliest cited application on the face of the ’588 patent. *See infra* Section V.D. I understand that Petitioners are not aware of any claim by the Patent Owner that the ’588 patent is entitled to an earlier priority date. My opinions regarding obviousness and the level of skill would not materially change if the challenged claims were entitled to a priority date within a year earlier. In addition, if the claims are entitled to a later priority date and not entitled to claim priority to 2011, it is my opinion that a person of ordinary skill in the art may have even further knowledge and ability, but that those incremental increases in the level of skill would not materially impact the analysis presented here.

66. In forming my opinions regarding the level of skill for a POSITA, I considered several things, including various prior art techniques relating to the field of video streaming and media security, the type of problems that such

techniques gave rise to, and the rapidity with which innovations were made. I also considered the sophistication of the technologies involved, and the educational background and experience of those actively working in the field at the time. I also considered the level of education that would be necessary to understand the '588 patent. Finally, I placed myself back in the relevant period of time and considered the engineers and programmers that I have worked with and managed in the field of video streaming and media security.

67. I came to the conclusion that a person of ordinary skill in the field of art of the '588 patent would have been a person with a bachelor's degree in mechanical engineering, electrical engineering, computer science, or a similar field with at least two years of experience in video streaming and media security or a person with a master's degree in mechanical engineering, electrical engineering, computer science, or a similar field with a specialization in video streaming and media security. A person with less education but more relevant practical experience may also meet this standard.

68. Given this level of skill in the 2011 time frame, a POSITA would have had the knowledge and skills necessary to program and build systems that meet all limitations of the challenged claims, including the following:

- Compressing and decompressing video, including by using common video standards such as MPEG and AVC. *See, e.g.*, Ex. 1007, ¶[0006]; Ex. 1009, 1:15-37.
- Using known video container file formats for streaming, such as the ISO base media file format, sometimes referred to as the MP4 container format, and 3GPP extensions to the ISO base media file format. *See, e.g.*, Ex. 1009, 1:15-37; Ex. 1010, 2:59-3:6. The '588 patent admits this was known in the art. *See* Ex. 1001, 2:27-41.
- Using known communication protocols for streaming, including Hypertext Transfer Protocol (HTTP). *See, e.g.*, Ex. 1009, 1:38-3:3; Ex. 1010, 3:7-22. The '588 patent admits this was known in the art. *See* Ex. 1001, 2:1-10.
- Using known adaptive streaming technologies to stream video over the Internet, including MPEG Dynamic Adaptive Streaming over HTTP (DASH) and adaptive bitrate streaming solutions from companies such as Microsoft, Adobe, and Apple. *See, e.g.*, Ex. 1009, 2:1-3:3. The '588 patent admits this was known in the art. *See* Ex. 1001, 2:12-34.
- Encrypting and decrypting data, including by using common symmetric key algorithms (e.g., Blowfish, AES, DES) and asymmetric key algorithms. *See, e.g.*, Ex. 1007, ¶[0008]; Ex. 1010, 7:34-42. The '588 patent admits this was known in the art. *See* Ex. 1001, 2:47-62.

- Using DRM for video security, including for authentication, key management (such as the assignment and identification of keys), and partial encryption of video streams. *See, e.g.*, Ex. 1011, 1:49-63, 2:1-8, 8:23-32; Ex. 1013, 6:15-56, 7:15-8:42.

See Section V.A.

IV. OVERVIEW OF THE TECHNOLOGY

69. By 2011, Netflix and Hulu had launched video streaming services. Many technology companies, such as Microsoft, Apple and Adobe, provided tools for adaptive bitrate streaming. *See, e.g.*, Ex. 1009, 2:1-10; Ex. 1010, 1:31-59, 2:59-3:22, 3:34-4:16 . By 2011, several common components of adaptive streaming had become standardized, including file formats, digital rights management (“DRM”) and encryption. *See, e.g.*, Ex. 1010, 8:47-9:8. Many open source tools existed for these features, which were commonly used together to deliver video streaming services. Adaptive bitrate streaming and file formats were used to deliver content to users, while DRM and encryption were used to protect content from unauthorized access.

A. Adaptive Bitrate Streaming

70. By 2011, adaptive bitrate streaming allowed video streaming to compensate for changes in network bandwidth, which was important when dealing with highly variable channel conditions over the Internet and Wi-Fi. In known

adaptive bitrate streaming methods, a video is encoded at several different bitrates, and playback of the video switches between the different encodings depending on the available bandwidth of the transmission channel. *See, e.g.*, Ex. 1009, 2:26-30; Ex. 1010, 14:61-15:14.

71. Popular standards, such as MPEG Dynamic Adaptive Streaming over HTTP (“DASH”) included a Media Presentation Description (MPD) that referenced the fragments of each representation of video. *See, e.g.*, Ex. 1009, 2:19-25. Such references could be in the form of uniform resource locators (URLs). *See, e.g., id.* For example, a MPD could describe 8 different representations of a video, with each representation corresponding to the video encoded at different bitrates. *See, e.g., id.*, 2:26-30. A client device would use the MPD to select an appropriate representation based on network conditions. If the network was congested or if the client device was on a low-capacity link, then the client device would use the MPD to select a low bitrate representation. *See, e.g., id.*, 2:30-32. If the network was uncongested and the client device was on a high-capacity link, then the client device would use the MPD to select a high bitrate representation. *See, e.g., id.*, 2:32-34. As network conditions changed, the client device adapted to the network conditions by selecting higher bitrate representations or lower bitrate representations. *See, e.g., id.*, 2:36-38. Thus, the client device adaptively streamed

the video by selecting appropriate bitrate representations based on network conditions.

72. Video streams were typically stored in media container files on a server.¹ Several media container formats existed, including the ISO base media file format, sometimes referred to as the MP4 file format (“.mp4” files), and Matroska (“.mkv” files). *See, e.g.*, Ex. 1010, 2:59-4:5. MPEG is a working group of ISO/IEC that produces standards in audio and video encoding, and the ISO base media file format was standardized as ISO/IEC 14496-14. *See, e.g.*, Ex. 1001, 2:27-34. Container specifications generally defined the format of a file, including the location of audio and video data within the file, along with metadata for indexing and accessing specific portions of the audio/video content. Adaptive streaming systems utilized these indices to jump to specific portions of the audio/video content when switching between alternate video files.

73. In terms of network protocols, video streaming systems typically used HTTP (or HTTP with transport-layer security, referred to as HTTPS). One reason was due to the ubiquity and widespread support for HTTP. HTTP allows a client

¹ Video content was often accompanied by audio content, such as for movies and TV shows. By 2011, video and audio encoding were both well known in the art, and popular media container files allowed video and audio tracks to be stored in the same file. In this Declaration, I may sometimes refer to a “video stream” or “video file,” but a POSITA would have understood and found it obvious for a “video stream,” “video file,” or similar terms to include audio content as well.

to request documents from a server, based on a URL. For example, the URL “http://company.com/folder/content.html” would lead to a client issuing a GET request to the server at “company.com” for the file at “/folder/content.html.” Starting at least with HTTP version 1.1 in 1997, the HTTP protocol allowed requests for specific byte ranges of files using the content-range attribute in the request. For example, a client request for “http://company.com/video/movie.mkv” could include a specific byte range in the GET request, which allowed retrieval of that specific portion of the file without the need to download the entire file, which could be rather large. Such GET requests for specific portions of files were sometimes referred to as partial GET requests. A typical partial GET request would include information specifying a target URL as well as a content-range header that would indicate a desired byte range. *See, e.g.*, Ex. 1009, 2:44-3:3, 2:60-63. The byte-range request feature of HTTP was commonly used, along with file container indices, to transition between alternative video files in adaptive bitrate streaming systems.

74. For example, a popular standard for adaptive bitrate streaming was called MPEG-Dynamic Adaptive Streaming over HTTP (“MPEG-DASH”). MPEG-DASH is an international standard ratified by Moving Picture Experts Group (MPEG) and International Organization for Standardization (ISO). Drafts for MPEG-DASH were published for comment as early as 2010. By 2011, full

reports for MPEG-DASH had been circulated for voting. By 2012, a ratified standard for MPEG-DASH had been published. MPEG-DASH incorporated many features from existing adaptive streaming technologies, including Microsoft Smooth Streaming (MSS), Apple Live Streaming (ALS) and Adobe HTTP Dynamic Streaming (AHDS), and sought to incorporate the features from these adaptive streaming technologies into an integrated standard. *See, e.g.*, 1009, 2:1-25.

75. Prior art MPEG-DASH systems generally divided media content into segments encoded at different bitrates. *See, e.g.*, 1009, 2:11-43, 5:34-6:8. These segments would be stored on a server and downloaded by a client through HTTP GET requests, including partial GET requests as described above. *See, e.g., id.*, 2:44-3:3, 5:34-6:8, 6:33-7:39. The client would adapt to varying network conditions by selecting segments encoded at bitrates appropriate for the current network condition. *See, e.g., id.*, 2:11-43; Ex. 1010, 15:15-30. If network conditions were poor, the client would select segments encoded at lower bitrates, and if network conditions were good, the client would select segments encoded at higher bitrates. Lower bitrate segments would provide media encoded at a lower quality and require less data. Higher bitrate segments would provide media encoded at a higher quality and require more data. By selecting segments encoded

at lower or higher bitrates, the client would adapt to available bandwidth and deliver the media content without interruption.

76. Prior art MPEG-DASH systems also provided an MPD to describe the segments of the media content. *See, e.g.*, 1009, 2:11-43, 9:57-10:36. The MPD acted as a top level index that describes the different qualities and bitrates of the media content and provides information to access the segments associated with the different qualities and bitrates. *See, e.g., id.* The different qualities and bitrates of the media content are often referred to as Representations. *See, e.g., id.*, 5:34-63, 9:57-10:36. A typical implementation of an MPD would be an XML data interchange file that lists different Representations of media content and provides Uniform Resource Locator (URL) templates associated with each Representation. A client device would utilize the MPD and send appropriate HTTP Get requests based on the URL templates to access specific portions of each Representation, effectively streaming media content while adapting to network conditions.

77. Segments can be further divided into subsegments or fragments. *See, e.g.*, 1009, 7:56-8:5.

78. Prior art MPEG-DASH systems typically described fragments using a segment index. *See, e.g.*, 1009, 6:33-47, 33:29-37. A typical segment index would provide various combinations of time ranges, time offsets, byte positions, byte offsets and byte ranges. A client would utilize the segment index as well as URL

templates provided by an MPD to generate partial GET requests to obtain specific portions of a selected Representation.

B. DRM

79. By 2011, DRM had long been used to protect content, particularly audio and video content, from unauthorized copying and use. For example, DRM systems typically encrypt content, so that only authorized users with the proper keys could decrypt and use (e.g., view) the content. DRM systems included mechanisms for granting, revoking and generally managing user rights, hence the name “digital rights management.” These were typically implemented using common cryptographic tools and protocols, including the use of certificates, signatures and keys. Most DRM systems included functionality for encryption/decryption; key management, which tracked the keys associated with encrypted media files and provided them to authorized client devices; access management; and authentication of client and server devices.

80. Most DRM systems used off-the-shelf encryption tools, which had largely been standardized and were nearly universally available in open source software by 2011. Even by the early 2000s, encryption algorithms had largely become standardized, with the most popular being the Advanced Encryption Standard (“AES”). DRM systems generally used standardized encryption algorithms such as AES to encrypt content.

1. Key Management

81. A common DRM approach was to encrypt each video (e.g., movie) with unique keys. This allowed a greater degree of control because an authorized user with the keys for one piece of content (e.g., one movie) can decrypt the streams for that content, but cannot use those keys for any other content. For prior art MPEG-DASH systems, it was common to use a single key or a single set of keys to encrypt all the alternative streams for a video because it simplified key management, given that permission to adaptively stream a video generally included permission to access the alternative representations of that video in its manifest file. For example, when a client began streaming a movie, the client was expected to switch between the alternative streams of that movie depending on the available bandwidth—this was the basic premise of adaptive bitrate streaming. *See supra* Section IV.A. Because access was typically granted to alternative streams as a group, it was common to encrypt them as a group, using the same key, which simplified key management by allowing the service to send over that one key, which the client would then use to decrypt the alternative streams. *See Ex. 1008, 18:62-67.* When a set of keys was used to encrypt a stream, for example by cycling through them, it was common to use that same set of keys to encrypt the alternative streams. *See, e.g., Ex. 1013, 7:15-52, Fig. 9.*

82. To manage keys, it was common to associate each key with a unique identifier called a “key ID” that could be used to reference a particular key. *See, e.g.*, Ex. 1014, 1:43-52 (“... a data deciphering section 24 for deciphering the separated enciphered data by means of a cipher key ks specified by the cipher key number N(ks).”), Figs. 4, 6, 17; Ex. 1013, 9:34-39 (“In the embodiment of FIG. 9, the frame decryption information 995 identifies, with respect to each frame of encrypted video stream 950, the frame number, the status of encryption (on or off), the offset length, the number of bytes encrypted, and a reference to the applicable encryption key.”), Fig. 9. This allowed systems to identify which keys were used to encrypt which files, thereby enabling the decryption of those files using the identified key. Since video was typically encrypted using symmetric key encryption, the same key used to encrypt data was needed to decrypt it. *See infra* Section IV.B.2. In the context of video streaming, it was common to store key IDs in encrypted video files to identify the key to use to decrypt the video content. Keys were often stored in key tables alongside their key IDs, so they could be easily organized and accessed. The use of key IDs allowed clients to identify, obtain, and verify and ultimately use the decryption key associated with the video.

2. Partial Encryption

83. Partial encryption was known and used since the 1990s. By the 2000s, partial encryption was a common DRM feature for video streaming. With

partial encryption, which was also referred to as selective encryption, only portions of a video stream were encrypted, leaving the remaining portions unencrypted. *See* Ex. 1012, 23:17-24:49. For example, many partial encryption techniques encrypted a portion of frames while other portions remained were unencrypted. *See id.* Various partial encryption techniques were widely known and utilized depending on various parameters, including security and computational complexity.

84. Partial encryption sought to find the right balance between security and computational load. Some applications, such as private video conferencing, needed higher security and thus encrypted a larger percentage of the video stream. Other applications, such as movie distribution, often tolerated some amount of information being accessible because the video was sufficiently degraded to inhibit piracy. Whether a larger percentage or a smaller percentage of a video stream is encrypted, partial encryption also considers what portion or portions of the video stream to encrypt. In many applications, the portion of the frame to encrypt should be selected to ensure that it is difficult to guess so as to disrupt a decoder without proper access. *See, e.g.,* Ex. 1007, ¶[0061]. The portions that are encrypted in a video stream can be identified by offset information stored in metadata associated with the video stream. *See id.*

V. THE '588 PATENT

A. Background Art Admitted by the '588 Patent

85. The '588 patent issued from an application filed in 2018. It generally “relates to adaptive bitrate streaming of encoded media,” but the '588 patent does not claim to have invented adaptive bitrate streaming. Ex. 1001, 1:39-40. The '588 patent admits that adaptive bitrate streaming was known, including streaming media “where the media is stored on a server and continuously sent to the playback device over a network during playback.” *Id.* 1:45-48. The '588 patent further admits that it was known that in adaptive bitrate streaming, “content is divided into multiple streams” and “some streams can be encoded as alternative streams that are suitable for different network connection bandwidths.” *Id.* 1:53-57.

86. The '588 patent also admits that it was known that adaptive bitrate streaming “involves detecting the present streaming conditions (e.g. the user’s network bandwidth and CPU capacity) in real time and adjusting the quality of the streamed media accordingly by selecting between different streams encoded for use at different network connection data rates.” *Id.* 1:59-64. “Typically, the source media is encoded at multiple bitrates and the playback device or client switches between streaming the different encodings depending on available resources.” *Id.*, 1:64-67.

87. The '588 patent admits that known “[a]daptive streaming solutions typically utilize... Hypertext Transfer Protocol (HTTP)... to stream media between a server and [the] playback device.” *Id.* 2:1-5. The '588 patent admits that it was known that “HTTP is a stateless protocol that enables a playback device to request a byte range within a file” such as a container file. *Id.* 2:5-7.

88. The '588 patent admits that it was known that in adaptive bitrate streaming systems, “the source media is typically stored on a media server as a top level index file pointing to a number of alternate streams.” Ex. 1001, 2:12-14. The '588 patent further admits it was known that “[e]ach stream is typically stored in one or more container files.” *Id.*, 2:14-15.

89. Thus, the '588 patent admits that in adaptive bitrate streaming, it was known that a playback device would obtain a top level index file identifying alternative streams of video (e.g., requesting a top level index file pointing to a number of alternate streams) and request portions of a selected stream with byte range requests (e.g., request a byte range within a container file using HTTP).

90. The '588 patent admits that “several adaptive streaming solutions” are known in the art, including IIS Smooth Streaming, Flash Dynamic Streaming, and HTTP Adaptive Bitrate Streaming. *See* Ex. 1001, 2:17-25. The '588 patent admits that IIS Smooth Streaming teaches utilizing “Synchronized Multimedia Integration Language (SMIL)... to create indexes,” including a top level index file. *Id.* 2:17-

23. The '588 patent further admits that IIS Smooth Streaming teaches using the MP4 container format, which was one of “[t]he most commonly used media container formats,” to store alternate streams of video. *Id.* 2:27-34. The '588 patent admits that Flash Dynamic Streaming also teaches utilizing SMIL indexes to create indexes such as a top level index file and utilizing the MP4 container format to store alternate streams of video. *See id.* 2:17-34.

91. The '588 patent admits that HTTP Adaptive Bitrate Streaming teaches utilizing “an extended M3U playlist file (.M3U8), which is a text file containing a list of URIs that typically identify a media container file” to create indexes, including a top level index file. *Id.* 2:23-27. Thus, the '588 patent admits that several existing adaptive bitrate streaming solutions teach the use of indexes, including top level index files, and container files, to store alternate streams of video, in adaptive bitrate streaming.

92. The '588 patent does not purport to have invented encryption or Digital Rights Management (“DRM”). The '588 patent admits that both were known in the art, where DRM technologies “typically involve the use of cryptographic information to control access to or protect a piece of content.” Ex. 1001, 2:50-52. The '588 patent admits that known methods of encrypting media or content included “using cryptographic information such as (but not limited to) one or more encryption keys to encrypt some or all of the content.” *Id.* 2:52-54. The

'588 patent admits that it was known that in “most DRM systems [there] is a mechanism for securely providing cryptographic information to authorized playback devices for use in accessing protected content.” *Id.* 2:54-57. Thus, the '588 patent admits that it was known for content, including video, to be protected through encryption, including partial encryption (e.g., using cryptographic information to protect content and using cryptographic information to encrypt some of the content) and for a playback device to obtain encryption information, including encryption keys (e.g., an authorized playback device being securely provided with cryptographic information), decrypting the protected content, and playing back the decrypted content (e.g. using the cryptographic information to access protected content).

93. As described above, the '588 patent admits that media container files, including the MP4 container format and the MPEG transport stream container, were known in the art. The '588 further admits that the Matroska container format was known and used in the art. The '588 patent admits that the Matroska container format was “a media container developed as an open standard project by the Matroska non-profit organization.” Ex. 1001, 2:35-37. The '588 patent also admits that it was known that “the Matroska container is supported by many consumer electronics (CE) devices.” *Id.* 2:40-41. The '588 patent admits that the

DivX Plus file format “utilizes an extension of the Matroska container format” and “is based upon the Matroska container format.” *Id.* 2:41-45.

B. Summary of the Alleged Invention

94. The ’588 patent claims recite various details of adaptive bitrate streaming and partial encryption, but its specification does not clearly identify what the alleged invention is purported to be. The “Summary of the Invention” is a recitation of embodiments that spans 5 columns. *See* Ex. 1001, cols. 2-7. The independent claims recite adaptive streaming features, such as a top level index file identifying a plurality of alternative streams of protected video, selecting a stream based on detected streaming conditions, and requesting portions of the selected stream based on the byte ranges, which the ’588 patent admits was known in the art. *See supra* Section V.A; Ex. 1001, 1:45-2:62.

95. The independent claims also recite basic DRM features, including partial encryption and encrypting the alternative streams using a set of common keys comprising at least one key. *See* Ex. 1001, claim 1. The ’588 patent specification is sparse with details regarding whether or how its use of a set of common keys differed from what was already widely known and practiced in the art. The specification does not offer any details regarding how its use of common cryptographic information or set of common keys distinguishes it from well-known encryption and key management techniques.

C. The Alleged Invention According to Patent Owner

96. According to PO, the '588 patent is directed to using “common frame encryption keys to encode alternate video streams.” Ex. 1015, ¶224. According to PO, the '588 patent recites use of “a new file type” and “byte ranges” to “provide the playback device with newly available computing resources.” Ex. 1015, ¶232. No other alleged point of novelty is explained in DivX’s complaint.

D. Priority Date

97. The '588 patent is a continuation of application 14/970,412, which is a continuation of application 14/564,003, which is a continuation of application 13/340,623, filed on December 29, 2011, which claims priority to provisional application 61/530,305, filed on September 1, 2011. *See* Ex. 1001. The claims recited in the '588 patent are unsupported by the provisional application. For example, the provisional application does not support the claim limitations regarding “receiving a container index that provides byte ranges.” *See* Ex. 1005. Thus, the priority date of the '588 patent is December 29, 2011 at the earliest. Regardless of whether the priority date is December 29, 2011 or September 1, 2011, the patents and publications cited in the grounds below constitute prior art to the '588 patent. My opinions presented in this Declaration would not materially change if the '588 patent were entitled to a priority date of December 29, 2011 versus September 1, 2011.

E. Challenged Claims

98. I understand that Petitioner is challenging the validity of claims 1-24 of the '588 patent in the Petition for Inter Partes Review to which this declaration will be attached. Those claims are reproduced in Appendix 3. This Petition and declaration are directed to the challenged claims, which are all claims 1-24 of the '588 patent. I have considered all claims 1-24 of the '588 patent, as well as portions of the '588 patent prosecution history in forming my opinions.

99. I understand that Lindahl and Hurst were not cited or considered during prosecution of the '588 patent because none of these references or any related patents are listed on the face of the '588 patent. In addition, I am not aware of Chen, Lindahl or Hurst being discussed in any office action.

VI. CLAIM CONSTRUCTION

100. For purposes of this inter partes review, I have considered the claim language, specification, and portions of the prosecution history, to determine the meaning of the claim language as it would have been understood by a person of ordinary skill in the art at the time of the invention. The “plain and ordinary meaning” or Philips standard has traditionally been applied in district court litigation, where a claim term is given its plain and ordinary meaning in view of the specification from the view point of a person of ordinary skill in the art.

101. I have applied the Philips standard in my analysis. Unless otherwise stated, I have applied the plain and ordinary meaning to claim terms. I have not formed and do not offer an opinion on the indefiniteness of the claims. While there may be issues that prevent a POSITA from determining the scope of the claims with reasonable certainty, my opinions here rely on prior art that falls squarely within the scope of the challenged claims of the '588 patent, nowhere near the boundaries of the claims. Therefore it is not necessary for determining the invalidity of the challenged claims to resolve whether the '588 patent informs a POSITA of the scope and boundaries of its claims with reasonable certainty, or if the claims are indefinite for other reasons. Thus I have not offered an opinion regarding the indefiniteness of the claims.

VII. SUMMARY OF THE PRIOR ART

102. There are a number of patents and publications which constitute prior art to the '588 patent. I have reviewed and considered the prior art discussed in this section, along with the materials listed in Appendix 2.

A. Invalidity Grounds

103. Based on my review and analysis of the materials cited herein, my opinions regarding the understanding of a POSITA in the 2010-2011 timeframe, and my training and experience, it is my opinion that the challenged claims of the '588 patent are invalid based on the following grounds:

Grounds	Claims	Statutory Basis	Prior Art
1	1-24	§ 103	Chen in View of Lindahl and Hurst

The reasons for my conclusions are explained more fully below.

104. I understand that Lindahl and Hurst were neither cited nor considered during prosecution of the '588 patent. Neither patent or any related patents are listed on the face of the '588 patent, nor did I see any discussion of them in the prosecution history. Chen was included in an information disclosure statement (“IDS”) alongside over 1,000 other references, in seven separate IDSs filed the same day, spanning 47 pages. *See* Ex. 1002, 102-107, 390-392, 405-407, 415-420, 433-454, 467-472, 485-488. Chen was never specifically discussed by the examiner or raised in any office actions that I reviewed. I reviewed the file history for the '588 patent and am not aware of any of the above references being discussed in any office action or in the prosecution history generally.

B. U.S. Patent Pub. No. 2011/0096828 (“Chen”) (Exhibit 1006)

105. Chen was filed on September 21, 2010, claims priority to September 22, 2009 and published April 28, 2011, which is before the priority date of the '588 patent. Therefore, Chen is prior art under at least pre-AIA §§102(a) and (e).

106. Chen teaches HTTP streaming involving “stream segmentation and the use of multiple streams.” Ex. 1006, Abstract, ¶[0063]. The multiple streams involve video content “encoded at multiple bitrates to form different versions, or

representations.” *Id.* These representations are “broken into smaller pieces, perhaps on the order of a few seconds each, to form segments.” *Id.* These representations and segments are stored on a server or content delivery network and streamed to a client device. *See, id.* ¶[0061-0063]. Chen teaches that “[t]he 3GPP File Format or any other file based on the ISO base media file format, such as the MP4 file format or the 3GPP2 file format, may be used as the container format” for these purposes. *Id.* ¶[0119].

107. A client device that requests “individual segments that are seamlessly spliced together by the client.” Ex. 1006, ¶[0064]. As the client device requests segments, it “switch[es] to different data rates based on available bandwidth.” *Id.* A media presentation description (“MPD”) file identifies the alternative representations of a media item, providing their respective bitrates, URLs, and other information, which the client uses to request an appropriate segment based on the available bandwidth. *See id.*, ¶¶[0063-0064, 0066-0068]. These requests “for files or segments” take the “form of HTTP requests that include a URL, and may also include a byte range, in order to request less than all of the file or segment indicated by the URL.” *Id.*, ¶[0111]. By requesting less than all of the file or segment, bandwidth utilization is more efficient.

C. U.S. Patent Pub. No. 2007/0083467 (“Lindahl”) (Exhibit 1007)

108. Lindahl was filed on October 10, 2005 and published on April 12, 2007, which is more than a year before the priority date of the ’588 patent. Therefore, Lindahl is prior art under pre-AIA §§102(a), (b), and (e).

109. Lindahl teaches DRM for media files, including techniques for partially encrypting portions of media files. Ex. 1007, Abstract. While other parts of the file remain in the clear, the partial encryption “substantially disrupt[s] usefulness of other parts of the media file which are not encrypted.” *See id.* Lindahl also teaches that decryption of a partially encrypted media file is “likewise also only partial, which makes decryption faster and less resource intensive.” *Id.* Lindahl also teaches several techniques for managing the keys used to encrypt media files. Lindahl’s teachings of partial encryption and key management schemes reflect widely-known techniques for video DRM.

D. U.S. Patent 8,683,066 (“Hurst”) (Exhibit 1008)

110. Hurst was filed on August 6, 2007, published on February 12, 2009 and granted on March 25, 2014. Hurst was filed and published more than a year before the priority date of the ’588 patent. Therefore, Hurst is prior art under pre-AIA §§102(a), (b) and (e).

111. Hurst teaches “adaptive-bitrate content streaming” with “multi-bitrate streamlet information.” Ex. 1008, Abstract, 1:6-10, 16:61-17:58. Hurst teaches

that content is encoded as “a plurality of streams having varying degrees of quality and bandwidth” *Id.*, 3:65-68, FIG. 2b The streams are divided into “streamlets.” *Id.*, 6:59-62. Hurst teaches that “streamlets 304 of different bitrates may be encrypted with the same key.” *Id.*, 18:64-67.

VIII. INVALIDITY

A. Ground 1: Chen in View of Lindahl and Hurst

112. It is my opinion that a POSITA would have found Claims 1-24 obvious based on the teachings of Chen in view of Lindahl and Hurst.

1. Motivation to Combine

113. A POSITA would have been motivated to apply teachings from Lindahl and Hurst to Chen. Chen teaches common adaptive streaming techniques that were widely known in the art, including using multiple representations of content, using byte range indexes, detecting network conditions, and selecting between the multiple representations of content accordingly. Lindahl teaches DRM techniques that were widely known in the art, including partial encryption and key management techniques. Hurst provides related DRM teachings for adaptive streaming, including teaching the use of a common key for alternative streams, which was widely known in the art. A POSITA would have been motivated to combine (a) Chen’s adaptive streaming teachings with (b) Lindahl’s DRM teachings, and (c) Hurst’s DRM teachings, as explained below.

114. First, adaptive bitrate streaming and DRM techniques, such as those taught by Chen, Lindahl, and Hurst, were commonly used together for video streaming, were widely recognized as complementary techniques, and would have been used here for their known and conventional purposes. *See, e.g.*, Ex. 1010, Abstract. Indeed, the use of DRM (protecting content intellectual property) and adaptive streaming (addressing network instability) together were understood by the technical community to be two complementary requirements for paid streaming services well before the priority date of the '588 patent. Chen, Lindahl, and Hurst are all directed at delivering content over the Internet. *See, e.g.*, Ex. 1006, ¶¶[0103-0104]; Ex. 1007, ¶¶[0036, 0039]; Ex. 1008, 3:12-23, 6:6-58. By 2011, it was widely known for video streaming to include adaptive bitrate streaming, to account for bandwidth variability over the Internet, and DRM, to address piracy and related concerns. *See id.* Both Chen and Lindahl provide motivations to combine their teachings in this manner. And Hurst specifically teaches DRM for adaptive streaming. *See, e.g.*, Ex. 1008, Abstract (“An apparatus, system, and method for maintaining a programming lineup of **adaptive-bitrate content streaming** is provided. ... The media content may comprise a plurality of streamlets. The apparatus also includes at least one data module configured to maintain multi-bitrate streamlet information.”), 1:6-10 (“The invention relates to video streaming over packet switched networks such as the Internet, and more

particularly relates to maintaining a programming lineup of adaptive-rate shifting of streaming content over such networks.”), 8:16-18, 16:61-17:58, Figure 10.

Thus, a POSITA would have understood that Hurst’s DRM teachings were complementary to the combination of Chen and Lindahl, and a POSITA would have been motivated to combine Chen with Lindahl and Hurst to enhance the efficiency and security of DRM for video streaming.

115. While Chen accounted for the use of DRM in its teachings (*see* Ex. 1006, ¶[0522] (“... because encryption is being employed to encrypt each fragment or block separately, and thus the entire fragment or block needs to be received before it can be decrypted.”)), Chen did not require any particular DRM implementation, leaving particulars such as encryption and key management to the POSITA’s knowledge and preferences. Lindahl and Hurst provided such DRM teachings. For example, Lindahl’s teachings prevented unauthorized access to media, and a POSITA would have been motivated to apply those teachings for streaming video over the Internet because piracy was a concern for content providers, given the industry’s previous experiences with music and video file sharing. *See, e.g.*, Ex. 1007, ¶[0095] (“The advantages of the invention are numerous. Different aspects, embodiments or implementations may yield one or more of the following advantages. One advantage of the invention is that digital media data can be protected from unauthorized usage through encryption of only

parts of the digital media data. Another advantage of the invention is that portable, battery-powered media devices can perform decryption in a power efficient manner since the computational burden to decrypt encrypted media files is substantially decreased through use of partial encryption. Still another advantage of the invention is that portable media devices can yield higher quality media output.”).

116. Similarly, Hurst provides DRM protection, and its teachings further specify the keys to be used for encrypting content for adaptive streaming. Hurst, like Chen, teaches encoding a video at different bitrates. *See id.*; Ex. 1006, ¶¶[0063-0064, 0080-0081]. A POSITA would have been motivated to apply Hurst’s teachings, e.g., where the alternative streams are encoded at different bitrates and encrypted as a group in the same manner, such as by using the same key, because they follow the usage model for video distribution in adaptive streaming, where the playback device gains access to the alternative streams as a group. *See* Ex. 1006, ¶¶[0063-0064]; *supra* Section IV.B; VIII.A.2[1e]. For example, a playback device adaptively streaming a movie is typically granted access to its alternative streams for the movie because it is expected to switch between them depending on channel bandwidth. Since the alternative streams are accessed as a group, it would have been obvious to encrypt them as a group, using the same set of keys, so that the alternative streams can be accessed as a group

using that set of keys. *See id.* Thus Hurst teaches the natural and obvious approach to DRM for adaptive streaming, which was well-known in the art before Hurst or the '588 patent was filed, and a POSITA would have been motivated to apply Hurst's DRM teachings to the combination of Chen and Lindahl.

117. Second, the prior art provides an express teaching and motivation to combine adaptive bitrate streaming, which Chen teaches, with partial encryption, which Lindahl teaches. *See* Ex. 1010 ("Chen-924"), Abstract ("Various embodiments are directed towards employing a container and communication protocol proxy component within a client, device to receive securely real-time streamed, progressively downloaded, or **adaptively streamed container over a network** using one container and communication protocol, and to securely decrypt the container and provide it to a media player using a different container and communications protocol... The container may be received as **selectively encrypted container.**") (emphasis added). Chen-924 teaches several embodiments that adaptively stream alternative representations of a video using selectively encrypted (i.e., partially encrypted) container files. *See, e.g.*, Ex. 1010, 14:41-15:60. A POSITA would have understood these teachings of adaptively streaming a selectively encrypted container as motivation to combine adaptive bitrate streaming techniques with selective encryption techniques to "securely and progressively download streaming digital media." Ex. 1010, 1:23-26. Thus a

POSITA would have been motivated to combine Chen and Lindahl based on the teachings of Chen-924.

118. Furthermore, Lindahl provides an express motivation to combine its DRM teachings with known streaming techniques, providing an illustrative example of purchasing and streaming a media file. *See* Ex. 1007, ¶[0065] (“Next, the media file is downloaded 708 to the user. In one embodiment, the download 708 of the media file can be performed by streaming the media file through the data network to the user, namely, to the user's client machine. Here, the downloading 708 of the media file can be efficient because the manner in which the media file can be stored facilitates its immediate download without the need for processing-intensive conversion.”). Given the widespread use of adaptive bitrate streaming, and the widely-recognized need to account for bandwidth variations on the Internet (which were particularly challenging to content providers in 2011), a POSITA would have been motivated to combine Lindahl’s DRM teachings with Chen’s teachings regarding adaptive bitrate streaming.

119. Third, the teachings from each reference improve efficiency and provide other benefits. Chen notes that efficiency concerns include efficient utilization of available bandwidth and efficient video playback to minimize decoding latency at the client. *See* Ex. 1006, ¶¶[0024], [0026] (“Other concerns, such as being able to decode the source blocks fast enough to keep up with the

source streaming rate, to minimize the decoding latency and to reduce the use of available CPU resources are issues.”). In addressing these concerns, a POSITA would have been motivated to apply Chen’s adaptive bitrate streaming teachings to provide video data while accounting for variable network conditions and efficiently utilizing available bandwidth, for example by allowing clients to select representations, encoded at different bitrates, appropriate for given network conditions and bandwidth availability. *See* Ex. 1006, ¶¶[0063-0065, 0080-0081]. This improved efficiency by allowing the highest bitrate representation that could be provided without interrupting the video stream for the given network conditions. *See id.* Thus a POSITA would have been motivated to apply Chen’s adaptive bitrate streaming teachings because, for example, they improved bandwidth efficiency, improved media presentation, improved user experience, provided fast startup and seek, and minimized unnecessary delivery. *See* Ex. 1006, ¶[0065] (“Advantages of HTTP streaming may include bit-rate adaptation, fast startup and seek, and minimal unnecessary delivery. These advantages come from controlling the delivery to be only a short time ahead of the playout, making maximum use of available bandwidth (through variable bit rate media), and optimizing stream segmentation and intelligent client procedures..”), ¶[0027]. Therefore, Chen’s adaptive bitrate streaming teachings provided for improved media streaming by optimizing the balance between bitrate and bandwidth.

120. Additionally, as was known in the art, Chen's adaptive bitrate streaming teachings provided for an index of byte ranges that facilitated requests for specific portions of a video stream. *See, e.g.*, Ex. 1006, ¶¶[0063-0064, 0086, 0111, 0119-0120, 0123, 0129, 0148, 0157-0158, 0175-0187, 0222]. This allowed a playback device to start a video stream at a specific portion, which was common when adaptively switching between streams due to network conditions, and allowed the playback device to seek to other portions of the video stream as desired by the user. *See, e.g.*, Ex. 1006, ¶[0086]. Therefore, adaptive bitrate streaming as taught by Chen provided fast startup and seek, and it facilitated adaptive bitrate streaming. Further, because Chen's adaptive bitrate streaming teachings facilitated requests for specific portions of a video stream, unnecessary portions of the video stream did not have to be delivered, further increasing bandwidth efficiency. *See* Ex. 1006, ¶[0065]. The teachings of Chen relied on in this Declaration reflect common, well-known aspects of adaptive streaming, and a POSITA would have been motivated to apply Chen's teachings in the manner explained in this Declaration, to achieve the efficiency, optimization, and user-experience benefits explained above. *See* Ex. 1006, ¶¶[0024], [0026]-[0027], [0063-0065], [0148].

121. Lindahl's DRM teachings would further have improved computational efficiency, and therefore a POSITA would have been motivated to

apply Lindahl’s DRM teachings—including its partial encryption and key management techniques—to address Chen’s concerns by improving video playback efficiency and reducing decoding latency at the client. *See* Ex. 1007, ¶[0095] (“Another advantage of the invention is that portable, battery-powered media devices can perform decryption in a power efficient manner since the computational burden to decrypt encrypted media files is substantially decreased through use of partial encryption.”). A POSITA would have understood that the distributed nature of the Internet was widely known to cause bandwidth and latency variation, and that the wide range of potential client devices on the Internet required video services to account for a wide range of computational resources, including portable devices with limited CPU power. *See id.*

122. Lindahl’s partial encryption teachings reduce computational load for encryption and decryption by reducing the amount of data that is encrypted and decrypted, while maintaining content protection by encrypting sufficient amounts of video data to render it degraded and effectively unusable to unauthorized users. *See* Ex. 1007, ¶[0095]. These performance enhancements enabled the use of a wide range of computing devices, including emerging mobile devices for streaming. Thus, a POSITA would have been motivated to apply these teachings because they provided a balanced level of security that was well-suited for video streaming applications. Other aspects of Lindahl’s DRM teachings, including its

key management teachings, provided additional security, simplicity and efficiency benefits that would have provided additional motivation to apply those teachings. *See infra*, Section VIII.A.2.[1e].

123. Additionally, I note that Lindahl is an Apple Corporation patent. As the '588 patent admits, Apple was active in developing and using prior art file formats and adaptive bitrate streaming. *See* Ex. 1001, 2:23-27 (“HTTP Adaptive Bitrate Streaming developed by Apple Computer Incorporated of Cupertino, Calif. implements index files using an extended M3U playlist file (.M3U8), which is a text file containing a list of URIs that typically identify a media container file.”). Thus a POSITA using Chen’s adaptive bitrate streaming teachings would have looked to Lindahl and been motivated to apply its teachings for the reasons explained above.

124. Further, the references provide express motivation to modify their teachings, which a POSITA would have been motivated to do by combining their teachings. *See* Ex. 1006, ¶[0554] (“The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense. It will, however, be evident that various modifications and changes may be made thereunto without departing from the broader spirit and scope of the invention as set forth in the claims and that the invention is intended to cover all modifications and equivalents within the scope of the following claims.”); Ex. 1007, ¶[0096]

(“Further, since numerous modifications and changes will readily occur to those skilled in the art, the invention should not be limited to the exact construction and operation as illustrated and described. Hence, all suitable modifications and equivalents may be resorted to as falling within the scope of the invention.”); Ex. 1008, 21:20-27.

125. While Hurst discloses other embodiments, a POSITA would have been motivated to apply Hurst’s teaching of encrypting the alternative streams in the same manner, for the reasons explained above. *See* Ex. 1008, 18:66-67. For example, Hurst discloses an embodiment involving standard and premium streams that are encrypted with different keys. *See* Ex. 1008, 19:1-8. Hurst describes these embodiments simply as separate options representing different usage models for content streaming. *See id.*, 18:66-9:8. My opinions regarding the combination of Chen, Lindahl, and Hurst do not rely on this other embodiment or a service that offers both standard and premium versions of content. Thus the disclosure of a separate embodiment would not have deterred a POSITA from applying Hurst’s first teaching, of encrypting the alternative streams in the same manner, because the first teaching was well-known in the art, offered several advantages, and was aligned with the usage model for adaptive streaming that was taught by Chen, where the alternative streams are accessed as a group based on streaming conditions. *See supra* Section VIII.A.[1h-1i].

126. **Reasonable Expectation of Success.** A POSITA would have had a reasonable expectation of success in combining the combined teachings because they were widely known in the art—even before the respective patents were filed—and were widely recognized as complementary and compatible techniques. Chen’s adaptive bitrate streaming teachings and Lindahl’s and Hurst’s DRM teachings would have each been used for their known purposes, and thus a POSITA would have been familiar with those techniques. By 2011, video streaming systems commonly (and almost universally) used adaptive bitrate streaming and DRM (including partial encryption and key management) together. For example, a video streaming system would encode multiple representations of a video at different bitrates, as taught by Chen, and the multiple representations would be partially encrypted, per Lindahl’s DRM teachings, using common keys, per Hurst’s teachings. Each aspect of the combination would be used in a conventional manner according to its known purpose, with adaptive streaming allowing the client to select alternative streams at different bitrates, and DRM providing encryption and key management functions. Thus, a POSITA would have recognized these as teachings to be complementary and compatible teachings that were intended to be used together, and in fact commonly used together for video streaming.

127. Furthermore, Chen and Hurst both apply their teachings to adaptive streaming frameworks such as MPEG-DASH. A POSITA would have recognized that Lindahl's partial encryption teachings were fully compatible with Hurst's DRM teachings because the former was directed to which portions of the video stream are encrypted, while the latter was directed to the separate issue of which key to use for encryption. Thus the combination of Chen, Lindahl and Hurst was easily within the level of ordinary skill for a POSITA.

128. **KSR Factors.** The alleged invention merely recites different aspects of video streaming that were already known and used together. The challenged claims are simply an obvious combination of prior art elements (e.g., adaptive streaming and partially encrypting alternative streams with the same set of keys) according to known methods (using Chen's adaptive bitrate streaming teachings to provide content to users, using Lindahl and Hurst's DRM teachings to encrypt content) to yield predictable results (a streaming service that accounts for network variability and protects access to content). The combination applies known techniques (e.g., Chen's adaptive streaming teachings, and Lindahl and Hurst's DRM teachings) to improve similar devices (e.g., streaming systems) in the same way as they were known and used in the prior art. For all of the reasons discussed above, a POSITA would have been motivated to combine Chen, Lindahl and Hurst.

2. Claim 1

1[a]. A playback device for playing protected content from a plurality of alternative streams, comprising:

129. The combination teaches or suggests a playback device (e.g., client device) for playing protected (e.g., partially encrypted) content from a plurality of alternative streams (e.g., representations). *See, e.g.*, Ex. 1006, ¶¶[0102, 0110, 0114-0118], FIG. 4, Ex. 1007, ¶¶[0017, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084, 0088-0090], FIG. 11. Chen and Lindahl teach a client device, at the consumer/user, for playing at least one of multiple representations of partially encrypted content. *See, e.g.*, Ex. 1006, ¶¶[0061] (“As described herein, a goal of a streaming system is to move media from its storage location (or the location where it is being generated) to a location where it is being consumed, i.e., presented to a user or otherwise ‘used up’ by a human or electronic consumer. Ideally, the streaming system can provide uninterrupted playback (or more generally, uninterrupted ‘consumption’) at a receiving end and can begin playing a stream or a collection of streams shortly after a user has requested the stream or streams.”), [0203] (“For example, referring again to FIG. 6 and the example described above where the client wants to start play out at approximately 22 seconds into the presentation, the client may request, through one or more requests, the data within block 4, and then feed this into media player as soon as it is available to start playback. Thus, in this example, the payout begins as soon as

the 42,011 bytes of block 4 are received at the client, thus enabling a fast channel zapping time. If instead the client needed to request the entire GoP before playout was to commence, the channel zapping time would be longer, as this is 144,211 bytes of data.”), [0366] (“Then, the client selects one or multiple representations based on the information in the representation attributes and other information, e.g., available bandwidth and client capabilities. Depending on grouping representations may be presented standalone or jointly with other representations.”); Ex. 1007, ¶[0011] (“In one embodiment, the partial encryption is performed by a host device and thereafter the partially encrypted media file is delivered to a client device. The client device, if authorized, can subsequently decrypt the partially encrypted media file and utilize (e.g., play) the media file. However, since the media files are only partially encrypted, the decryption is likewise also only partial, which makes decryption faster and less resource intensive.”).

130. Hurst refers to its client device as media player and teaches that it is for playing protected content from a plurality of alternative streams. *See* Ex. 1008, 1:15-19 (“Many portable media players are equipped with network connections and enabled to play music or videos. The music or video files (hereinafter ‘media files’) can be ... streamed or downloaded from a server.”), 6:13-17 (“The content module 112 may be configured to receive, process, and store content. In one

embodiment, the processed content is received by a client module 114 configured to play the content on the end user station 104.”), 8:47-49 (“In certain embodiments, once the streamlets 304 have been received and processed, the client module 114 may request streamlets 304 using HTTP from the web server 116.”). A POSITA would have recognized these teachings from Chen, Lindahl, and Hurst to refer to analogous concepts of client devices, which in this context were widely-known and widely-used types of playback devices. A POSITA would also have understood that a receiver as taught by Chen is synonymous with a client device. For example, Chen teaches that the two terms would be used interchangeably. *See, e.g.*, Ex. 1006, ¶[0201] (“the client or receiver”).

131. Given these teachings, it would have been obvious to a POSITA for the client device to play protected content from a plurality of alternative streams, as further explained with regard to the other claim limitations below.

[1b] a set of one or more processors; and a non-volatile storage containing an application for causing the set of one or more processors to perform the steps of:

132. The combination teaches or suggests a playback device (e.g., client device) comprising a set of one or more processors housed within a computer or device and a non-volatile storage (e.g., disk storage, ROM) containing an application (e.g., software, programs) for causing the set of one or more processors to perform the steps with respect to the challenged claims, as discussed herein.

See, e.g., Ex. 1006, ¶¶[0102, 0110, 0114-0118], FIG. 4, Ex. 1007, ¶¶[0088-90], FIG. 11. Chen and Lindahl both implement their teachings with client devices comprising a processor executing software written on disk storage or ROM. *See, e.g.*, Ex. 1006, ¶¶[0116]-[0118]:

As illustrated in FIG. 4, the client system may be implemented as a combination of hardware and software components, according to various embodiments. The client system may comprise a set of instructions that can be executed to cause the system to perform any one or more of the methodologies discussed herein. The system may be realized as a specific machine in the form of a computer. The system may be a server computer, a personal computer (PC), or any system capable of executing a set of instructions (sequential or otherwise) that specify actions to be taken by that system... The client system may include the client processor 402 (e.g., a central processing unit (CPU)), a memory 404 which may store program code during execution, and disk storage 406, all of which communicate with each other via a bus 400... The disk storage unit 406 may include a machine-readable medium on which may be stored one or more sets of instructions (e.g., software) embodying any one or more of the methodologies or functions described herein. The instructions may also reside, completely or at least partially, within the memory 404

and/or within the client processor 402 during execution thereof by the system, with the memory 404 and the client processor 402 also constituting machine-readable media.

See also *id.*, FIG. 4:

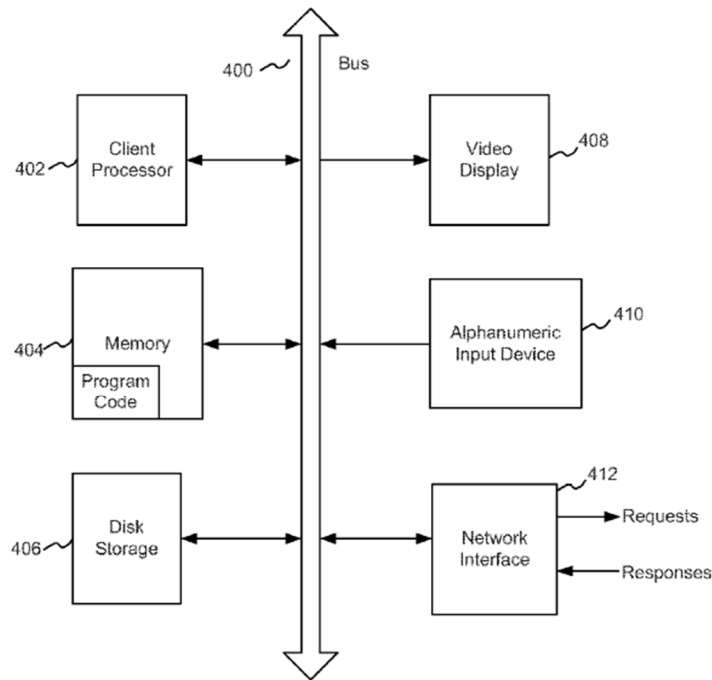


FIG. 4

See also Ex. 1007, ¶[0088] (“FIG. 11 is a block diagram of a media player 1100 according to one embodiment of the invention. The media player 1100 includes a processor 1102 that pertains to a microprocessor or controller for controlling the overall operation of the media player 1100... The media player 1100 also includes a RAM 1120 and a Read-Only Memory (ROM) 1122. **The ROM 1122 can store programs, utilities or processes to be executed in a non-volatile manner.** The

RAM 1120 provides volatile data storage, such as for the cache 1106.”) (emphasis added), FIG. 11:

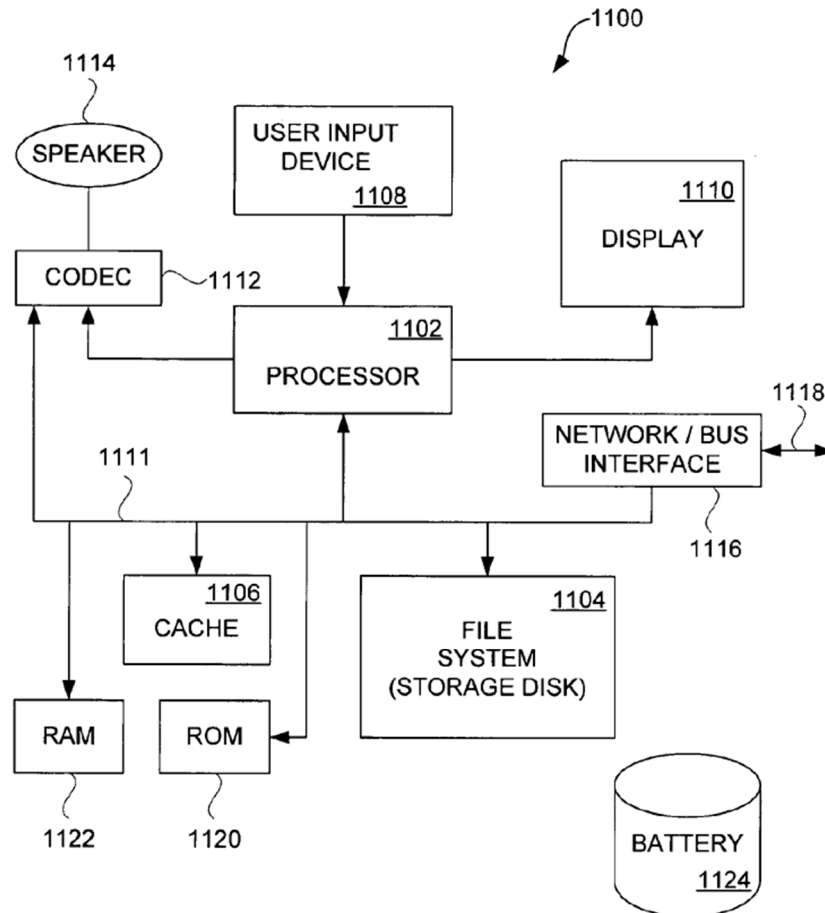


FIG. 11

133. Hurst further renders this limitation obvious by teaching a playback device comprising a set of one or more processors and a non-volatile storage containing an application for causing the set of one or more processors to perform the steps discussed herein. Hurst implements its teachings with computers (with processors) executing software programs stored in non-volatile storage. See Ex.

1008, 4:64-65 (“Modules may also be implemented in software for execution by various types of processors.”), 5:26-39:

Reference to a signal bearing medium may take any form capable of generating a signal, causing a signal to be generated, or causing execution of a program of machine-readable instructions on a digital processing apparatus. A signal bearing medium may be embodied by a transmission line, a compact disk, digital-video disk, a magnetic tape, a Bernoulli drive, a magnetic disk, a punch card, flash memory, integrated circuits, or other digital processing apparatus memory device. In one embodiment, a computer program product including a computer useable medium having a computer readable program of computer instructions stored thereon that when executed on a computer causes the computer to carry out operations for multi-bitrate content streaming as described herein.

134. Hurst teaches that software, computer program products and computer programs are stored on computer useable mediums, such as compact disks, digital-video disks, magnetic tapes, etc., and are executed by various types of processors. *See id.* A POSITA would have recognized Hurst’s teachings of software, computer program products and computer programs as widely-known and widely-used types of applications, and that Hurst’s teachings of computer useable mediums as widely-known and widely-used types of non-volatile storage. *See id.*

135. Given these teachings, a POSITA would have understood that the references teach client devices that perform the steps described for the challenged claims using software that runs on processors and is stored on non-volatile storage,

such as disk storage or ROM. It would have been obvious to a POSITA that a client device would have one or more processors as it was widely known in the art that computing devices, including the client devices as taught by Chen and Lindahl, generally had at least one processor. The most common computer architecture, going back decades, has included a processor running software, with the software being stored on non-volatile storage. A POSITA would have found it obvious to store software on non-volatile storage so that the device software is retained even when power is removed. A POSITA would have recognized disk storage and ROM as widely-known, and widely-used types of non-volatile storage. Additionally, a POSITA would have understood and found it obvious for a client device implementing Chen, Lindahl, and Hurst's teachings to do so with a client device, having one or more processors, using software applications written to non-volatile storage, to perform the steps described below with respect to the challenged claims.

[1c] obtaining a top level index file identifying a plurality of alternative streams of protected video,
--

136. The combination teaches or suggests obtaining a top level index file (e.g., a Media Presentation Description “MPD” file) identifying a plurality of alternative streams (e.g., representations) of protected (e.g., partially encrypted) video. *See, e.g.*, Ex. 1006, ¶¶[0033, 0063, 0121, 0201, 0216-0217, 0219-0267],

FIG. 5, Ex. 1007, ¶¶[0010, 0033, 0055-0061], FIG. 5A-5C, 6A-6B. Chen teaches a Media Presentation Description (“MPD”) file that identifies a plurality of alternative representations of a video, which are streamed to the client device. *See, e.g.*, Ex. 1006, ¶¶[0219] (“In the methods described herein, a ‘Media Presentation Description’ is provided that comprises information regarding the representations of the media presentation that are available to the client. **Representations may be alternatives in a sense that the client selects one out the different alternatives,** or they may be complementary in a sense that the client selects several of the representations, each possibly also from a set of alternatives, and presents them jointly.”) (emphasis added), [0227-0253] (“Among others, a media presentation description may contain elements and attributes as follows....**Alternative Representation: Describes a list of representations.**”) (emphasis added), [0063] (“HTTP streaming is a specific type of streaming. ... The media, such as video, may be encoded at multiple bitrates to form different versions, or representations. The terms ‘version’ and ‘representation’ are used synonymously in this document.”), FIG. 5:

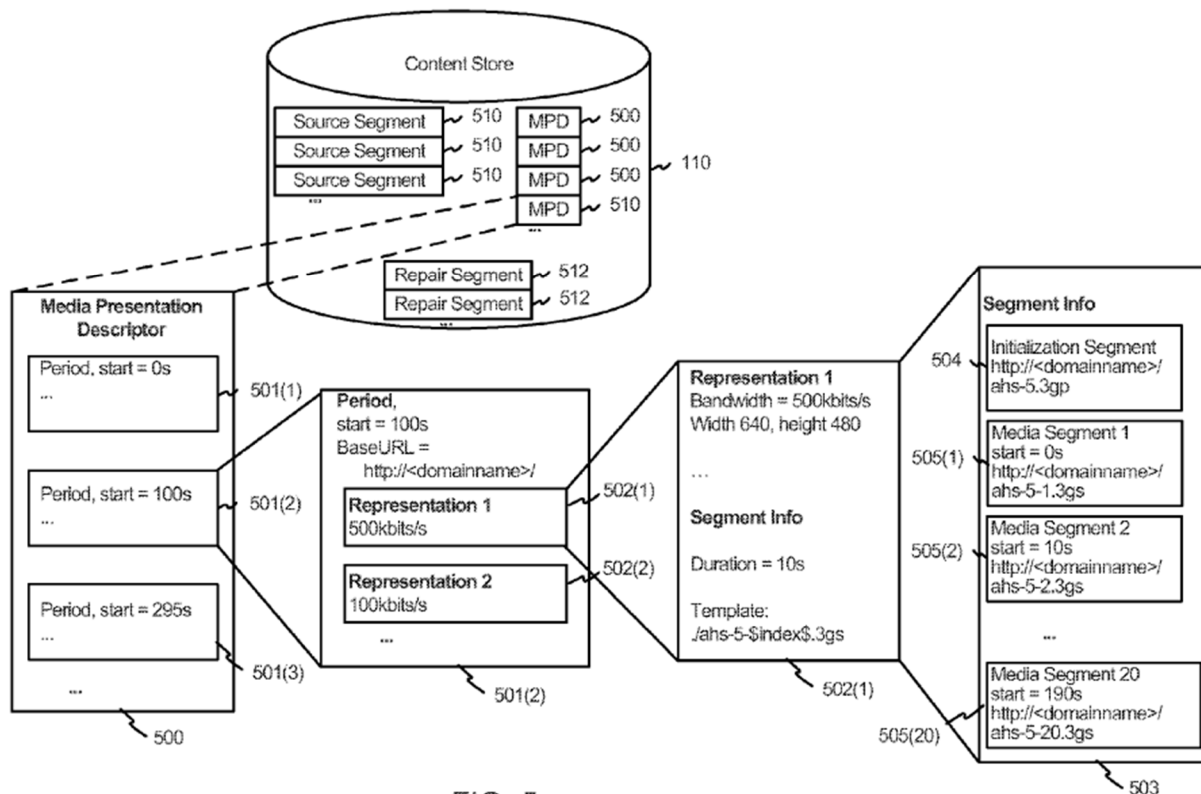


FIG. 5

See also *id.*, ¶[0225] (“Alternative representations may comprise of one or more 3GP files or parts of 3GP files conforming to the 3GPP file format or at least to a well defined set of data structures that can be easily converted from or to a 3GP file.”). A POSITA would have been motivated to utilize Chen’s MPD teachings, which identify alternative representations of a video, because it was the typical and efficient mechanism for adaptive streaming, as admitted by the ’588 patent. See Ex. 1001, 2:11-14.

137. Thus, Chen teaches a top-level index MPD file that identifies a plurality of alternative representations of protected video. A POSITA would have been motivated to apply these teachings because accessing alternative

representations using HTTP was one of the most widely-known and widely-used video streaming architectures by 2011. *See* Ex. 1006, ¶[0226]; *supra* Sections III, IV. For example, Chen explains that representations “typically” are alternatives. *See* Ex. 1006, ¶[0223]. And Chen explains the use of alternative representations in its embodiments. *See, e.g.,* Ex. 1006, ¶¶[0121] (“[T]he client may use the Media Presentation Description simply to select the appropriate alternative version from amongst multiple available versions.”), [0226]. A POSITA would have found additional motivation to use alternative representations, each one separately encoded at a different bitrate, as taught by Chen, because its widespread use as the typical implementation for adaptive streaming meant that a substantial amount of infrastructure and support facilitated its use for videos streaming. For example, Chen provides details for implementing the XML `AlternativeRepresentationElement` in an MPD file. *See* Ex. 1006, ¶¶[0246-0253] (“`AlternativeRepresentation Element: ... RepresentationID: A unique ID for this specific Alternative Representation within the media presentation. ...`”). These sorts of implementation details were known in the art because alternative representations were commonly used. Separately encoding a video stream at different bitrates was the simplest and most common technique for adaptive streaming in the years before 2011, and thus an ecosystem of hardware, software, and standardization existed to facilitate its use.

138. Chen teaches the alternative representations are encoded at different bitrates. *See, e.g.*, Ex. 1006, ¶[0063] (“The media, such as video, may be encoded at multiple bitrates to form different versions, or representations.”). Similarly, the ’588 patent states that “[i]n many instances, alternative streams encode media content (such as but not limited to video) at different maximum bitrates.” Ex. 1001, 11:5-7. Thus, a POSITA would have understood that Chen teaches obtaining a top level index file identifying a plurality of alternative streams of protected video.

139. Hurst further renders this limitation obvious by teaching a plurality of alternative streams of protected video, including alternative representations of a content file encoded at varying bitrates and quality. *See* Ex. 1008, 6:32-58 (“...FIG. 2 b is a schematic block diagram illustrating one embodiment of **a plurality of streams 202 having varying degrees of quality and bandwidth**. In one embodiment, the plurality of streams 202 comprises a low quality stream 204, a medium quality stream 206, and a high quality stream 208. **Each of the streams 204, 206, 208 contains an encoded representation of the content file 200 encoded and compressed to varying bit rates....**”) (emphasis added), FIG. 2b:

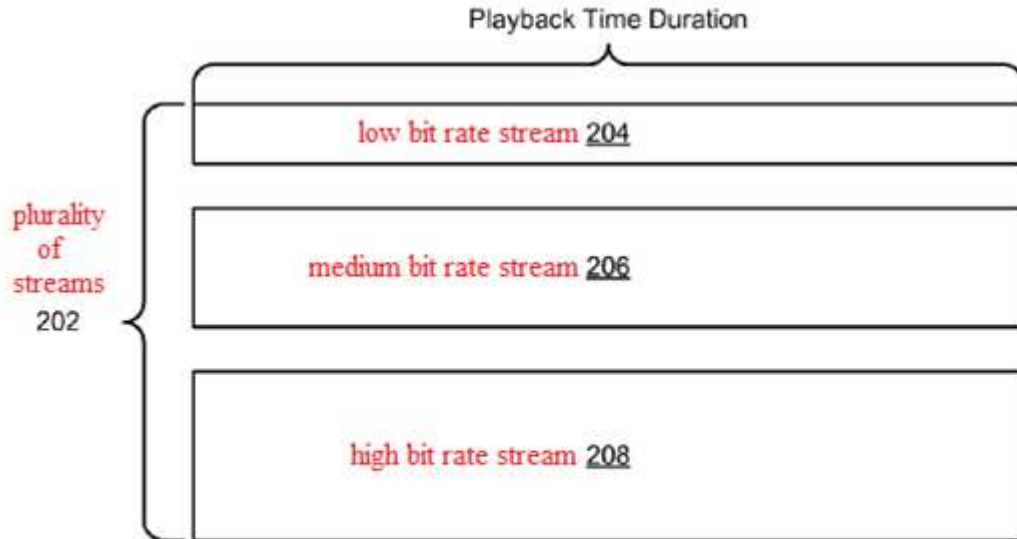


FIG. 2b
(ANNOTATIONS ADDED)

A POSITA would have recognized Hurst’s teachings of alternative streams to be analogous to Chen’s and thus been motivated to apply Hurst’s analogous teachings to reinforce those of Chen.

140. The combination further teaches that each of the alternative video streams is protected, through encryption, as explained below. *See* Section VIII.A.2.[1d-1e]. Thus the combination of Chen, Lindahl and Hurst teach a plurality of alternative streams of protected video.

[1d] wherein each of the alternative streams of protected video includes partially encrypted video frames

141. The combination teaches or suggests that each of the alternative streams (e.g., representations) of protected video includes partially encrypted video frames. *See, e.g.,* Ex. 1006, ¶¶[0070, 0121, 0140, 0219, 0223, 0225, 0246-0251],

Ex. 1007, ¶¶[0010, 0033, 0055-0061], FIG. 5A-5C, 6A-6B. Lindahl teaches that portions of each frame are encrypted, while other portions remain unencrypted, providing an illustration using a representative frame. *See, e.g.*, Ex. 1007, ¶[0055] (emphasis added):

In FIG. 5B, **a representative one of the frames from the media file 500 is depicted as a frame 520.** The frame 520 is divided into blocks B1, B2 and B3. Each of the blocks B1, B2 and B3 are of the same size. FIG. 5C illustrates a representative block 540 according to one example. The representative block 540 can represent any of the blocks B1, B2 or B3 of the frame 520 illustrated in FIG. 5B. As shown in FIG. 5C, **only a portion of the block 540 is encrypted. More particularly, the block 540 includes an encrypted portion 542. The encrypted portion 542 is substantially less than the entire block 540. Hence, those portions of the block 540 other than the encrypted portion 542 remain unencrypted.**

See id., FIG. 5A-5C:

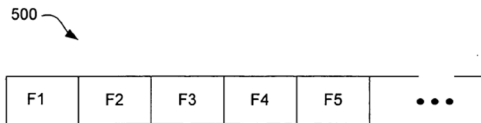


FIG. 5A

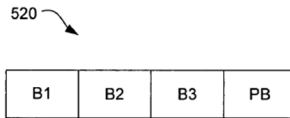


FIG. 5B

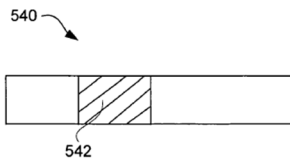


FIG. 5C

See also *id.* ¶[0056] (“Partial encryption and the use of encryption parameters is discussed in additional detail below with reference to FIGS. 6A and 6B. The encryption parameters control **the portion of a block that is to be encrypted. The remaining portions of a block that are not encrypted remain unencrypted** (i.e., ‘in the clear’).”) (emphasis added).

142. A POSITA would have found it obvious that Lindahl’s teaching of a partially-encrypted frame applies to all frames in the video because the partially-encrypted frame is described as “a representative one of the frames from the media file,” meaning it is representative of how all the frames in the video are encrypted. See *id.*, ¶¶[0055], [0060]. Furthermore, a POSITA would have found it obvious and been motivated to partially encrypt all frames, as taught by Lindahl, because

doing so would have protected video data in each frame, rendering each frame and the video as a whole effectively unwatchable for unauthorized users. Here, Lindahl is teaching that each frame is partially encrypted because each frame is divided into at least 3 blocks, each of which contains encrypted and unencrypted portions. *See id.* A POSITA would have understood Lindahl to be teaching the well-known and efficient technique of partial encryption, which was widely known and used in the art of video DRM by 2011 for encrypting portions of each frame of a video stream. *See supra* Section IV.B.

[1e] that are encrypted using a set of common keys comprising at least one key,
--

143. The combination teaches or suggests that each of the alternative streams (e.g., representations) of protected video includes partially encrypted video frames that are encrypted using a set of common keys comprising at least one key. *See, e.g.,* Ex. 1006, ¶¶[0070, 0121, 0140, 0219, 0223, 0225, 0246-0251], Ex. 1007, ¶¶[0017, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084]. Lindahl and Hurst both teach this limitation, and either teaching is sufficient by itself to render the limitation obvious.

144. **First**, Lindahl teaches that the client may receive a global key from the server, and media files may be encrypted (and therefore decrypted²) with the global key. See Ex. 1007, ¶[0066] (“Once the media file is downloaded 708, the media file can then be decrypted 710 at the user's local machine using the cryptographic key. **The global key or other cryptographic key may also be used.** If the metadata of the media file itself is encrypted, the metadata can also be decrypted using the global key which has previously been provided to the user who has purchased the access rights to the media file.”) (emphasis added). Here, Lindahl teaches one embodiment where a common “global” key is used to encrypt and decrypt all of the media files on the server. A POSITA would have been motivated to apply this teaching to encrypt all media files, including all alternative representations, using a global key because it would simplify key management particularly for content that is not particularly sensitive. Lindahl teaches that when data “is not that sensitive ... the use of a common global key to encrypt such is adequate.” See, e.g., Ex. 1007, ¶[0048] (emphasis added):

The metadata of the media file is optionally encrypted, such as with a global key. In one embodiment, the global key is used to encrypt the metadata for all the media files at the media server. In another embodiment, if desired, different global keys can be used with different media servers. Typically, the metadata is not that sensitive so

² A POSITA would have understood Lindahl’s teachings of using symmetric encryption algorithms such as Blowfish, AES and DES meant the same key was used for encryption and decryption. See e.g., Ex. 1007, ¶[0008].

the use of a ***common global key*** to encrypt such is adequate. However, if stronger encryption is desired, then more specialized keys can be used to increase the level of security. In another embodiment, the metadata is not encrypted.

145. While this particular passage uses metadata as one example, a POSITA would have understood this teaching to refer to a well-known security principle: reusing a global (e.g., single) key for many videos can provide sufficient protection for content that is less sensitive, including for example (a) marketing or promotional videos, such as advertisements, commercials, etc., where a primary goal is to spread viewership of the commercials, and piracy is less of a concern, (b) non-profit materials, such as educational or public awareness videos, (c) academic materials that are provided for free and (d) public access videos. These are examples where content was provided at low cost or free, but where DRM was often still used, following similar transaction models where the user purchases content and receives a license with encryption keys, but where the purchase price was small or zero. DRM was used to provide a basic level of protection to prevent crimes of convenience. But since would-be pirates do not have much incentive to break the encryption, despite its relative simplicity, these usage models require DRM but do not require as rigorous security as other applications. Given Lindahl's teachings, a POSITA would have found it obvious and been motivated to use a common global key to encrypt/decrypt media files,

including each of the alternative streams of protected video, and thus each of the alternative streams of protected video would include partially encrypted video frames that are encrypted using a set of common keys comprising at least one global key.

146. **Second**, a POSITA would have found it obvious to use a common set of keys to encrypt the alternative streams because the application of Lindahl to Chen leaves only a finite number—two possibilities—of identified, predictable solutions for encrypting alternative streams for adaptive streaming: either (1) use the same set of keys for all representations of a video, or (2) use a different set of keys for each representation. A POSITA would have found it obvious to use the same set of keys as one of those two possibilities; recognized the design need to apply encryption to adaptive streaming because it was widely known and taught in the art that adaptive video streaming would require DRM to secure copyrighted content over the Internet; and recognized the market pressure to do so given the growth of popular commercial streaming services such as Netflix and Hulu. *See supra* Section IV. A POSITA would have found it obvious to do so—or at least obvious to try given the finite number of possibilities.

147. Hurst reinforces the obviousness of this teaching by expressly stating that the alternative “streamlets 304 of different bitrates may be encrypted with the same key.” *See* Ex. 1008, 18:64-67. Hurst teaches that these streamlets of

different bitrates come from alternative streams. Hurst teaches encoding a content file, such as a video file, into a plurality of streams at different bitrates, such as low, medium and high bitrate streams. See Ex. 1008, FIG. 2b:

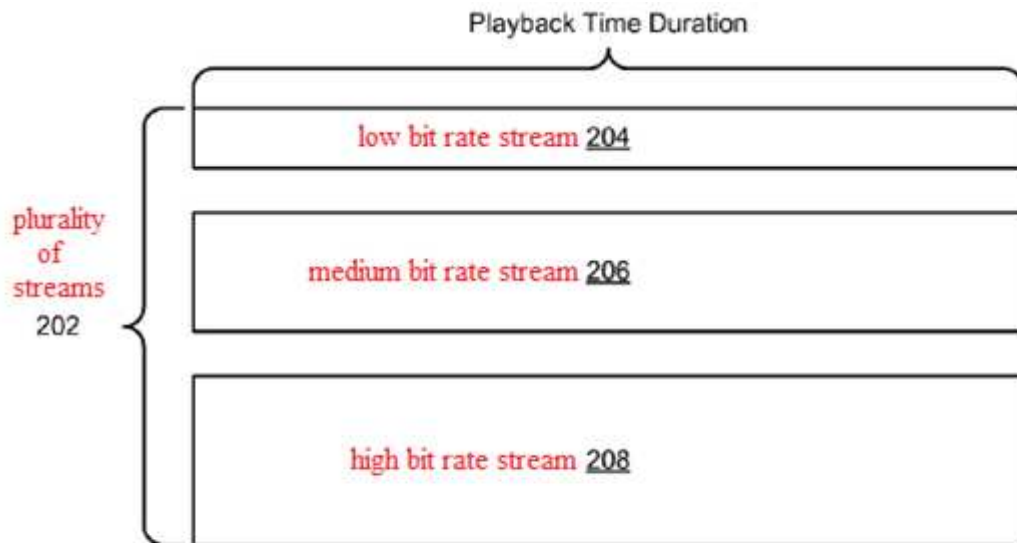


FIG. 2b
(ANNOTATIONS ADDED)

See also, *id.* 6:32-58 (emphasis added):

FIG. 2b is a schematic block diagram illustrating one embodiment of a **plurality of streams 202 having varying degrees of quality and bandwidth**. In one embodiment, the plurality of streams 202 comprises a low quality stream 204, a medium quality stream 206, and a high quality stream 208. **Each of the streams 204, 206, 208 contains an encoded representation of the content file 200 encoded and compressed to varying bit rates**. For example, the low quality stream 204 may be encoded and compressed to a bit rate of 100 kilobits

per second (kbps), the medium quality stream 206 may be encoded and compressed to a bit rate of 200 kbps, and the high quality stream 208 may be encoded and compressed to 600 kbps.

A POSITA would have understood Hurst's streams to be alternative streams because they represent the same content (e.g., the content file 200) and are encoded at varying degrees of quality and bandwidth so that they may be selected, in the alternative, based on channel bandwidth. *See, e.g.*, Ex. 1008, 8:16-18, Abstract, 1:6-10, 16:61-17:58, Figure 10. Each "stream 302 divided into a plurality of source streamlets 303." *Id.*, 6:59-62, FIG. 3a:

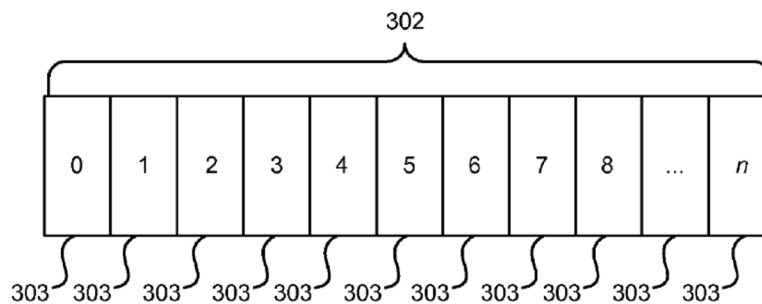


FIG. 3a

148. Putting this together (applying Fig. 3a to 2b), Hurst illustrates the alternative low, medium and high bitrate streams divided into streamlets. *Id.*, 7:9-20 ("The Set 306a includes encoded Streamlets 304 having low 204, medium 206, and high 208 bitrates. ... any number of streams having different bitrates may be generated from the original content 200."), FIG. 3b:

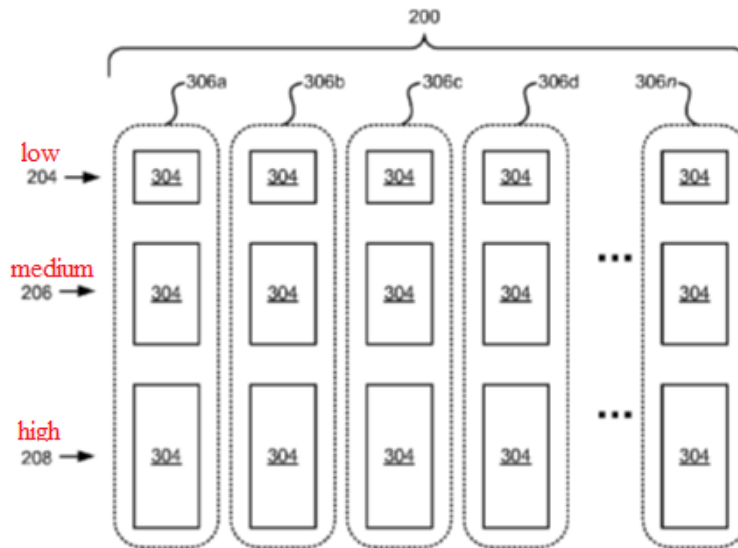


FIG. 3b
(ANNOTATIONS ADDED)

Figure 3b depicts sets of streamlets, where “the term ‘set’ refers to a group of streamlets 304 having identical time indices and durations but varying bitrates.”

Id., 7:9-20. Hurst teaches “[t]he streamlets 304 of different bitrates may be encrypted with the same key[,]” which in reference to Figure 6b teaches that each of the streamlets from the low 204, medium 206, and high 208 bitrates are encrypted using the same key. Ex. 1008, 18:64-67.

149. For demonstrative purposes, I have annotated Figures 2b and 3b below to reflect Hurst’s teachings regarding the relationship between Hurst’s low, medium, and high bitrate streams, shown in green, yellow, and red respectively, and Hurst’s streamlets, shown in Figure 3b. See Ex. 1008, FIGS. 2b, 3b:

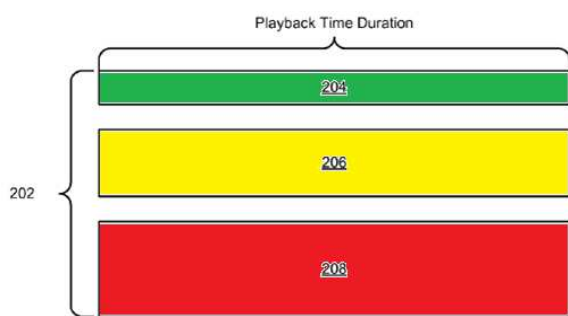


FIG. 2b
(ANNOTATIONS ADDED)

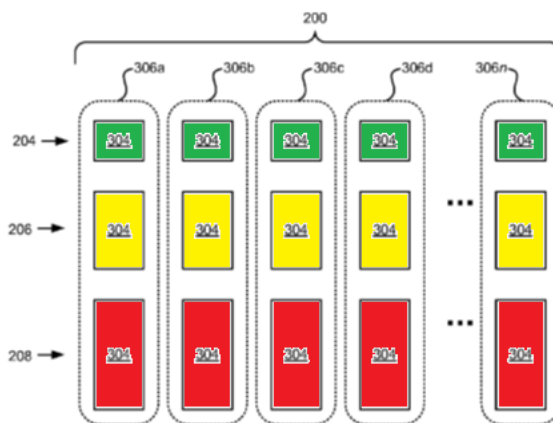


FIG. 3b
(ANNOTATIONS ADDED)

150. A POSITA would have understood that Hurst’s streamlets are analogous to Chen’s segments. Both teach content, such as a video, encoded at different bitrates, where the alternative streams are then divided based on time. Chen refers to these subdivisions as “segments,” while Hurst refers to them as “streamlets,” but both serve the same purpose within the context of adaptive streaming by allowing a playback device to switch between the alternative representations, accessing different portions at different times, depending on the currently available bandwidth. *See* Ex. 1006, ¶¶[0022-0025, 0027, 0063-0065, 0080-0082, 0121, 0148, 0408-0418]; Ex. 1008, 7:9-20. In this context, applying Lindahl’s teachings, it would have been obvious and straightforward to apply Hurst’s key management teachings, with respect to the alternative streams, to Lindahl’s teachings regarding cryptographic keys. *See* Ex. 1007, ¶¶[0064]-[0065]; Ex. 1008, 18:40-19:14.

151. Thus it would have been obvious based on the combined teachings of the references to use a common set of keys to encrypt the alternative representations of a video, and thus each of the alternative streams of protected video would include partially encrypted video frames that are encrypted using a set of common keys comprising at least one key.

[1f] and wherein the partially encrypted video frames contain encrypted portions and unencrypted portions of data;

152. The combination teaches or suggests that the partially encrypted video frames contain encrypted portions and unencrypted portions of data. *See, e.g.*, Ex. 1007, ¶¶[0010, 0033, 0055-0061], FIG. 5A-5C, 6A-6B. Lindahl teaches that each partially encrypted video frame contains partially encrypted blocks, which have one or more encrypted portions and one or more unencrypted portions. *See, e.g.*, Ex. 1007, ¶[0055] (“In FIG. 5B, a representative one of the frames from the media file 500 is depicted as a frame 520. The frame 520 is divided into blocks B1, B2 and B3. ... The representative block 540 can represent any of the blocks B1, B2 or B3 of the frame 520 illustrated in FIG. 5B. As shown in FIG. 5C, ***only a portion of the block 540 is encrypted. More particularly, the block 540 includes an encrypted portion 542. The encrypted portion 542 is substantially less than the entire block 540. Hence, those portions of the block 540 other than the encrypted portion 542 remain unencrypted.***”) (emphasis added), Fig. 5C:

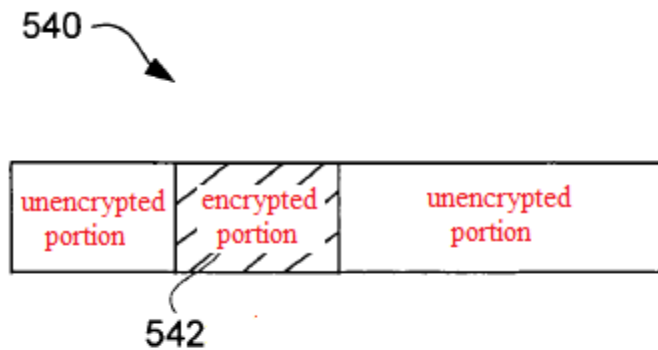


FIG. 5C
(ANNOTATIONS ADDED)

See also, id., ¶[0061] (“In general, each partially encrypted block can have one or more portions encrypted and one or more portions unencrypted. Preferably, the one or more portions being encrypted are not at the start of the block. The portion to encrypt should be important and difficult to guess so that a decoder is disrupted and not able to easily recover from the errors produced by the partial encryption. The percentage or amount of partial encryption to be performed with respect to a block can be set by the encryption parameters.”). Thus Lindahl teaches a video frame with encrypted portions and unencrypted portions of data. A POSITA would have been motivated to apply Lindahl’s partial encryption teachings for several reasons, including to improve computational efficiency, as explained previously. *See supra* Section VIII.A.1.

153. Additionally, Lindahl teaches that encryption parameters control what portions are encrypted, to create partially encrypted video frames containing encrypted portions and unencrypted portions of data. *See* Ex. 1007, ¶[0056]

(“Partial encryption and the use of encryption parameters is discussed in additional detail below ... The encryption parameters control the portion of a block that is to be encrypted. The remaining portions of a block that are not encrypted remain unencrypted (i.e., ‘in the clear’).”), ¶[0060] (“In one embodiment, the partially encrypted blocks 600 and 620 can represent all blocks of a given frame that are partially encrypted.”), FIG. 6A-6B:

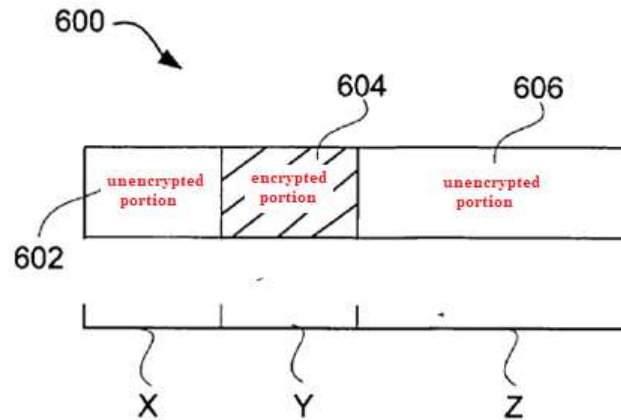


FIG. 6A

(ANNOTATIONS ADDED)

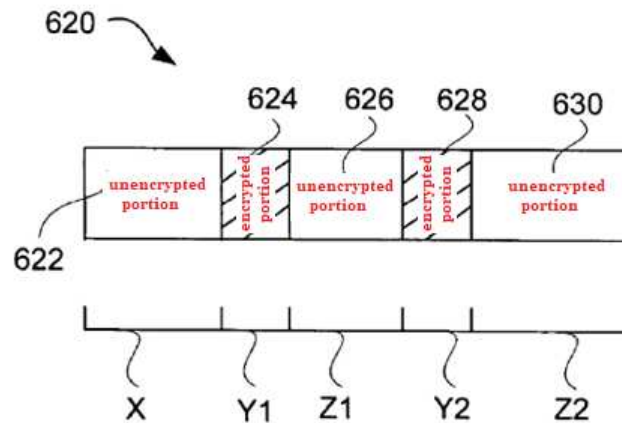


FIG. 6B

(ANNOTATIONS ADDED)

[1g] obtaining a copy of the set of common keys;

154. The combination teaches or suggests obtaining a copy of the set of common keys. *See, e.g.*, Ex. 1007, ¶¶[0008, 0017, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084]. For example, the references teach

obtaining the set of common keys to decrypt the content. Lindahl teaches that as part of playback processing, “the associated encryption key is obtained 706 from the selected media file” so that “the encrypted media file can be decrypted 808 using the associated encryption key.” *See* Ex. 1007, ¶¶[0069-0070]. Similarly, Hurst teaches “[t]he DRM server 1204 is further configured to supply encryption keys to the end user upon authenticating the end user.” Ex. 1008, 18:53-55. Thus Lindahl and Hurst both teach the client device obtaining the requisite keys to decrypt the selected content.

155. In the context of this combination, the keys associated with a particular piece of content—including its alternative streams—would have been a set of common keys comprising at least one key. *See supra* Section VIII.A.2.[1e]. Given Lindahl and Hurst’s teachings, a POSITA would have found it obvious to obtain the same set of common keys used to encrypt the alternative streams, in order to decrypt those streams.

156. A POSITA would have found it obvious to decrypt the streams with the same keys used to encrypt them because the combination teaches symmetric encryption. Lindahl teaches using symmetric encryption algorithms to encrypt media files, where the same “key both encrypts and decrypts the data.” *See* Ex. 1007, ¶[0008] (“Symmetric key algorithms, such as Blowfish, AES and DES, work with a single, prearranged key that is shared between sender and receiver. This key

both encrypts and decrypts the data.”). Lindahl teaches the use of symmetric keys for several embodiments. *See, e.g.*, Ex. 1007, ¶¶[0049] (“Here, the media file is encrypted 306 with the cryptographic key corresponding to that particular media file. At this point, the media file (encrypted media file) is secured through encryption and ready for download to potential users.”), [0066] (“Once the media file is downloaded 708, the media file can then be decrypted 710 at the user's local machine using the cryptographic key. The global key or other cryptographic key may also be used.”). A POSITA would have found it obvious and been motivated to use symmetric keys to encrypt video data, as Lindahl teaches, because that had been the general practice for video encryption since at least the 1990s. Since video often includes a large amount of data, it was widely known, as a best practice, to encrypt video data using symmetric keys, which was (and is) significantly more computationally efficient, while saving asymmetric encryption for key distribution. Given Lindahl’s teaching that the playback device decrypts media, it would have been obvious for the playback device to obtain the symmetric keys that were used to encrypt the media—i.e., the associated encryption keys—because they are required for decrypting the media. *See id.*

157. Various mechanisms were known in the art for obtaining and transferring keys, including for example encrypting and then sending the keys, which had been used in the art by 2011. *See supra* Section IV.B. Lindahl teaches

several such mechanisms for the playback device to obtain associated keys, including by downloading the keys along with the media, or by downloading them separately. *See* Ex. 1007, ¶¶[0060, 0079]. Given Lindahl and Hurst’s teachings of obtaining an encryption key to play selected media and transferring a cryptographic key to a media player, and given a POSITA’s knowledge of common cryptographic techniques such as transferring keys, it would have found it obvious for the playback device to obtain a copy of the set of common keys to decrypt the alternative streams of protected video.

[1h] detecting streaming conditions for the playback device;

158. The combination teaches or suggests detecting streaming conditions for the playback device (e.g., the client device). *See, e.g.,* Ex. 1006, ¶¶[0022-0025, 0027, 0063-0065, 0080-0082, 0121, 0148, 0408-0418]. For example, Chen teaches the client device³ detecting the available bandwidth so that a stream with a matching data rate can be selected. *See, e.g.,* Ex. 1006, ¶¶[0081] (“This selection may be based on any number of criteria, including **matching of the media data rate to the available bandwidth, even when the available bandwidth is changing over time ...**”) (emphasis added), [0121] (“the client may use the Media Presentation Description simply to select the appropriate alternative version from

³ As a POSITA would have understood, Chen refers to the client device as a receiver. *See, e.g.,* Ex. 1006,[0201] (“the client or receiver”).

amongst multiple available versions. Clients may also use HTTP partial get requests with files compliant to the Progressive Download profile to request subsets of each alternative version”), [0063-0065] (emphasis added):

HTTP streaming is a specific type of streaming. ... The media, such as video, may be encoded at multiple bitrates to form different versions, or representations. The terms “version” and “representation” are used synonymously in this document. Each version or representation may be broken into smaller pieces, perhaps on the order of a few seconds each, to form segments. ... On the client side, requests may then be made, using HTTP for individual segments that are seamlessly spliced together by the client. **The client may switch to different data rates based on available bandwidth. ... Triggers for switching may include buffer occupancy and network measurements**, for example. ... These advantages come from controlling the delivery to be only a short time ahead of the playout, **making maximum use of available bandwidth (through variable bit rate media)**, and optimizing stream segmentation and intelligent client procedures.

159. A POSITA would have been motivated to apply these teachings from Chen because detecting streaming conditions, including through buffer occupancy and network measurements, was a widely-known and practiced, basic function in adaptive streaming—as admitted by the ’588 patent (*see* Ex. 1001, 1:59-67)—as it

provides the receiver with information regarding the available channel bandwidth to facilitate selecting an appropriate representation so as to avoid errors or interruptions in playback. *See supra* Sections IV.A; VIII.A.1; *infra* Section VIII.A.2.[1i]. Chen’s teaching of a playback device detecting situations wherein available bandwidth falls low is an illustration of this widely-known and widely-used aspect of adaptive streaming.

160. Given Chen’s teaching that the client “switch[es] to different data rates based on available bandwidth” and “[t]riggers for switching may include buffer occupancy and network measurements,” it would have been obvious that Chen teaches buffer occupancy and network measurements for detecting the available bandwidth for streaming—thereby detecting streaming conditions for the playback device. *See* Ex. 1006, ¶¶[0063-0065]. Changes in the detected bandwidth trigger the client device switching between the alternative streams, as explained below. *See id.*; *infra* Section VIII.A.2[1i], VIII.A.8[7b]. Thus Chen teaches detecting streaming conditions for the playback device and renders this limitation obvious.

[1i] selecting a stream from the plurality of alternative streams of protected video based on the detected streaming conditions;

161. The combination teaches or suggests selecting a stream from the plurality of alternative streams (e.g., representations) of protected (e.g., partially

encrypted) video based on the detected streaming conditions (e.g., network measurements/available bandwidth). *See, e.g.*, Ex. 1006, ¶¶[0022-0025, 0027, 0063-0065, 0070, 0080-0082, 0121, 0140, 0148, 0219, 0223, 0225, 0246-0251, 0408-0418], FIG. 5, Ex. 1007, ¶¶[0010, 0033, 0055-0061], FIG. 5A-5C, 6A-6B. Chen teaches a client device selecting blocks from alternative representations, encoded at different bitrates, based on available bandwidth, which was detected in the previous step. *See, e.g.*, Ex. 1006, ¶¶[0080-0081] (“[T]he division of the media presentation into blocks allows each block of encoded media data to be **selected from a set of multiple available encodings. This selection may be based on any number of criteria, including matching of the media data rate to the available bandwidth**, even when the available bandwidth is changing over time ...”) (emphasis added), [0063-0065] (emphasis added):

HTTP streaming is a specific type of streaming. ... **The media, such as video, may be encoded at multiple bitrates to form different versions, or representations.**

The terms “version” and “representation” are used synonymously in this document. Each version or representation may be broken into smaller pieces, perhaps on the order of a few seconds each, to form segments. ... On the client side, requests may then be made, using HTTP for individual segments that are seamlessly spliced together by the client. **The client may switch to different data rates based on available**

bandwidth. ... Triggers for switching may include buffer occupancy and network measurements, for example. ... These advantages come from controlling the delivery to be only a short time ahead of the playout, **making maximum use of available bandwidth (through variable bit rate media),** and optimizing stream segmentation and intelligent client procedures.

162. Thus, Chen teaches that video is encoded at multiple bitrates to form different representations, and a receiver selects a representation, based on its bitrate, to match the bitrate with the available bandwidth. For example, when a receiver detects insufficient bandwidth, it selects an alternative representation with a lower bitrate that uses less bandwidth. *See id.* These are general descriptions of common adaptive bitrate streaming implementations, and Chen explains its teachings in the context of HTTP streaming. *See id.* ¶¶[0063-0065]. A POSITA would have understood such teachings to be widely known in the art of adaptive bitrate streaming.

163. Chen illustrates the different bitrates for alternative representations in Figure 5. *See, e.g.,* Ex. 1006, FIG. 5:

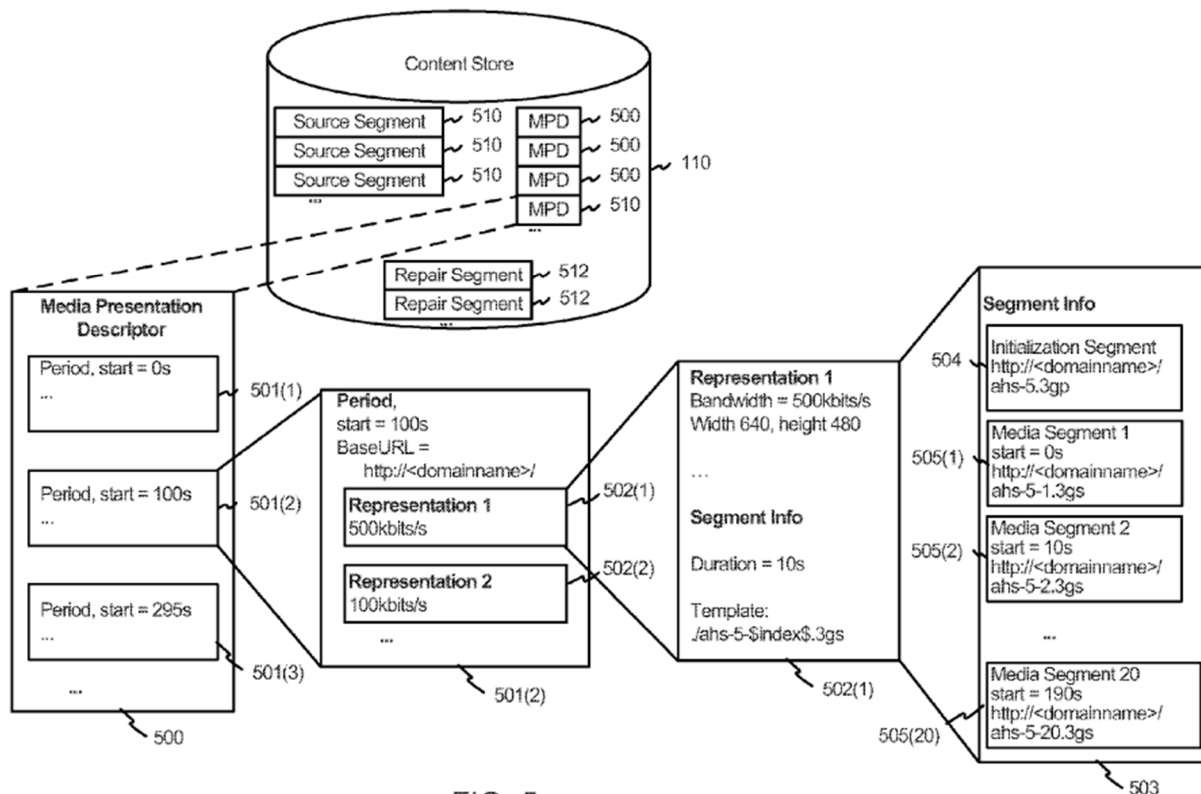


FIG. 5

In Figure 5, Chen illustrates information used to select an appropriate representation based on streaming conditions, including an MPD that provides information for multiple representations and periods. Here, a closeup view of the Period starting at 100s shows Representation 1, which is associated with a 500kbts/s bitrate, and Representation 2, which is associated with a 100kbts/s bitrate. Chen teaches selecting a stream based on the detected available bandwidth. See Ex. 1006, ¶¶[0063-0065], [0080-0081]. For example, if Representation 1 is being streamed, and the available bandwidth falls below 500 kb/s to 400 kb/s, Chen teaches selecting a lower bitrate representation, which in this case would have been Representation 2, whose bitrate of 100 kb/s can be

streamed within the currently available bandwidth. Thus Chen teaches this limitation.

164. A POSITA would also have found it obvious that each representation of video would be protected through encryption, including by partially encrypting its video frames. *See supra* Section VIII.A.2[1d].

[1j] receiving a container index that provides byte ranges for portions of the selected stream of protected video within an associated container file;
--

165. The combination teaches or suggests receiving a container index (e.g., segment index/map) that provides byte ranges for portions (e.g., fragments/blocks) of the selected stream (e.g., representation) of protected (e.g., partially encrypted) video within an associated container file (e.g., segment). *See, e.g.*, Ex. 1006, ¶¶[0033, 0063-0064, 0072, 0111, 0119-0123, 0129, 0147-0150, 0157-0158, 0175-0187, 0197, 0201, 0216-0217, 0219-0267], FIG. 5, 6, 9(a), Ex. 1007, ¶¶[0010, 0033, 0055-0061], FIG. 5A-5C, 6A-6B. For example, Chen teaches that after a representation is selected (*see supra* Section VIII.A.2.[1i]), the client accesses associated segments for that representation, which are stored as files on the server. *See* Ex. 1006, ¶¶[0063-0064] (“Each version or representation may be broken into smaller pieces, perhaps on the order of a few seconds each, to form segments. Each segment may then be stored on a web server or CDN as a separate file. **On the**

client side, requests may then be made, using HTTP, for individual segments that are seamlessly spliced together by the client..”) (emphasis added), Fig. 5:

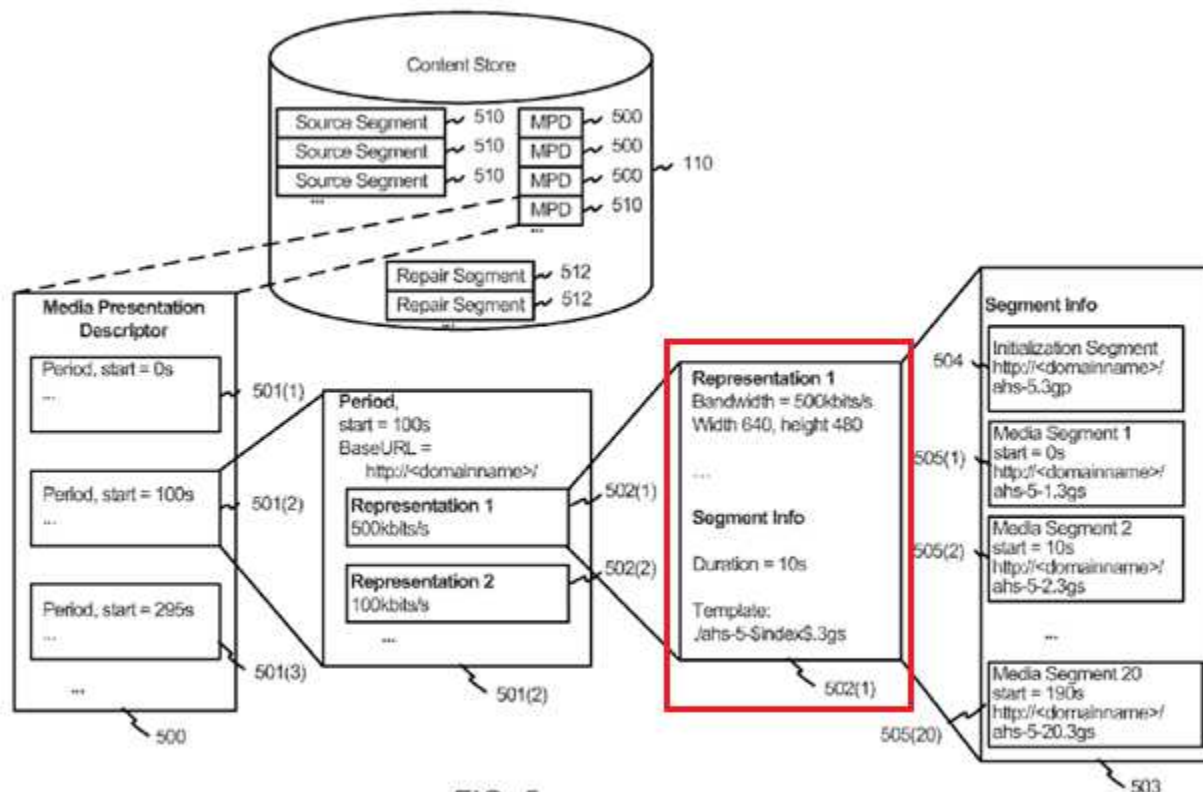


FIG. 5
(ANNOTATIONS ADDED)

Chen teaches that the segments are files. See Ex. 1006, ¶¶[0072] (“It should be noted that throughout this description references to a ‘segment’ should be understood to include any data object which is fully or partially constructed or read from a storage medium or otherwise obtained as a result of a file download protocol request...**The terms ‘file’ and ‘segment’ are used synonymously in this document unless otherwise specified.**”) (emphasis added). Chen teaches that each container file (segment) contains an index. See Ex. ¶[0119] (“A segment

index may be included in each segment to signal time offsets and byte ranges...”), [0120] (“ The “moov” box may be placed **at the start of the file**, followed by fragment offset data, also referred to as **a segment index**, containing offset information in time and byte ranges.”) (emphasis added), [0123] (“This **segment indexing may be included within the segment**...Especially in cases where **the segment index is contained in the segment**, the receiver may download some or all of this segment map quickly...” (emphasis added), ¶[0222] (“As described in this reference, **information may be provided within the file** permitting fast and efficient mapping between time segments of the media presentation and **byte ranges within the file.**”) (emphasis added), Fig. 27:

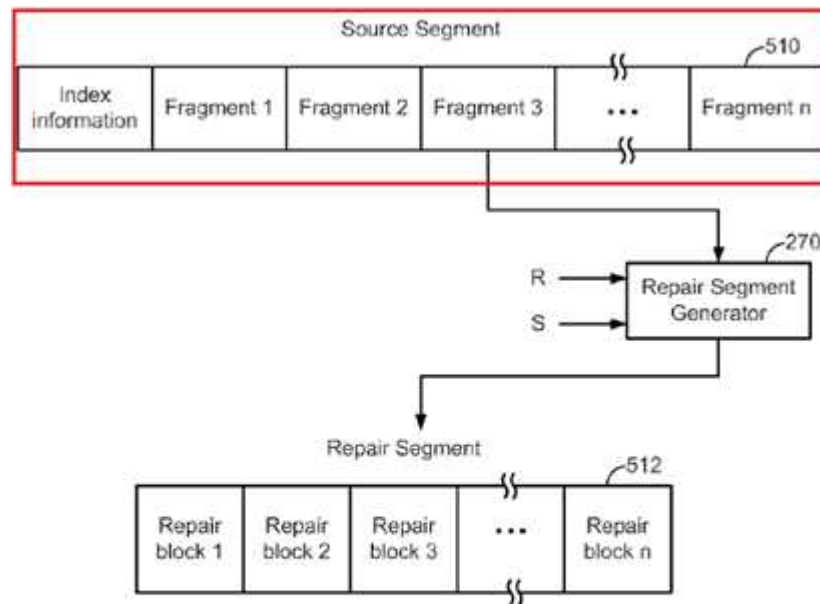


FIG. 27
(ANNOTATIONS ADDED)

Chen further teaches the files/segments are container files that, for example, follow 3GPP or ISO file formats for containers. Ex. 1006, ¶¶[0119] (“The 3GPP File Format or any other file based on the ISO base media file format, such as the MP4 file format or the 3GPP2 file format, **may be used as the container format for HTTP streaming.**”) (emphasis added), [0175] (“The container for ‘sidx’ box is the file or segment directly”). A POSITA would have found it obvious and been motivated to apply these teachings because storing video in container files was a common practice for video streaming many years before 2011, particularly the ISO/MP4 base media file format and 3GPP extensions to that format, which were widely used standards in the art of streaming video, as admitted by the ’588 patent. See Ex. 1001, 2:12-34. Thus Chen teaches a playback device receiving a container index associated with a container file in the form of a segment index associated with a file that is a container.

166. The segment index provides byte ranges for portions of the selected stream of protected video. See, e.g., Ex. 1006, ¶¶[0119] (“**A segment index may be included in each segment to signal time offsets and byte ranges,** such that the client can download the appropriate pieces of files or media segments as required.”) (emphasis added), [0120] (“The “moov” box may be placed at the start of the file, followed by fragment offset data, also referred to as a segment index, **containing offset information in time and byte ranges** for each fragment or at

least a subset of fragments in the containing segment.”) (emphasis added), [0129] (“Segment metadata may include **byte range** versus time information for requests within a segment and identification of Random Access Points (RAPS) or other seek points, where some or all of this information may be part of a segment indexing or segment map.”) (emphasis added), [0148] (“To improve bandwidth usage, segments can include an index file as metadata, where **the index file maps the byte ranges** of individual blocks with the time ranges that the blocks correspond to, called segment indexing or segment map.”) (emphasis added), [0222] (“As described in this reference, information may be provided within the file permitting fast and **efficient mapping between time segments of the media presentation and byte ranges** within the file.”) (emphasis added), FIG. 9(a):

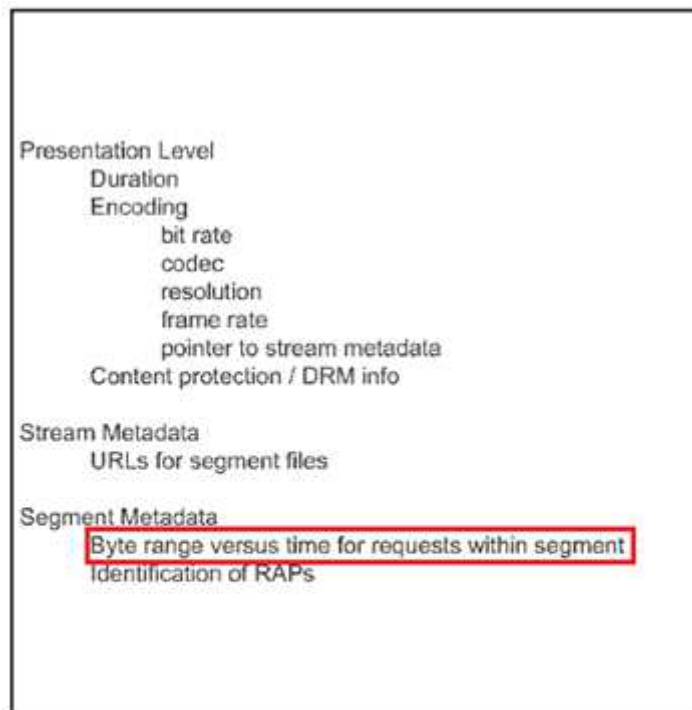


FIG. 9(a)
(ANNOTATIONS ADDED)

Thus, Chen teaches the playback device receiving a container index that provides byte ranges for portions of the selected stream of protected video within an associated container file.

167. Chen's teaching of byte ranges is not by happenstance. Chen provides byte ranges to facilitate adaptive streaming over HTTP—the same reason that indexing byte ranges was well known in adaptive streaming before Chen or the '588 patent. Chen teaches that HTTP requests include a byte range in order to request a portion of a file or segment. See Ex. 1006, ¶[0111]. To facilitate these HTTP range requests, Chen teaches including a segment index in each segment to

indicate byte ranges for a client to download appropriate portions of the segment. *See*, Ex. 1006, ¶[0119]. A POSITA would have found it obvious and been motivated, based on Chen’s teachings, to provide byte ranges in a segment index to facilitate HTTP range requests, as Chen teaches.

168. As explained in Chen, its adaptive streaming technique often accesses segments in the middle or towards the end of the file, as opposed to the beginning, because the playback device jumps between streams, based on bandwidth conditions, and the user may skip ahead in the video. *See, e.g.*, Ex. 1006, ¶¶[0064], (“The client may switch to different data rates based on available bandwidth.”), [0086] (“a user may decide to start playing the content not from the beginning, but from some time within the stream. The user indicates to their client device to seek to a time position and the user might expect that the selected time is rendered instantaneously.”). Thus, Chen teaches that HTTP requests include a byte range to request a portion of a segment, rather than the entirety of each segment. *See* Ex. 1006, ¶[0111]. To facilitate these HTTP requests, Chen teaches providing “byte ranges” in the segment index, so that specific fragments within a segment file may be directly requested. *See, e.g.*, Ex. 1006, ¶¶[0119-0123, 0129, 0147-0150, 0201], FIG. 6.

169. Chen teaches its segment index to be a “Segment Index Box (‘sidx’)” according to the ISO MP4 base file format. *See* Ex. 1006, ¶¶[0157-0158, 0175-

0187]. The above teachings are sufficient to render claim limitation [1j] obvious, though I explain additional details regarding the structure of Chen’s index embodiments below, including the sidx embodiment.

170. I note that the patent owner admits this limitation is met. In its Amended Complaint, PO alleged this limitation [1j] is met “because, for example, MP4 container files ... include ***container indexes in the form of an sidx box, which provides byte ranges*** for portions of a stream of protected video within the container file. For example, the MPEG-DASH Standard requires including an sidx box within the MP4 container file...” See Ex. 1015 ¶429 (emphasis added). As explained above, Chen teaches segment files including a Segment Index Box (“sidx”) according to the ISO base file format, which the MP4 file format is based on. See, e.g., Ex.1006, ¶¶[0158-0189] (teaching sidx using the ISO base media file format), [0119] (“The 3GPP File Format or any other file based on the ISO base media file format, such as the MP4 file format ... may be used as the container format for HTTP streaming ...”). Thus PO admits that limitation [1j] taught by the combined teachings of Chen and Lindahl.

171. **Segment Index Embodiments.** Chen’s teachings, explained above, render this claim limitation obvious. Nonetheless, I provide additional explanation of some of Chen’s embodiments. Chen provides an example of an index in a source segment file. See Ex. 1006, Fig. 6:

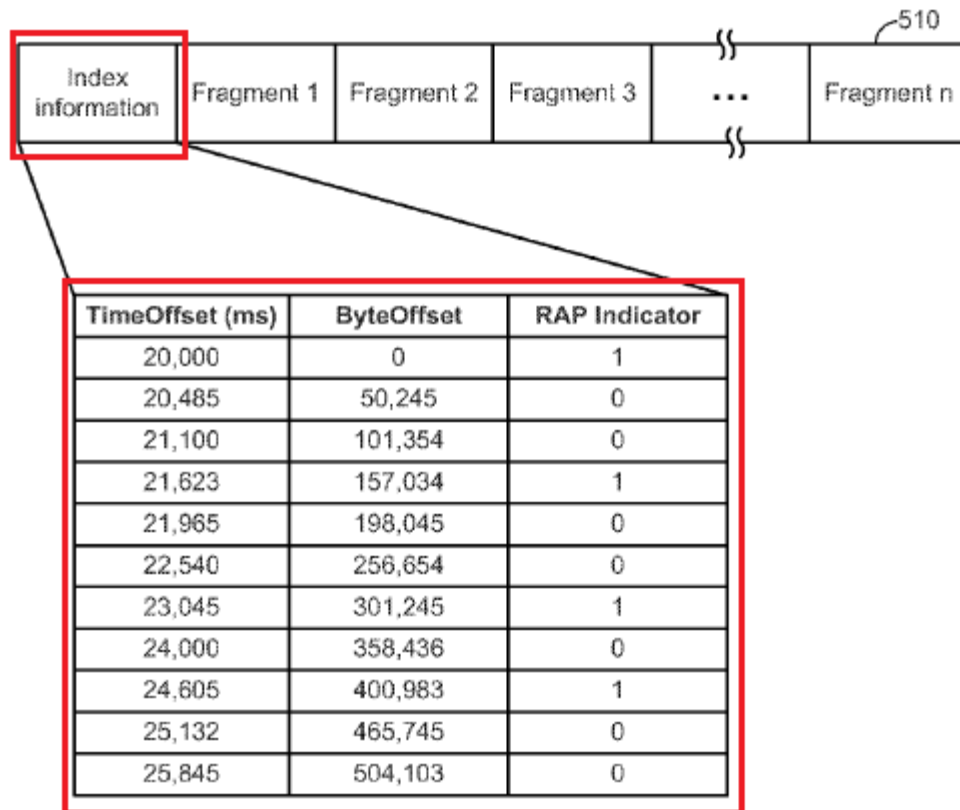


FIG. 6
(ANNOTATIONS ADDED)

Chen explains that, in this example, a representation of a media stream is contained in one or more continuous media files called “media segments”. *See* Ex. 1006, ¶[0149] (“the original source data for one representation of a media stream may be contained in one or more media files herein called a ‘media segment’, wherein each media segment contains the media data used to playback a continuous time segment of the media”). “Within each segment, either at the beginning or spread throughout the source segment, there can also be indexing information, which comprises a time/byte-offset segment map.” Ex. 1006, ¶[0150]. Chen teaches one

way for providing byte ranges using Figure 6, where the time/byte-offset pairs provide time ranges and byte ranges corresponding to the fragments included in the media segment. *See id.* (“T(i-1) represents a start time within the segment for playback of the i-th fragment..., T(i) represents an end time for the i-th fragment (and thus the start time for the next fragment), and the byte-offset B(i-1) is the corresponding byte index of the beginning of the data within this source segment..., and B(i) is the corresponding end byte index of the i-th fragment (and thus the index of the first byte of the next fragment).”). So, for example, Fragment 2 of the media segment illustrated in FIG. 6 has beginning byte index of B(1), (byte offset of 50,245), and an end byte index of B(2) (byte offset 101,354), which is the index of the first byte of the next fragment, thus Chen teaches that the byte range for Fragment 2 is 50,245-101,353. *See, e.g.,* Ex. 1006, ¶¶[0150], [0197] (“The blocks for a given version of the content do not overlap and every frame of the content is contained within exactly one block of each version.”). Given Chen’s teachings, a POSITA would have found it obvious that each fragment starts at its offset and ends where the next fragment begins, based on the end byte index. *See id.*

172. **Location of Segment Index.** Chen teaches that the segment index/map is typically located at the beginning of the segment file. “Preferably, indexing information, also called ‘segment indexing’ or ‘segment map’, is

generated” and “may be included within the segment, typically at the beginning of the segment (at least some of the segment map is at the beginning) and is often small.” Ex. 1006, ¶[0123]. A POSITA would have been motivated to include segment indices inside segment files because Chen teaches several advantages. “Especially in cases where the segment index is contained in the segment, the receiver ***may download some or all of this segment map quickly*** and subsequently use this to determine the mapping between time offsets and corresponding byte positions of fragments associated with those time offsets within the file.” *Id.* ¶[0123] (emphasis added).

173. To the extent PO argues that Chen’s segment index may also be in a separate file, that would still render this limitation obvious. First, the ’588 patent states that the container index does not have to be in the container file and can instead be in a top-level index file or in a separate index file. *See, e.g.*, Ex. 1001, 10:11-14, 20:27-37, 23:38-42. Second, the claim limitation only recites an index *associated* with a container file. Chen teaches downloading a segment index/map that provides information, including URLs and byte ranges for accessing segments. *See, e.g.*, Ex. 1006, ¶¶[0119-0123, 0129, 0147-0150, 0201], FIG. 6. Thus Chen teaches that the segment index/map is associated with the segment for which the URL and byte ranges are provided.

174. Additionally, A POSITA would also have found it obvious for the video to be protected through encryption, including by partially encrypting its video frames. *See infra* Section VIII.A.2.[1d].

[1k] requesting portions of the selected stream of protected video based on the provided byte ranges;

175. The combination teaches or suggests requesting portions (e.g., fragments) of the selected stream (e.g., representation) of protected (e.g., partially encrypted) video based on the provided byte ranges. *See, e.g.*, Ex. 1006, ¶¶[0033, 0063-0064, 0070-0072, 0074, 0083, 0111, 0119-0123, 0129, 0147-0150, 0157-0158, 0175-0187, 0197, 0201, 0216-0217, 0219-0267, 0517, 0530], FIG. 5, 6, 9(a), 27, Ex. 1007, ¶¶[0010, 0033, 0055-0061], FIG. 5A-5C, 6A-6B. Chen teaches that, after selecting a representation (*see supra* Section VIII.A.2.[1i]) and receiving a segment index for an associated segment (*see supra* Section VIII.A.2.[1j]), the client requests a fragment of the associated segment based on byte ranges provided by the segment index, which is stored as metadata at the beginning of the file. *See, e.g.*, Ex. 1006, ¶¶[0070] (“The client may then request an initial portion of the file to learn the media metadata (which typically is stored in the Movie header box, also referred to as “moov” box) together with movie fragment times and byte offsets. **The client may then issue HTTP partial get requests to obtain movie fragments as required.**”) (emphasis added), [0147] (“Multiple fragments may be

concatenated into a single segment and **clients may issue requests that identify the specific byte range within the segment that correspond to the required fragment** or subset of the fragment.”) (emphasis added), [0517] (“The client may, for example, initially request the ‘tidx’ or ‘sidx’ box from the beginning of the media segment using byte range requests. **By use of the ‘tidx’ or ‘sidx’ boxes, segment timing can be mapped to byte ranges of the segment. By continuously using partial HTTP requests, only the relevant parts of the media segment need be accessed**, for improved user experience and low start-up delays.”) (emphasis added), FIG. 5:

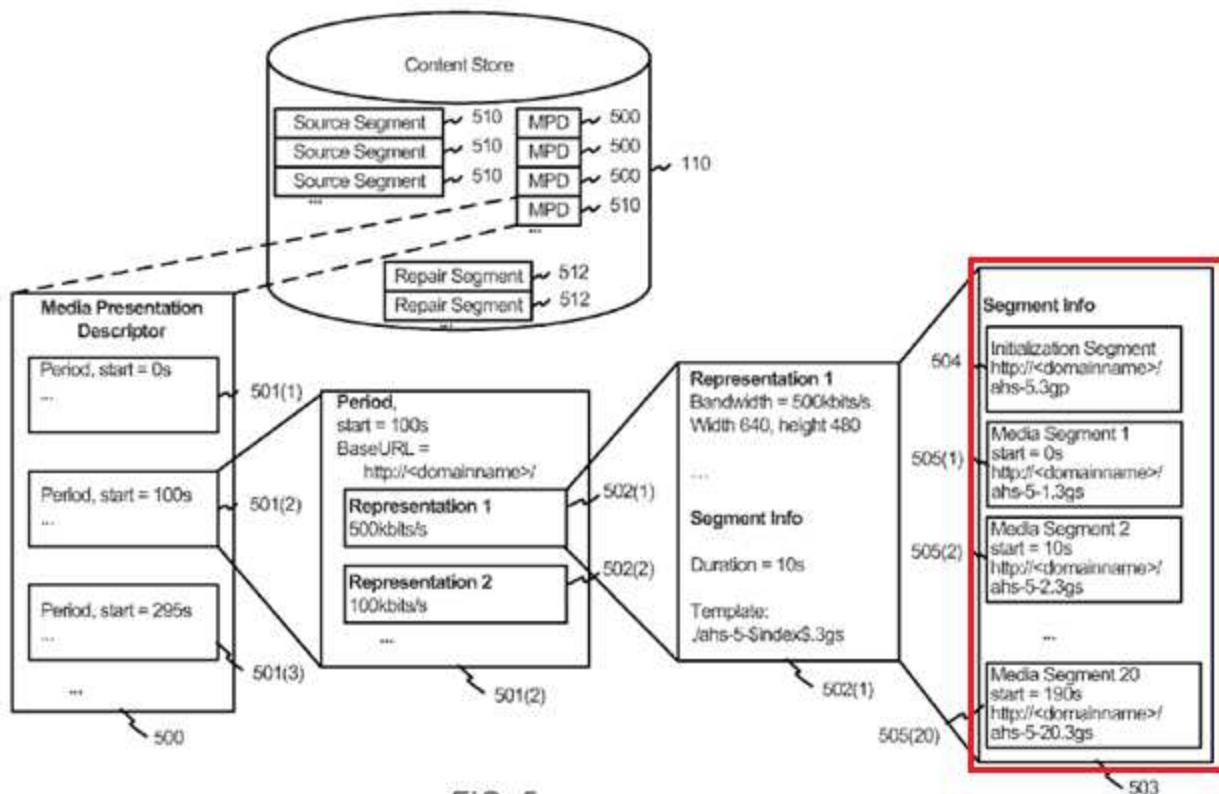


FIG. 5
(ANNOTATIONS ADDED)

176. Chen teaches requests for fragments (or portions of fragments) are made using partial HTTP requests, or HTTP partial get requests that were established as part of the HTTP standards. *See, e.g.*, 1006, ¶¶[0070] (“The client may then request an initial portion of the file to learn the media metadata (which typically is stored in the Movie header box, also referred to as “moov” box) together with movie fragment times and byte offsets. **The client may then issue HTTP partial get requests to obtain movie fragments** as required.”) (emphasis added), [0074] (“Based on the available indexing, **a client can request different portions of a fragment in different HTTP requests, or can request one or more consecutive fragments or portions of fragments in one HTTP request.**”) (emphasis added), ¶[0083].

177. Chen further teaches that fragments are blocks and that both are portions of a segment (e.g., container file)⁴. *See, e.g.*, Ex. 1006, ¶¶[0074] (“**The terms “block” and “fragment” are used synonymously** in this document unless otherwise specified and generally refer to the smallest aggregation of data that is indexed.”) (emphasis added), [0147] (“**Multiple fragments may be concatenated into a single segment** and clients may issue requests that identify the specific byte

⁴ As discussed in Section VIII.A.2.[1j], Chen teaches that segments are files, and Chen teaches that segments are also containers. *See* Ex. 1006, ¶¶[0072], [0119]. A POSITA would have understood that Chen teaches that segments are container files. *See supra* Section VIII.A.2.[1j].

range within the segment that correspond to the required fragment or subset of the fragment.”) (emphasis added), FIG. 27:

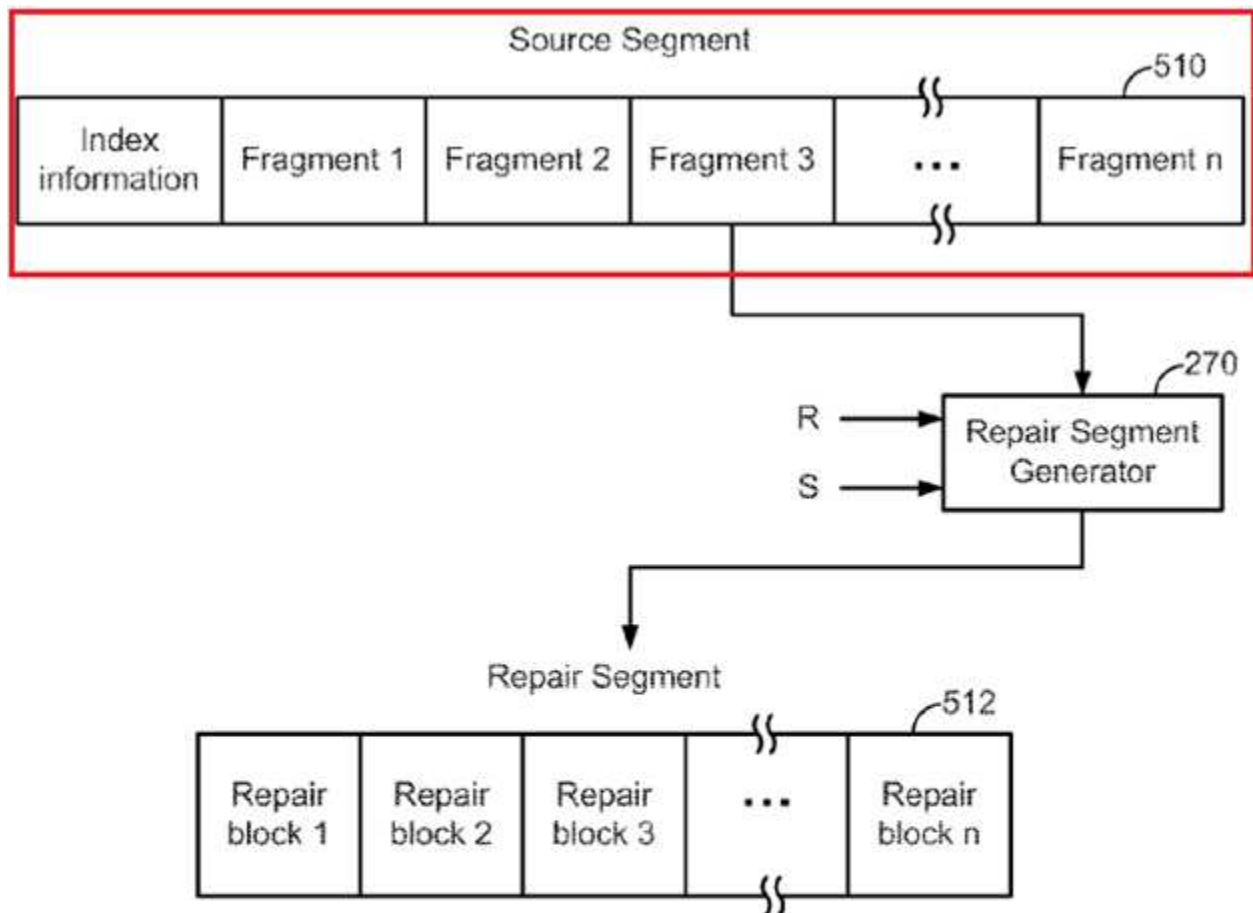


FIG. 27
(ANNOTATIONS ADDED)

178. Chen teaches that a segment file contains a container index at the front, followed by a series of fragments. See Ex. 1006, ¶[0071] (“In case that the segment format is based on the 3GP file format, then segments contain non-

overlapping time slices of the movie fragments, called ‘time-wise splitting.’”), Figs. 6, 27. Therefore, Chen teaches that fragments/blocks are portions of a segment. Further, it would have been obvious to a POSITA that concatenating multiple fragments into a segment, as taught by Chen, teaches that a segment includes multiple fragments and therefore, the fragments are portions of the segment. A POSITA would additionally have found it obvious that portions of fragments/ blocks would be portions of segments because fragments/blocks are themselves portions of segments. Thus, a POSITA would have understood that Chen’s teaching of requesting a fragment/block, or a portion of a fragment/block all teach requesting a portion of a segment, which is a container file. *See supra* Section III.A.2.[1j]. When Chen teaches requesting a portion of a segment, Chen teaches requesting a portion of the selected stream because a segment is a portion of the selected stream (e.g., representation). *See supra* Section VIII.A.2[1i]-[1j]. As discussed in Section VIII.A.2[1i]-[1j], the combination teaches selecting a representation from a plurality of alternative representations of protected video and receiving a segment index that provides byte ranges for fragments or blocks of the selected representation of protected video within an associated segment. *See supra* Section VIII.A.2[1i]-[1j]. It would have been obvious to a POSITA that a request for a fragment of a segment, as taught by Chen, would be a request for a

portion of a selected representation of a partially encrypted video. Thus, it would have been obvious to a POSITA that Chen teaches this limitation.

179. Chen further explains that a request for fragments of a segment are based on byte ranges provided by a segment index. *See, e.g.*, Ex. 1006, ¶¶[0129] (“Segment metadata may **include byte range** versus time information **for requests within a segment** and identification of Random Access Points (RAPS) or other seek points, where **some or all of this information may be part of a segment indexing or segment map.**”) (emphasis added), [0517] (“The client may, for example, initially request the ‘tidx’ or ‘sidx’ box from the beginning of the media segment using byte range requests. **By use of the ‘tidx’ or ‘sidx’ boxes, segment timing can be mapped to byte ranges of the segment. By continuously using partial HTTP requests, only the relevant parts of the media segment need be accessed,** for improved user experience and low start-up delays.”) (emphasis added), [0529] (“With the No-FEC method, portions of the same fragment are requested over different connections, e.g., using HTTP byte range requests for portions of the fragment, and thus for example **each HTTP byte range request is for a portion of the byte range indicated in the segment map for the fragment.**”) (emphasis added), [0530] (“With the FEC method, portions of the same fragment and FEC repair data generated from that fragment are requested over different connections, using HTTP byte range requests for portions of the

fragment, and thus for example **each HTTP byte range request is for a portion of the byte range indicated in the segment map for the fragment.**") (emphasis added). Chen thus teaches that requests, such as HTTP byte range requests, for fragments of a segment are requests based on byte ranges provided in a segment index/map of the segment. Additionally, a POSITA would have been motivated to request portions of the selected stream of protected video based on the provided byte ranges using HTTP, because (1) Chen teaches this, as explained above, (2) they facilitate byte range requests in known HTTP partial get requests, and (3) it was widely known that "[a]daptive streaming solutions typically utilize[d] ... Hypertext Transfer Protocol (HTTP)... to stream media between a server and a playback device. HTTP is a stateless protocol that enables a playback device to request a byte range within a file." See Ex. 1001 2:1-11. Thus, the combination teaches using HTTP to request portions of the selected stream of protected video based on the provided byte ranges provided, e.g., by the segment index sidx.

180. I note that the patent owner admits this limitation is met. In its Amended Complaint, PO alleged this limitation [1k] is met by "us[ing], for example, the sidx box to make HTTP byte range requests for content." Ex. 1015, ¶430. Therefore PO admits this limitation is satisfied by the combined teachings of the references.

181. **Hurst.** Hurst further renders this obvious with analogous teachings “where streamlet requests may comprise requesting pieces of any streamlet file.” Ex. 1008, 14:6-11. “Splitting the streamlet 304 into smaller pieces or portions beneficially allows for an increased efficiency potential, and also eliminates problems associated with multiple full-streamlet requests sharing the bandwidth at any given moment.” *Id.*; *see also id.*, 15:10-14 (“A single streamlet request may be issued for an entire streamlet, or multiple requests may be issued, each for a different part or portion of the streamlet. If the streamlet is requested in several parts, the parts may be recombined by the client module 114 streamlet.”). Additionally, a POSITA would also have found it obvious for the video to be protected through encryption, including by partially encrypting its video frames. *See supra* Section VIII.A.2[1d]. Given the teachings of the references, the combination renders this limitation obvious.

[11] locating encryption information that identifies encrypted portions of frames of video within the requested portions of the selected stream of protected video;

182. The combination teaches or suggests locating encryption information (e.g., encryption parameters) that identifies encrypted portions of frames of video within the requested portions of the selected stream (e.g., representation) of protected (e.g., partially encrypted) video. *See, e.g.*, Ex. 1007, ¶¶[0010, 0033, 0055-0061, 0072], FIG. 5A-5C, 6A-6B. Lindahl teaches using encryption

parameters to identify encrypted portions and unencrypted portions. *See, e.g.*, Ex. 1007, ¶¶[0056] (“Partial encryption and the use of encryption parameters is discussed in additional detail below with reference to FIGS. 6A and 6B. The encryption parameters control the portion of a block that is to be encrypted. The remaining portions of a block that are not encrypted remain unencrypted (i.e., ‘in the clear’).”), FIGS. 6A-6B:

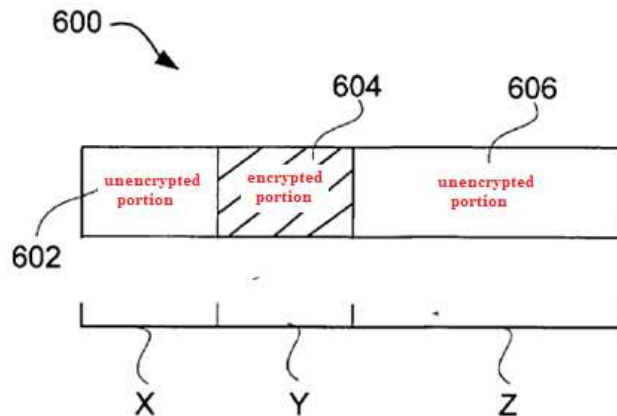


FIG. 6A
(ANNOTATIONS ADDED)

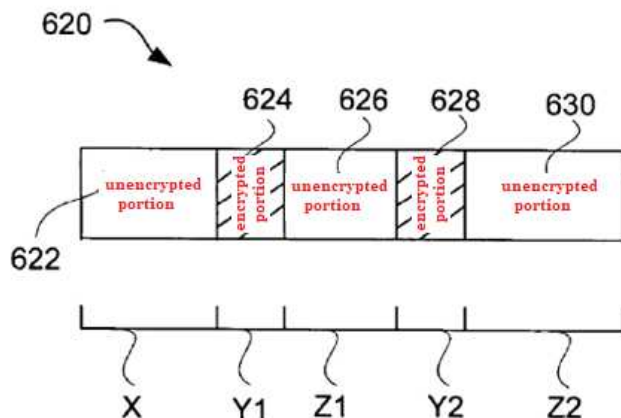


FIG. 6B
(ANNOTATIONS ADDED)

183. Lindahl teaches encryption parameters, including X, Y and Z, that identify encrypted portions of video frames. *See* Ex. 1007, ¶¶[0056-0059]. For Figure 6A, variable X denotes an offset into the block (location of the encrypted portion) by specifying the size of an initial unencrypted portion, thereby providing

the location of the encrypted portion 604; Y denotes the size of the encrypted portion 604; and Z denotes the size of the unencrypted portion 606 at the end of the block. *See also id.*, ¶[0057] (“FIG. 6A is a diagram of a partially encrypted block 600 according to one embodiment. The partially encrypted block 600 has an initial unencrypted portion 602, followed by an encrypted portion 604, and followed by an unencrypted portion 606. **The size of the unencrypted portion 602 is denoted by the variable X, the size of the encrypted portion 604 is denoted by the variable Y, and the size of the unencrypted portion 606 is denoted by the variable Z.** The size of the partially encrypted block 600 is the sum of the variables X, Y and Z. ... in one representative embodiment, the variables X, Y and Z can be sized as 16 bytes, 64 bytes and 512 bytes, respectively.”) (emphasis added), [0059] (“The variable X denotes the size of the initial unencrypted portion 602 or 622 for the partially encrypted block 600 or 620. This initial portion, having its size denoted by the variable X, provides an offset into the partially encrypted block 600 or 620. This offset causes the encryption portion to begin somewhere internal to the block. Typically, a block will include various standard header information at the beginning of each block; hence, this type of information is generally known or easily determined.”). Lindahl’s first embodiment 6A contains one encrypted portion per block. *See id.* FIG. 6A.

184. Lindahl teaches another embodiment in FIG. 6B with more than one encrypted portion of a block. *See id.*, FIG. 6B. As in the previous embodiment, encryption parameters X, Y, and Z identify the encrypted portions of the video frame, with variable Y expanded into Y1 and Y2, and Z expanded into Z1 and Z2 to indicate the size and positions of the additional encrypted portions. *See id.*, ¶[0058] (“FIG. 6B illustrates a partially encrypted block 620 according to another embodiment. The partially encrypted block 620 includes an unencrypted portion 622, a first encrypted portion 624, an unencrypted portion 626, a second encrypted portion 628, and a final unencrypted portion 630. In this example, there are two separate parts of the partially encrypted block 620 that are encrypted. **The unencrypted portion 622 has a length (e.g., number of bits or bytes) denoted by the variable X, the first encrypted portion 624 has a length indicated by the variable Y1, the unencrypted portion 626 has a length indicated by the variable Z1, the second encrypted portion 628 has a length indicated by the variable Y2, and the unencrypted portion 630 has a length indicated by the variable Z2.** As compared to the partially encrypted block 600, the overall percentage being encrypted can be the same. In other words, the sum of the variables Y1 and Y2 can equal the variable Y, and the sum of the variables Z1 and Z2 can equal the variable Z.”) (emphasis added).

185. Both embodiments satisfy the challenged claims because both include partially encrypted video frames with encrypted portions and unencrypted portions of data. *See supra* Sections VIII.A.2.[1d, 1f]. Both embodiments further teach encryption information (e.g., encryption parameters) that identify encrypted portions of frames of video within the requested portions of the selected stream of protected video, as explained above. Lindahl teaches that a video frames are divided into blocks. *See* Ex. 1007, ¶¶[0055, 0060-0061]; *supra* Section VIII.A.2.[1f]. For both embodiments, Lindahl's encryption parameters identify the encrypted portions of each block. Therefore, Lindahl teaches encryption parameters that identify encrypted portions of frames.

186. A POSITA would have been motivated to use encryption parameters, as taught by Lindahl, to control which portions of a block are encrypted because it allows for flexibility in what portions are encrypted and what portions are left unencrypted. *See* Ex. 1007, ¶¶[0057-0058]. This facilitates different levels of security, for example, by encrypting larger amounts of each frame to increase protection, or encrypting less to reduce computational load. Furthermore, a POSITA would have been motivated to encrypt two or more portions of each block, as taught by FIG. 6B, to facilitate the use of multiple keys, as explained below for Claim 11. *See infra* Section VIII.A.12.

187. Lindahl teaches that the encryption parameters are retrieved to decrypt the video media, which is divided into frames and then blocks—each block is decrypted in accordance with the encryption parameters. *See* Ex. 1007, ¶[0072] (“After a media file to be decrypted has been identified 902, encryption parameters associated with the media file are retrieved 904. The encryption parameters can be the same parameters that were utilized with respect to the media file encryption process 400 illustrated in FIG. 4.”). Furthermore, a POSITA would have found it obvious and been motivated to retrieve the encryption parameters because they are needed to identify the encrypted portions of the content so that they may be decrypted efficiently.

[1m] decrypting each encrypted portion of the frames of video identified within the located encryption information using the set of common keys;
--

188. The combination teaches or suggests decrypting each encrypted portion of the frames of video identified within the located encryption information (e.g. encryption parameters) using the set of common keys. *See, e.g.*, Ex. 1007, ¶¶[0010, 0033, 0055-0061, 0066, 0070, 0072], FIG. 5A-5C, 6A-6B. First, Lindahl teaches that the encryption parameters are located and retrieved to decrypt the video stream, which is divided into frames and then blocks—each block is decrypted using the encryption parameters, which identify the encrypted portions of each frame. *See* Ex. 1007, ¶[0072] (“After a media file to be decrypted has

been identified 902, encryption parameters associated with the media file are retrieved 904. The encryption parameters can be the same parameters that were utilized with respect to the media file encryption process 400 illustrated in FIG. 4. The encryption parameters are again utilized to decrypt the media file that was previously encrypted. These encryption parameters can also be referred to as decryption parameters. In other words, the parameters for encryption and decryption are the same. Next, the media file is examined 906 to locate frame boundaries within the media data. Then, each frame is divided 908 into a plurality of blocks. Each block is then decrypted 910 in accordance with the encryption parameters.”), [0056-0059]; *supra* Section VIII.A.2.[11]. The encryption parameters identify the encrypted portions of frames of video. *See id.*

189. Lindahl further teaches that decryption is performed using the keys that were used to encrypt the video stream via symmetric key encryption. For example, when a downloaded file was encrypted using a cryptographic key, Lindahl teaches decrypting it with the same cryptographic key. *See, e.g.,* Ex. 1007, ¶¶[0066] (“Once the media file is downloaded 708, the media file can then be decrypted 710 at the user's local machine using the cryptographic key. The global key or other cryptographic key may also be used. If the metadata of the media file itself is encrypted, the metadata can also be decrypted using the global key which has previously been provided to the user who has purchased the access rights to the

media file.”), [0070] (“Next, the encrypted media file can be decrypted 808 using the associated encryption key. At this point, the media portion of the media file is in the ‘clear.’”).

190. As explained above, the combination teaches that the content was encrypted using a set of common keys. *See supra* Section VIII.A.2[1e]. The combination also teaches encrypting the video frames using symmetric encryption, meaning the same keys used to encrypt data are also used to decrypt the data. *See supra* Section VIII.A.2.[1g]. Therefore, because the video frames were symmetrically encrypted using a set of common keys, a POSITA would have found it obvious to decrypt the video frames using the same set of common keys that were used to encrypt the video frames—in this context, the set of common keys would be the media file’s “associated encryption key[s]” referenced by Lindahl. *See* Ex. 1007, ¶[0070]. As a POSITA would have understood, this is how symmetric encryption works. *See supra* Section III. Therefore, the combination renders obvious the step of decrypting each encrypted portion of the frames of video identified within the located encryption information using the set of common keys.

191. Additionally, in this context, because portions of each frame are encrypted, a POSITA would have found it obvious to decrypt those encrypted portions, as Lindahl teaches. *See also* Ex. 1007, ¶[0072]. Further, Lindahl teaches

located encryption information that identifies encrypted portions of frames of video. *See infra* Section VIII.A.3.[2b]. A POSITA would have understood that encrypted portions of frames of video were identified so that they could be effectively decrypted by targeting decryption towards the encrypted portions that were identified. Therefore, a POSITA would have found it obvious to decrypt each encrypted portion of the frames of video identified within the located encryption information in order to effectively decrypt the frames of video and enable playback of the content. Thus, based on Lindahl’s teaching of identifying encrypted portions of frames of video within encryption information, a POSITA would have found it obvious to decrypt each encrypted portion of frames of video identified within the located encryption information.

[1n] playing back the decrypted frames of video obtained from the requested portions of the selected stream of protected video.

192. The combination teaches or suggests playing back the decrypted frames of video obtained from the requested portions of the selected stream of protected video. *See, e.g.*, Ex. 1006, ¶¶[0033, 0063-0064, 0070-0072, 0074, 0083, 0111, 0119-0123, 0127, 0129, 0147-0150, 0157-0158, 0175-0187, 0197, 0201, 0216-0217, 0219-0267, 0517, 0530], FIG. 5, 6, 9(a), 27, Ex. 1007, ¶¶[0010, 0033, 0055-0061, 0066, 0070, 0072], FIG. 5A-5C, 6A-6B. For example, Chen’s teachings “enable[] clients to request blocks of the highest media quality that may

be received in time ***to be played out with continuity.***” See, e.g., Ex. 1006, ¶[0127] (emphasis added). Chen teaches that media delivery systems send content to users, who then play back and display the content on their devices. See *id.* ¶¶[0016]-[0019]. “As described herein, a goal of a streaming system is to move media from its storage location ... to a location where it is being consumed, i.e., presented to a user or otherwise ‘used up’ by a human or electronic consumer. Ideally, the streaming system can provide uninterrupted playback ... at a receiving end and can begin playing a stream or a collection of streams shortly after a user has requested the stream or streams.” *Id.* ¶[0061]. Playing back streams for users is one of the goals of a streaming system. *Id.*

193. Hurst further renders this obvious by teaching that the client device plays back the content. See Ex. 1008, 1:15-19 (“Many portable media players are equipped with network connections and enabled to play music or videos....”), 6:14-22 (“The content module 112 may be configured to receive, process, and store content. In one embodiment, the processed content is received by a client module 114 configured to play the content on the end user station 104. In a further embodiment, the client module 114 is configured to receive different portions of a content stream from a plurality of locations simultaneously. For example, the client module 114 may request and receive content from any of the plurality of web servers 116.”). Thus, given the combined teachings of the references, it would

have been obvious, after decrypting frames of video obtained from the requested portions of the selected stream of protected video, as explained in the steps above, to playback those decrypted frames of video because playing back the content for the user was the purpose of Chen, Hurst, and Lindahl's teachings. *See also*, Ex. 1007, ¶¶[0005], [0011].

3. Claim 2

2[a] The playback device of claim 1, wherein the step of requesting portions of the selected stream further comprises the step of obtaining a container file containing protected video from at least one of the plurality of alternative streams,

194. The combination teaches or suggests the playback device and the steps of claim 1 and further teaches or suggests obtaining a container file (e.g., segment) containing protected (e.g., partially encrypted) video from at least one of the plurality of alternative streams (e.g., representations). *See, e.g.*, Ex. 1006, ¶¶[0033, 0063-0064, 0070-0072, 0074, 0083, 0111, 0119-0123, 0127, 0129, 0133, 0147-0150, 0157-0158, 0175-0187, 0197, 0201, 0216-0217, 0219-0267, 0517, 0530], FIG. 5, 6, 9(a), 27, Ex. 1007, ¶¶[0010, 0033, 0055-0061, 0066, 0070, 0072], FIG. 5A-5C, 6A-6B. For example, Chen teaches that after a representation is selected, the client obtains associated segments for that representation, which are stored as container files on the server. *See* §VIII.A.2[1i]-[1j]; Ex. 1006, ¶¶[0063-0064]. Chen teaches that “[t]he terms ‘file’ and ‘segment’ are used synonymously” and further teaches that a segment can be “obtained as a result of a

file download protocol request, including for example an HTTP request.” Ex. 1006, ¶[0072]. A POSITA would have found it obvious and been motivated to obtain a container file using a file download protocol request, such as an HTTP request, because (1) Chen teaches this, as explained above, and (2) HTTP requests were widely utilized for adaptive streaming solutions to request and obtain files. *See* Ex. 1001, 2:1-11. It would have been obvious to a POSITA for the container file to contain protected video from at least one of the plurality of alternative streams based on teachings by Chen, Lindahl, and Hurst, as described above. *See supra* Section VIII.A.2.[1d-1k].

195. As explained for claim 1, it would have been obvious to a POSITA to use HTTP requests to obtain a file. *See supra* Section VIII.A.2.[1k]. Chen teaches requesting a whole segment file, which a POSITA would have been motivated to do, because a request for a whole segment file allows a content delivery network (CDN) to cache the whole segment file. *See* Ex. 1006, ¶[0133] (“Most CDNs will, however, serve range requests from cache if the data is available. It may therefore be advantageous to have some portion of the client requests be for a whole segment file. These requests can later be cancelled if necessary.”). Caching the whole segment file allows the CDN to deliver portions of the cached segment file in response to range requests, such as partial HTTP requests. *See id.* A POSITA would have understood that delivering cached data increases efficiency. *See id.*

Therefore, Chen teaches obtaining a container file containing protected video from at least one of the plurality of alternative streams.

196. Additionally, a POSITA would have found it obvious to obtain a file in response to an HTTP request, because HTTP GET requests most frequently return files. *See supra* Section IV.A. Therefore, when a fragment is requested, as described above, it would have been obvious to obtain a file in response – it would have been obvious for this to have been a container file containing requested video. *See supra* Section VIII.A.2.[1k]. Thus, the combination teaches the playback device and the steps of claim 1 and further teaches obtaining a container file containing protected video from at least one of the plurality of alternative streams.

[2b] where the container file also includes encryption information that identifies portions of frames of video that are encrypted and
--

197. The combination teaches or suggests that the container file also includes encryption information (e.g., encryption parameters) that identifies portions of frames of video that are encrypted. *See, e.g.*, Ex. 1006, ¶¶[0063-0064, 0066, 0070-0074, 0111, 0119-0123, 0129, 0147-0150], Ex. 1007, ¶¶[0010, 0033, 0055-0061, 0066, 0070, 0072]. As explained above, Lindahl teaches encryption parameters that identify encrypted portions of video frames. *See supra* Section VIII.A.2.[1l]. Lindahl teaches that these encryption parameters can “be provided on a per file basis”, including “in the media file itself.” Ex. 1007, ¶[0053]. A

POSITA would have been motivated to do so because, for example, it provides flexibility, particularly for adaptive streaming situations, to account for varying bitrates for each stream, given that the encryption parameters are provided in units of bytes. *See, e.g., id.* ¶¶[0052] (“frame size can vary... depending on target resolution.”), [0057]. For example, this would allow for encryption of corresponding regions of blocks in alternative streams with different bitrates. Furthermore, Lindahl teaches that “the encryption parameters could be provided on a per frame or block basis, e.g., with each frame or block potentially having a different set of parameters.” *Id.* ¶[0053]. A POSITA would have been motivated to apply this teaching to improve the balance between computation and security by tailoring the parameters based on the data, such as encrypting higher or lower percentages of i-frames. Because encryption parameters vary between frames or blocks within a video, it would have been obvious to store the encryption parameters “in the media file itself.” *See id.* Thus, the combination teaches that the container file also includes encryption information (e.g., encryption parameters) that identifies portions of frames of video that are encrypted.

[2c] a reference to at least one key from the set of common keys to utilize in accessing the encrypted portions of the frames of video.
--

198. The combination teaches or suggests a reference to at least one key from the set of common keys to utilize in accessing the encrypted portions of the

frames of video. *See, e.g.*, Ex. 1007, ¶¶[0008, 0017, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084]. Lindahl teaches locating the appropriate cryptographic key for decryption and playback. *See* Ex. 1007, ¶[0083] (“[I]t should be understood that since the media items are secured (i.e., encrypted), the play module 1032 must first locate the appropriate encrypted cryptographic key before a given media item can be played.”). Thus Lindahl teaches that the cryptographic key must be identified, and a POSITA would have been motivated and found it obvious for Lindahl’s encryption parameters to include a reference to this key, for example a key ID, to indicate which key was used to encrypt which block (because Lindahl’s use of symmetric encryption requires the same key to be used for decryption). *See* Ex. 1007, ¶[0008]; *see also supra* Section VIII.A.2.[1g]. Lindahl teaches that blocks are decrypted using one or more cryptographic keys, in accordance with the encryption parameters; thus it would have been obvious to store a reference to the cryptographic key in the encryption parameters so that it may be used for decryption. *See* Ex. 1007, ¶[0072] (emphasis added):

The encryption parameters can be the same parameters that were utilized with respect to the media file encryption process 400 illustrated in FIG. 4. **The encryption parameters are again utilized to decrypt the media file that was previously encrypted.** These encryption parameters can also be referred to as decryption parameters. In other words, the parameters for

encryption and decryption are the same. ... Each block is then decrypted 910 in accordance with the encryption parameters. In addition, it should be recognized that the decryption of each block also makes use of one or more cryptographic keys...

Furthermore, a POSITA would have been motivated to include a key reference, such as key ID, in the encryption parameters because it facilitates key management. It allows a server with many files to quickly identify and confirm the encryption keys for each file. If key IDs were not included in encrypted video files, the server could potentially lose track of the encryption keys for those files, and it would be prohibitively difficult to remap all the encrypted video files to their corresponding keys. That is why it is a common practice to include key IDs with encrypted files for video streaming DRM applications.

199. Additionally, as explained for claims 1 and 11, the combination teaches encrypting the alternative streams using a common set of keys comprising a plurality of keys. *See supra* Section VIII.A.2.[1e]; *infra* Section VIII.A.12. This includes using more than one key to encrypt and decrypt blocks within the video stream, as taught by Lindahl. *See* Ex. 1007, ¶¶[0054] (“[T]he encryption process also makes use of one or more encryption keys when encrypting each block.”), ¶[0072] (“[T]he decryption of each block also makes use of one or more cryptographic keys...”). When more than one key is used to encrypt a video file, a

POSITA would have been motivated and found it obvious for Lindahl's encryption parameters to include references, such as key IDs, to indicate which key was used to encrypt which block, because Lindahl's use of symmetric encryption requires the same key to be used for decryption. *See* Ex. 1007, ¶[0008]; *see also supra* Section VIII.A.2.[1g]. Otherwise, the decryption process would need to guess at which key was used for each block, which can become prohibitively inefficient when multiple keys are used. Lindahl teaches the use of encryption parameters in the context of decrypting using one or more cryptographic keys, and thus a POSITA would have found it obvious to store references to those keys in the encryption parameters. *See id.* ¶[0072] ("Each block is then decrypted 910 in accordance with the encryption parameters. In addition, it should be recognized that the decryption of each block also makes use of one or more cryptographic keys..."). Thus, the combination teaches a reference to at least one key from the set of common keys to utilize in accessing the encrypted portions of the frames of video.

4. Claim 3

3	The playback device of claim 1, wherein the located encryption information comprises a reference to the start of an encrypted block of data.
----------	---

200. The combination teaches or suggests the playback device of claim 1 and further teaches or suggests that the located encryption information (e.g.,

encryption parameters) comprises a reference to the start of an encrypted block of data (e.g., offset). *See, e.g.*, Ex. 1007, ¶¶[0008, 0017, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084]. As explained above, Lindahl teaches locating encryption parameters that identify encrypted portions of video frames, including variable X which is an offset to where an encrypted portion of data begins. *See supra* Section VIII.A.2.[11]; Ex. 1007, ¶¶[0059] (“This initial portion, having its size denoted by the variable X, provides an offset into the partially encrypted block 600 or 620. This offset causes the encryption portion to begin somewhere internal to the block. ... the partial encryption can be more effective when an initial offset into the block is utilized such as provided by the encryption parameter associated with the variable X.”), [0057-0058]. This offset is a reference to the start of an encrypted block of data because it points to the start of the encrypted block. *See id.* ¶¶[0057-0058], FIGS. 6A-6B:

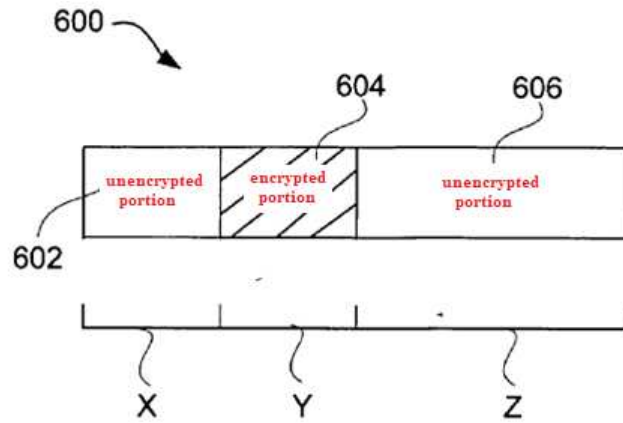


FIG. 6A

(ANNOTATIONS ADDED)

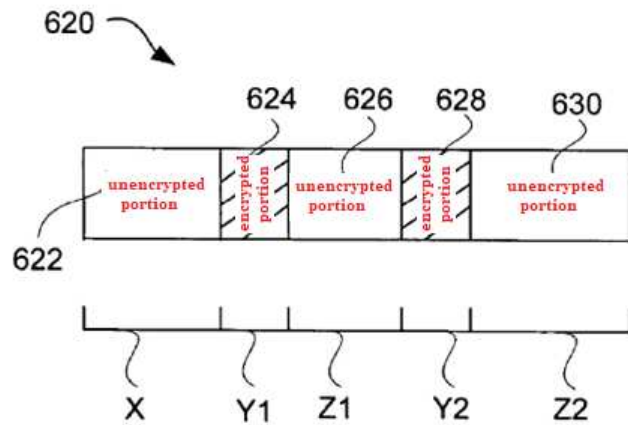


FIG. 6B

(ANNOTATIONS ADDED)

5. Claim 4

4	The playback device of claim 3, wherein the located encryption information further comprises the size of the encrypted block of data.
----------	--

201. The combination teaches or suggests the playback device of claim 1 and further teaches or suggests that the located encryption information (e.g., encryption parameters) further comprises the size of the encrypted block of data, including variable Y. As explained above, Lindahl teaches locating encryption parameters that identify encrypted portions of video frames, including a size of an encrypted portion of data. *See supra* Section VIII.A.2.[11], VIII.A.3[2b]. For example, as explained above, Lindahl teaches “the size of the encrypted portion 604 is denoted by the variable Y...” Ex. 1007, ¶[0057], FIGS. 6A-6B.

6. Claim 5

5	The playback device of claim 3, wherein the located encryption information further comprises cryptographic information that can be utilized to access the encrypted portion of the frame.
----------	--

202. The combination teaches or suggests the playback device of claim 3 and further teaches or suggests that the located encryption information (e.g., encryption parameters) further comprises cryptographic information that can be utilized to access the encrypted portion of the frame. For example, the combination teaches that encryption parameters that comprise the size and location of the encrypted portions of data, which are used to access it. *See supra* Sections

VIII.A.2.[11], VIII.A.3[2b], VIII.A.4-5. Furthermore, as described above, the combination teaches that the encryption parameters include references, such as key IDs, to the keys in the set of common keys to be used for decryption, which can be utilized to access the encrypted portion of the frame. *See supra* Section VIII.3.[2c]. Dependent claim 6, discussed below, confirms that the recited “cryptographic information” can be satisfied by a reference to at least one key from the set of common keys—as explained above, this is rendered obvious by the combination.

7. Claim 6

6	The playback device of claim 5, wherein the cryptographic information is a reference to at least one key from the set of common keys.
----------	--

203. The combination teaches or suggests the playback device of claim 5 and further teaches or suggests that the cryptographic information is a reference to at least one key from the set of common keys. *See, e.g.*, Ex. 1007, ¶¶[0008, 0017, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084]. As described above, the combination teaches that the encryption parameters include a reference, for example a key ID, to at least one key from the set of common keys. *See supra* Section VIII.3.[2c]. Thus, the combination teaches the playback device of claim 5 and further teaches that the cryptographic information is a reference to at least one key from the set of common keys.

8. Claim 7

7[a]	The playback device of claim 1, wherein the application is further for causing the set of processors to perform the steps of:
-------------	--

204. The combination teaches or suggests the playback device of claim 1.

See supra Section VIII.A.2.

[7b]	detecting a change in the streaming conditions;
-------------	--

205. The combination teaches or suggests detecting streaming conditions for the playback device to select a stream based on the detected streaming conditions. *See supra* Section VIII.A.2.[1h]-[1j]. For example, Chen teaches detecting changes in the available bandwidth so that a stream with a matching data rate can be selected. *See, e.g.*, Ex. 1006, ¶¶[0081] (“This selection may be based on any number of criteria, including **matching of the media data rate to the available bandwidth, even when the available bandwidth is changing over time**, matching of the media resolution or decoding complexity to client capabilities or configuration, or matching to user preferences.”) (emphasis added), [0063-0065], [0121]. Further, “buffer occupancy and network measurements” may trigger the client “switch[ing] to different data rates based on available bandwidth.” *Id.*, ¶¶[0063-0065]. It would have been obvious for changes in network bandwidth to trigger the client to switch to representations of different bitrates. Thus, given these teachings, it would have been obvious for the client

device to detect a change in the streaming conditions to facilitate such switches. Patent Owner’s district court complaint states, “Netflix documentation clearly indicates that the Netflix player application (‘client’) detects streaming conditions because it ‘adaptively selects the optimal stream’ and takes ‘current network conditions into account’ during device playback.” Ex. 1015 at 222 ¶427. In support, the complaint excerpts documentation that states, “we generate multiple quality representations at different bitrates (ranging from 100kbps to 16 Mbps) and the Netflix client **adaptively selects the optimal stream given the instantaneous bandwidth.**” *Id.* (emphasis added). Thus, Patent Owner admits that Chen teaches this limitation.

[7c] identifying a second alternative stream of protected video based on the detected change;

206. The combination teaches or suggests selecting a stream from the plurality of alternative streams of protected video based on the detected streaming conditions. *See supra* Section VIII.A.2.[1i]. For example, Chen teaches that video is encoded at multiple bitrates to form different representations, and the client switches to different data rates based on available bandwidth. *See id.*; Ex. 1006, ¶¶[0063-0065]. Thus, Chen teaches switching to a second alternative stream (e.g., a different representation) of protected video based on the detected change in the streaming conditions (e.g., a detected change in available bandwidth). *See supra*

Section [1i]. A POSITA would have found it obvious to identify a second alternative stream in order to switch to it because the stream must be identified before it can be looked up in the MPD and requested by the playback device.

**[7d] receiving a container index that provides byte ranges for portions of the second alternative stream of protected video within a second associated container file;
requesting portions of the second alternative stream of protected video based on the provided byte ranges;**

207. These limitations are the same as limitations [1j-1k] and are obvious for the same reasons. *See supra* Section VIII.A.2.[1j-1k]. While claim [7d] refers to “the second alternative stream” as compared to claim 1, which refers to “the selected stream,” the obviousness analysis from claim 1 still applies. In both situations, the recited stream refers to one of the alternative streams from claim 1, and the claimed steps (e.g., “receiving a container index...” and “requesting portions...” function similarly for each of the alternative streams. Claim 7’s recitation of a “second alternative stream” does not alter the analysis, other than by applying the teachings to a different one of the alternative streams.

[7e] decrypting each encrypted portion of the frames of video from the requested portions of the second alternative stream of protected video using the set of common keys; and

208. As explained for claim 1, the combination teaches or suggests decrypting each encrypted portion of the frames of video identified within the located encryption information using the set of common keys. *See supra* Section

VIII.A.2[1m]. Claim 7 specifies that this decrypting step is applied to “the requested portions of the second alternative stream of protected video.” This does not alter the obviousness analysis from claim 1, other than by applying the teachings to a different one of the alternative streams (the identified second alternative stream) because the decrypting step in claim 1 was also applied to encrypted portions of video frames that were from requested portions of the alternative stream of protected video requested in the preceding steps. *See supra* Section VIII.A.2[1k-1m]. Given that each of the alternative streams is encrypted in the same manner, it would have been obvious to decrypt them in the same manner. *See* Section VI.A.2[1e]. Claim 7’s recitation of a “second alternative stream” does not alter the analysis, other than by applying the teachings to a different one of the alternative streams.

[7f] playing back the decrypted frames of video from the requested portions of the second alternative stream of protected video.

209. This limitation is encompassed within limitation [1n] and is obvious for the same reasons. *See supra* Section VIII.A.2.[1n]. While claim 7 refers to “the second alternative stream” as compared to claim 1, which refers to “the selected stream,” the obviousness analysis from claim 1 still applies. In both situations, the recited stream refers to one of the alternative streams from claim 1, and the claimed steps (e.g., “receiving a container index...” and “requesting

portions...” function similarly for each of the alternative streams. Claim 7’s recitation of a “second alternative stream” does not alter the analysis, other than by applying the teachings to a different one of the alternative streams.

9. Claim 8

8 The playback device of claim 1, wherein the application is further for causing the set of processors to perform the steps of: transmitting a request for content to a set of one or more content distribution servers; and receiving the content from the set of one or more content distribution servers.

210. The combination teaches or suggests that transmitting a request for content to a set of one or more content distribution servers and receiving the content from the set of one or more content distribution servers. Chen teaches requesting streaming media over a channel from one or a plurality of media servers, which a POSITA would have understood to be content distribution servers because they distribute content to client devices. *See, e.g.*, Ex. 1006, ¶¶[0079] (“It is assumed in many of these examples that a client is coupled to a media server or a plurality of media servers and the client requests streaming media over a channel or a plurality of channels from the media server or the plurality of media servers.”), [0072]. Chen teaches the client device requesting streaming media from one or more media servers. *See, e.g.*, Ex. 1006, ¶¶[0018, 0023-0024, 0063, 0079].

211. A POSITA would have been motivated and found it obvious to apply these teachings from Chen and use a plurality of media servers as content

distribution servers because as this was the standard architecture for content distribution networks (“CDN”), which were widely used for Internet websites and streaming media systems. *See, e.g.*, Ex. 1006, ¶[0452] (“For example this is a common method of spreading the load of requests from multiple clients across multiple physical machines. In particular this approach is commonly taken by Content Delivery Networks (CDNs).”). Chen teaches the use of CDNs, and a POSITA would have understood Chen’s media server teachings to be an example of this well-known content-delivery architecture, with the media servers being content distribution servers. Given Chen’s teachings, it would have been obvious for client devices to transmit a request for content to a set of one or more media servers, and then to receive the content from those servers.

212. Hurst further renders this obvious by teaching a client requesting and receiving content from a plurality of web servers. *See, e.g.*, Ex. 1008, 6:17-22 (“In a further embodiment, the client module 114 is configured to receive different portions of a content stream from a plurality of locations simultaneously. For example, **the client module 114 may request and receive content from any of the plurality of web servers 116.**”) (emphasis added), 8:53-57 (“Additionally, since **the client module 114 initiates the request**, the web server 116 is only required to retrieve and serve the requested streamlet 304. In a further embodiment, **the client module 114 may be configured to retrieve streamlets**

304 from a plurality of web servers 116.”) (emphasis added). A POSITA would have recognized the web servers taught by Hurst as content distribution servers because the web servers host content and distribute the content to requesting users. *See, e.g., id.* 6:23-25. A POSITA would have recognized both references to be teaching well-known content-delivery architectures on the Internet, going back decades, and been motivated to apply these teachings. Thus, the combination teaches transmitting a request for content to a set of one or more content distribution servers and receiving the content from the set of one or more content distribution servers.

10. Claim 9

**9 The playback device of claim 8, wherein the application is further for causing the set of processors to perform the steps of:
transmitting a request for cryptographic information including the set of common keys to a set of one or more DRM servers; and
receiving the cryptographic information from the set of one or more DRM servers.**

213. The combination teaches or suggests transmitting a request for cryptographic information including the set of common keys to a set of one or more DRM servers and receiving the cryptographic information from the set of one or more DRM servers. *See, e.g., Ex. 1007, ¶¶*[0017, 0039, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084]. Hurst renders this limitation obvious by teaching a DRM server authenticating a user and supplying the user with appropriate encryption keys. *See, e.g., Ex. 1008, 18:51-55* (“The

DRM server 1204 is configured to provide digital rights management capability to the system 1200. The DRM server 1204 is further configured to supply encryption keys to the end user upon authenticating the end user.”), 19:5-8 (“The DRM server 1204 is configured to authenticate a client device and accordingly communicate the appropriate decryption key.”). It would have been obvious to a POSITA for Hurst’s DRM server to authenticate a user in response to a request for access to protected content (including cryptographic information needed for playback, e.g., keys, algorithms, and related parameters), which would entail the client device transmitting a request for cryptographic information, including the set of common keys that will be used to decrypt the protected content, to a set of one or more DRM servers. *See supra* Section VIII.A.2[1m]. After successful authentication, it would have been obvious for the DRM server to send, and for the client device to receive, the cryptographic information that will be used to decrypt the protected content. *See id.* Thus, the combination renders this limitation obvious.

214. Additionally, Lindahl teaches a media distribution system that gives authorized users, such as users who purchased a media file, appropriate cryptographic information to decrypt the media file. *See, e.g.*, Ex. 1007, ¶¶[0039] (“The media distribution system 100 includes a media server 102. The media server 102 stores, manages... and can also prepare the media files for download to prospective users (i.e., purchasers)”), [0044] (“since these media files are partially

encrypted, they need to be processed (i.e., decrypted) to remove the encryption before they are usable. Authorized users or devices are given the appropriate cryptographic information (e.g., keys and encryption parameters) to remove the partial encryption.”). Here, Lindahl is teaching that the media distribution system is performing widely-known functions of a typical DRM server. *See id.*

215. Lindahl teaches transmitting a request, when the user purchases a media file, for cryptographic information including the set of common keys to a set of one or more DRM servers, and then receiving that cryptographic information in response. *See* Ex. 1007, ¶¶[0062-0066] (“[A] user selects 704 a media file for purchase. Then, the user purchases 706 the media file and receives an associated cryptographic key... Next, the media file is downloaded to the user... by streaming the media file through the data network to the user... Once the media file is downloaded 708, the media file can then be decrypted... using the cryptographic key... which has previously been provided to the user who has purchased the access rights to the media file.”). A POSITA would have found it obvious for the user’s device to transmit a request, given that Lindahl teaches the user purchasing the media file from the server, and the server sending cryptographic keys and other DRM information in response. It would have been obvious that the appropriate cryptographic information would include the keys required to decrypt the content, including the set of common keys. *See supra* Section VIII.A.2.[1e, 1g].

11. Claim 10

10	The playback device of claim 1, wherein the application is further for causing the set of processors to perform the step of obtaining the container index from at least one file selected from the group consisting of: the associated container file containing the selected stream of protected content; and a separate file to the associated container file containing the selected stream of protected content.
-----------	---

216. The combination teaches or suggests the playback device of claim 1 and further teaches or suggests obtaining the container index from at least one file selected from the group consisting of: the associated container file containing the selected stream of protected content; and a separate file to the associated container file containing the selected stream of protected content. *See, e.g.*, Ex. 1006, ¶¶[0063-0064, 0066, 0070-0074, 0111, 0119-0123, 0129, 0147-0150], Ex. 1007, ¶¶[0010, 0033, 0055-0061, 0066, 0070, 0072]. The combination teaches both options of the recited group. First, as described above, the combination teaches obtaining a container file containing a selected stream of protected content. *See supra* Section VIII.A.2.[1j], VIII.A.3[2a]. Chen teaches that the container index (e.g., segment index) is typically placed at the beginning of the segment file. *See, e.g.*, Ex. 1006, ¶¶[0123] (“This segment indexing may be included within the segment, typically at the beginning of the segment.”), [0120] (“The “moov” box may be placed at the start of the file, followed by fragment offset data, also referred to as a segment index, containing offset information in time and byte ranges for each fragment or at least a subset of fragments in the containing

segment.”); *supra* Section VIII.A.2.[1j]. Thus Chen teaches the step of obtaining the container index from the associated container file containing the selected stream of protected content. Further, Chen teaches providing the segment index in a separate file. *See, e.g.*, Ex. 1006, ¶[0123] (“The segment index may also be provided in a separate index segment or file.”). Thus Chen teaches obtaining the container index from a separate file to the associated container file containing the selected stream of protected content. It would have been obvious to a POSITA, based on Chen’s teachings, to place a container index in an associated container file or in a separate file. Both techniques were widely-known in the art, and a POSITA would have been motivated to apply them depending on the situation. A POSITA would have been motivated to place the container index in the associated container file in order to, for example, facilitate quickly obtaining the container index when the video stored in the associated container file is requested. *See, e.g.*, Ex. 1006, ¶[0123]. A POSITA would have been motivated to place the container index in a separate file in order to, for example, provide an overall index of multiple container files to facilitate switching between streams. Thus, the combination teaches the playback device of claim 1 performing the additional limitations of claim 10.

12. Claim 11

11	The playback device of claim 1, wherein the common set of keys comprises a plurality of keys.
-----------	--

217. The combination teaches or suggests that the common set of keys comprises a plurality of keys (e.g., more than one key). *See, e.g.*, Ex. 1007, ¶¶[0017, 0044, 0046, 0048-0049, 0054, 0062-0067, 0069-0072, 0077-0080, 0082-0084], FIGS. 5A-5C, 6A-6B. For example, Lindahl teaches using more than one key to encrypt and decrypt the blocks in a video file. *See, e.g.*, Ex. 1007, ¶¶[0054] (“The encryption of each block in accordance with the encryption parameters is discussed in greater detail below with respect to FIGS. 5A-5C and FIGS. 6A-6B. ... the encryption process also makes use of one or more encryption keys when encrypting each block.”), [0072] (“[T]he decryption of each block also makes use of one or more cryptographic keys...”). For example, the referenced Figures 6A and 6B illustrate two embodiments for encrypting blocks, where Figure 6A has one encrypted portion of the block, and Figure 6B has two encrypted portions of the block. *See* Ex. 1007, FIGS. 6A-6B:

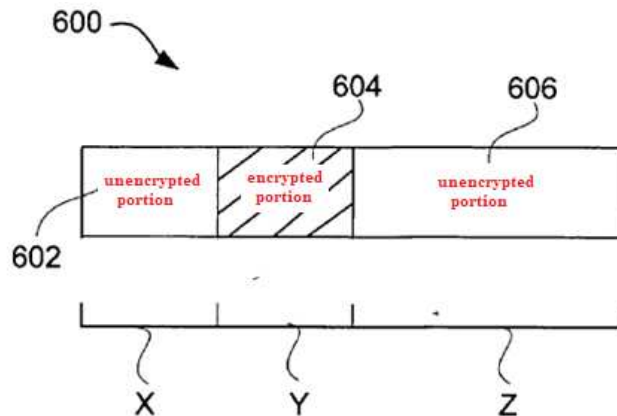


FIG. 6A
(ANNOTATIONS ADDED)

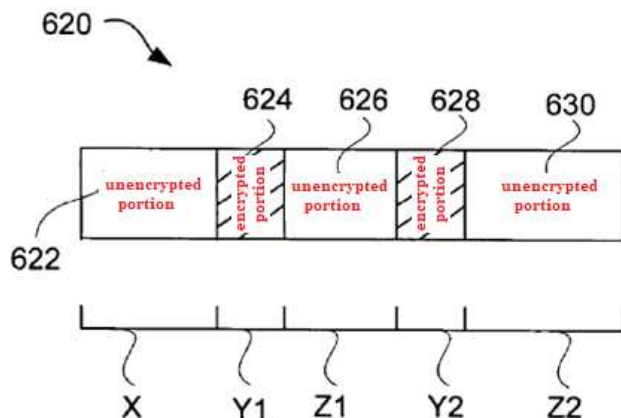


FIG. 6B
(ANNOTATIONS ADDED)

218. A POSITA would have found it obvious for the different encrypted portions of FIG. 6B to be encrypted using different cryptographic keys because FIGS. 6A-6B teach one or more encrypted portions for each block, and Lindahl teaches that encryption as discussed for FIGS. 6A-6B “makes use of one or more

encryption keys when encrypting each block.” *See id.*, ¶¶[0054, 0072], FIGS. 6A-6B. Lindahl teaches that a video is comprised of blocks, therefore Lindahl teaches using more than one encryption key to encrypt a video. *See id.*, ¶¶[0054, 0057-0059]. A POSITA would have been motivated to apply these teachings from Lindahl because using more than one encryption key for a video increases the security of the encryption. Each key is used to encrypt a smaller amount of data, making it more difficult to access the media. And even if one key is compromised, it only decrypts a part of the video stream, leaving portions of each block of each from of video encrypted. It was common in the art to use multiple keys in this manner, going back years if not decades, with the keys stored and organized in key tables. *See* Section I.A.

219. Hurst teaches or suggests that the alternative streams of content, such as video files, may be encrypted in the same manner. *See, e.g.*, Ex. 1008, 18:62-67. For example, if a stream is encrypted with one key, Hurst teaches encrypting each of the alternative streams with the same key. *See id.*; *see supra* Section VIII.B.2.[1e]. Lindahl teaches encrypting a video using a plurality of keys, which in the context of this combination would have suggested encrypting a video stream using a plurality of keys. When combined with Lindahl, Hurst’s teachings would have suggested to a POSITA to encrypt each of the alternative streams in the same manner, with that same plurality of keys. *See, e.g.*, Ex. 1007, ¶¶[0054], [0072].

This concept is illustrated below, with an example of at least 3 keys (K1, K2, K3) being used to encrypt each low bitrate stream (row 204); given Hurst's teachings, it would have been obvious to use that same set of keys to encrypt the medium (row 206) and high bitrate (row 208) streams for the same content. See Ex. 1008, Fig. 3b:

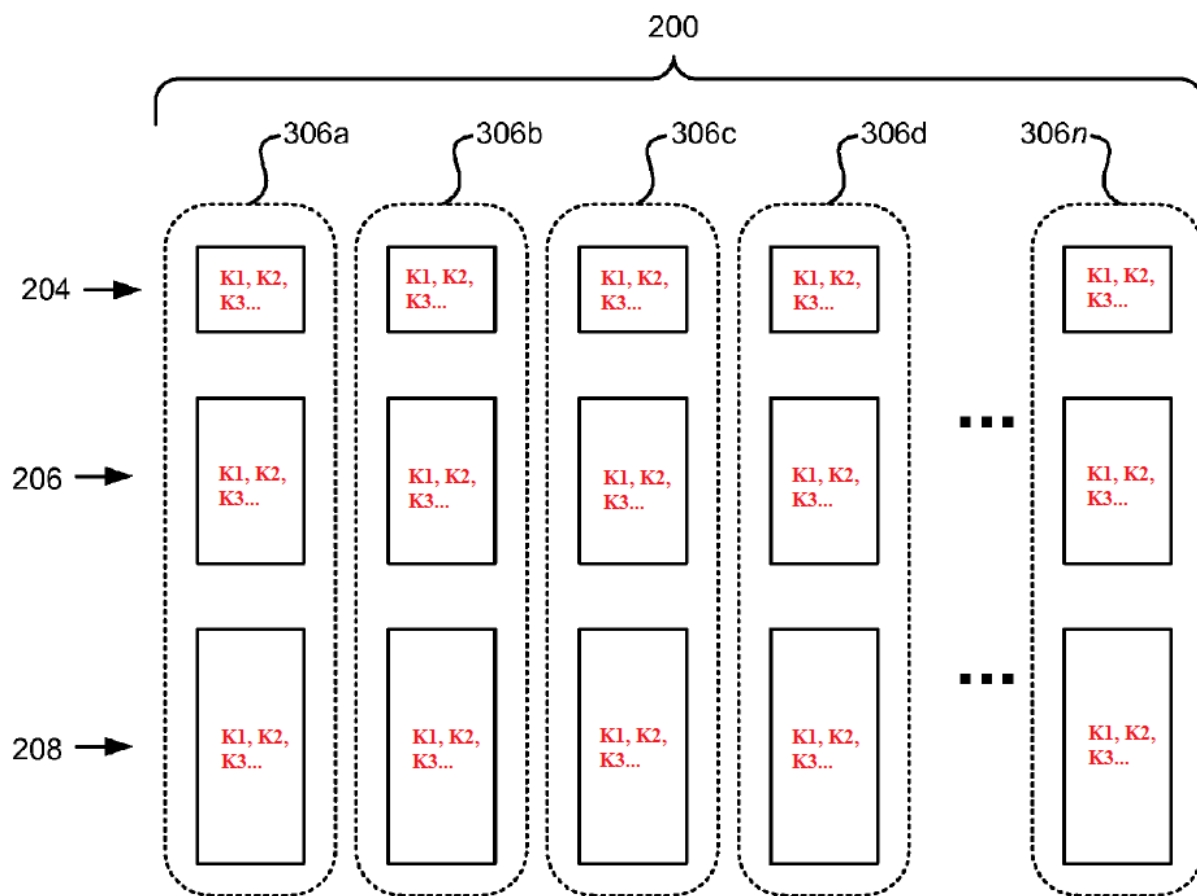


FIG. 3b
(ANNOTATIONS ADDED)

A POSITA would have been motivated to do this because using a plurality of keys, as taught by Lindahl, increases security, and applying the same set of keys to each

alternative stream, as taught by Hurst, aligns with the usage model for adaptive streaming where access is typically granted to the alternative streams as a group. *See supra* Section VIII.A.1. Furthermore, encrypting the alternative streams in the same manner is one of a finite number of possibilities that would have at least been obvious to try. *See supra* Section VIII.A.2[1e]. Thus, given the combined teachings of the references, it would have been obvious for the partially encrypted video frames to be encrypted using a set of common keys comprising a plurality of keys.

13. Claim 12

12	A method for playing protected content from a plurality of alternative streams on a playback device, the method comprising: obtaining a top level index file identifying a plurality of alternative streams of protected video, wherein each of the alternative streams of protected video includes partially encrypted video frames that are encrypted using a set of common keys comprising at least one key, and wherein the partially encrypted video frames contain encrypted portions and unencrypted portions of data; obtaining a copy of the set of common keys; detecting streaming conditions for the playback device; selecting a stream from the plurality of alternative streams of protected video based on the detected streaming conditions; receiving a container index that provides byte ranges for portions of the selected stream of protected video within an associated container file; requesting portions of the selected stream of protected video based on the provided byte ranges; locating encryption information that identifies encrypted portions of frames of video within the requested portions of the selected stream of protected video; decrypting each encrypted portion of the frames of video identified within the located encryption information using the set of common keys; and playing back the decrypted frames of video obtained from the requested portions of the selected stream of protected video using a decoder.
----	---

220. Claim 12 recites the same limitations as claim 1 except for “using a decoder” and is obvious for the same reasons explained above. *See supra* Section VIII.A.2.1[a]-[1n]. Given the obviousness of an apparatus performing the steps of claim 12, explained above for claim 1, it would have been obvious to perform these steps as a method.

221. The combination further teaches or suggests playing back decrypted frames of video using a decoder. Chen and Lindahl both teach using decoders in the claimed manner. *See, e.g.*, Ex. 1006, ¶[0106] (“An example of a media decoder would be a function that transforms data in the format described in the H.264 video coding standard into analogue or digital representations of video frames, such as a YUV-format pixel map with associated presentation timestamps for each frame or sample.”); Ex. 1007, ¶[0091] (“the particular media item is first decrypted by the processor 1102 using processing discussed above and then supplied to a coder/decoder (CODEC) 1112. The CODEC 1112 then produces analog output signals for a speaker 1114 and/or the display 1110.”). Chen teaches a decoder that transforms data in a video coding standard into analog or digital representations of video frames. Lindahl teaches a decoder that produces output signals based on a decrypted media item. Therefore Chen and Lindahl render this limitation obvious.

222. Hurst further renders this obvious by teaching encoding of a plurality of streams. *See, e.g.*, Ex. 1008, 8:19-25 (“The encoding module 406 may utilize encoding schemes such as DivX®, Windows Media Video 9®, Quicktime 6.5 Sorenson 3®, or Quicktime 6.5/MPEG-4®.”), 16:3-5 (“In a further embodiment, the encoding scheme comprises a proprietary codec such as WMV9®.”). It would have been obvious to a POSITA that playing back an encoded stream requires

decoding the encoded stream. Hurst teaches a media player for playing music and videos. *See, e.g.*, 1008, 1:15-19; *supra* Section VIII.B.2.1[a].

223. It would have been obvious to a POSITA, based on these teachings, to use a decoder to play back decrypted frames of video, because this is one of the known (if not principal) functions of decoders. Thus, the combination teaches playing back decrypted frames of video using a decoder.

14. Claim 13

13	The method of claim 12, wherein requesting portions of the selected stream further comprises obtaining a container file containing protected video from at least one of the plurality of alternative streams, where the container file also includes encryption information that identifies portions of frames of video that are encrypted and a reference to at least one key from the set of common keys to utilize in accessing the encrypted portions of the frames of video.
-----------	--

224. Claim 13 recites the same limitations as claim 2 and is obvious for the same reasons explained above. *See supra* Section VIII.A.3.2[a]-[2c]. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

15. Claim 14

14	The method of claim 12, wherein the located encryption information comprises a reference to the start of an encrypted block of data.
-----------	---

225. Claim 14 recites the same limitations as claim 3 and is obvious for the same reasons explained above. *See supra* Section VIII.A.4. Given the

obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

16. Claim 15

15	The method of claim 14, wherein the located encryption information further comprises the size of the encrypted block of data.
-----------	--

226. Claim 15 recites the same limitations as claim 4 and is obvious for the same reasons explained above. *See supra* Section VIII.A.5. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

17. Claim 16

16	The method of claim 14, wherein the located encryption information further comprises cryptographic information that can be utilized to access the encrypted portion of the frame.
-----------	--

227. Claim 16 recites the same limitations as claim 5 and is obvious for the same reasons explained above. *See supra* Section VIII.A.6. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

18. Claim 17

17	The method of claim 16, wherein the cryptographic information is a reference to at least one key from the set of common keys.
-----------	--

228. Claim 17 recites the same limitations as claim 6 and is obvious for the same reasons explained above. *See supra* Section VIII.A.7. Given the

obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

19. Claim 18

18. The method of claim 12 further comprising: detecting a change in the streaming conditions; identifying a second alternative stream of protected video based on the detected change; receiving a container index that provides byte ranges for portions of the second alternative stream of protected video within a second associated container file; requesting portions of the second alternative stream of protected video based on the provided byte ranges; decrypting each encrypted portion of the frames of video from the requested portions of the second alternative stream of protected video using the set of common keys; and playing back the decrypted frames of video from the requested portions of the second alternative stream of protected video.

229. Claim 18 recites the same limitations as claim 7 and is obvious for the same reasons explained above. *See supra* Section VIII.A.8. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

20. Claim 19

19	The method of claim 12 further comprising: transmitting a request for content to a set of one or more content distribution servers; and receiving the content from the set of one or more content distribution servers.
-----------	--

230. Claim 19 recites the same limitations as claim 8 and is obvious for the same reasons explained above. *See supra* Section VIII.A.9. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

21. Claim 20

20	The method of claim 19 further comprising: transmitting a request for cryptographic information including the set of common keys to a set of one or more DRM servers; and receiving the cryptographic information from the set of one or more DRM servers.
-----------	---

231. Claim 20 recites the same limitations as claim 9 and is obvious for the same reasons explained above. *See supra* Section VIII.A.10. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

22. Claim 21

21	The method of claim 12 further comprising obtaining the container index from at least one file selected from the group consisting of: the associated container file containing the selected stream of protected content; and a separate file to the associated container file containing the selected stream of protected content.
----	---

232. Claim 21 recites the same limitations as claim 10 and is obvious for the same reasons explained above. *See supra* Section VIII.A.11. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

23. Claim 22

22	The method of claim 12, wherein the common set of keys comprises a plurality of keys.
----	--

233. Claim 22 recites the same limitations as claim 11 and is obvious for the same reasons explained above. *See supra* Section VIII.A.12. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

24. Claim 23

23	The playback device of claim 1, wherein: the container index is part of a hierarchical index; and the application is further for causing the set of processors to perform the step of: obtaining a lower layer index that identifies the location of frames within a specific requested portion of the selected stream of protected video; and identify partially encrypted video frames from within the specific requested portion of the selected stream using the lower layer index.
----	---

234. The combination teaches or suggests the playback device of claim 1 and further teaches or suggests that the container index is part of a hierarchical index, and the step of obtaining a lower layer index that identifies the location of frames within a specific requested portion (e.g., fragment) of the selected stream of protected video (e.g., the selected representation). *See, e.g.*, Ex. 1006, ¶¶[0033, 0063, 0121, 0148, 0164, 0190-202, 0216-0217, 0219-0267], FIG. 5, 7(b), Ex. 1007, ¶¶[0010, 0014-0016, 0033, 0052, 0055-0061, 0072], FIG. 5A-5C, 6A-6B. First, Chen teaches hierarchical indexing. *See, e.g.*, Ex. 1006, ¶¶[0148] (“The indexing can be simple, wherein the time and byte ranges of each block are absolute relative to the start of the file, or **they can be hierarchical**, wherein some blocks are grouped into parent blocks (and those into grandparent blocks, etc.) and the time and byte range for a given block is expressed relative to the time and/or byte range of the block's parent block.”) (emphasis added), [0164] (“A Time Index Box with targettype of Time Index Box can be used to construct **a hierarchical**

time index, allowing users of the file to quickly navigate to the required portion of the index.”) (emphasis added), FIG. 7(b):

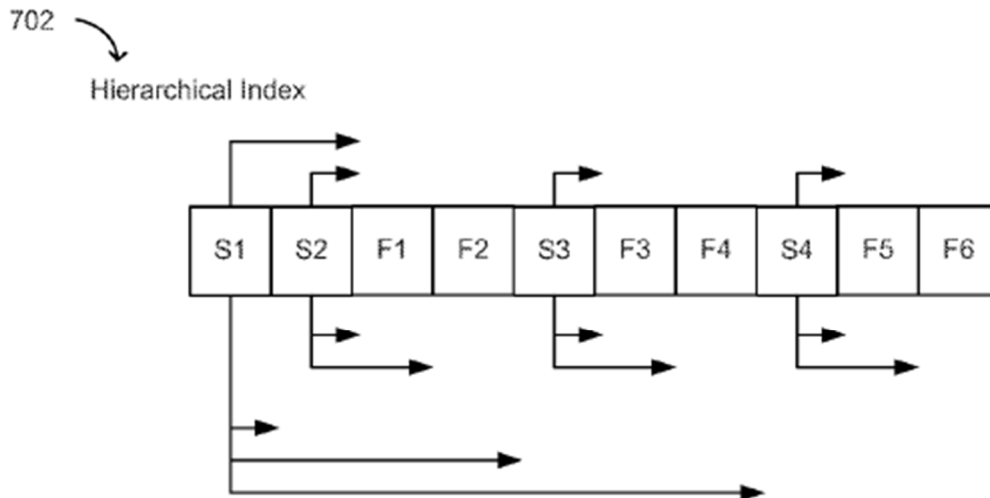


FIG. 7(b)

Thus Chen teaches, and it would have been obvious, for Chen’s container index to be part of a hierarchical index.

235. Chen further teaches a “seek point structure preferably is used to determine the block structure. ... The blocks for a given version of the content do not overlap and every frame of the content is contained within exactly one block of each version.” Ex. 1006, ¶[0197]. Thus a POSITA been motivated and found it obvious for Chen’s hierarchal index to identify the location of frames within blocks because (a) it allows playback devices to seek to a specific point in a video, such as the start of a scene or an arbitrary point in the video that the user wants to access, and (b) it allows client devices to switch between alternative streams at any

frame within the video, providing greater flexibility for adaptive streaming and allowing faster adjustments for changes to streaming conditions. *See id.*

Furthermore, it would have been obvious for Chen's hierarchal index to identify the location of frames within blocks to facilitate decrypting and decoding of those frames, whose boundaries and locations typically needed to be identified, as Lindahl teaches. *See* Ex. 1007, ¶¶[0014-0016] ("As a method for decrypting media data, one embodiment of the invention includes at least the acts of... examining the media data in the media file to locate frame boundaries for the frames within the media data."), [0052], [0072]. It would have been obvious for a lower layer index to contain this identification information for the location of frames because within the hierarchal index, the lower layer is closer to the frames. *See* Ex. 1006, ¶[0148]. Thus, given Chen's teachings, it would have been obvious for the client device, when attempting to access a specific part of the video or switch between alternative streams, to obtain a lower layer index that identifies the location of frames within a specific requested portion of the selected stream of protected video—and then use that index to identify partially encrypted video frames from within the specific requested portion of the selected stream to access that frame on the client device.

236. As explained above, the combination teaches requesting portions of the selected stream of protected video. *See supra* Section VIII.A.2[1k]. Thus, a POSITA would have found it obvious, based on the combined teachings of Chen

and Lindahl, to have a container index that is part of a hierarchical index, and to obtain a lower layer index that identifies the location of frames within a specific requested portion of the selected stream of protected video.

237. As described above, the combination teaches encryption parameters that identifies the encrypted portions of data. *See supra* Section VIII.A.2.[11], VIII.A.2.[2b]. The combination also teaches that each of the alternative streams of protected video includes partially encrypted video frames. *See supra* Section VIII.A.2.[1d]. The lower layer index identifies the location of frames within the specific requested portion of the selected stream, and thus, it identifies partially encrypted video frames from within the specific requested portion of the selected stream.

25. Claim 24

24	The method of claim 12, wherein: the container index is part of a hierarchical index; and the method further comprises: obtaining a lower layer index that identifies the location of frames within a specific requested portion of the selected stream of protected video; and identify partially encrypted video frames from within the specific requested portion of the selected stream using the lower layer index.
----	---

238. Claim 24 recites the same limitations as claim 23 and is obvious for the same reasons explained above. *See supra* Section VIII.A.24. Given the obviousness of an apparatus performing these steps, explained above, it would have been obvious to perform these steps as a method.

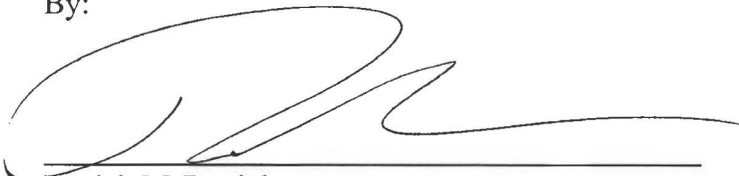
IX. RESERVATION OF RIGHTS

239. My opinions are based upon the information that I have considered to date. I am unaware of any evidence of secondary considerations with respect to the '588 Patent that would render any of the challenged claims non-obvious. I reserve the right, however, to supplement my opinions in the future to respond to any arguments raised by the owner of the '588 Patent and to take into account new information that becomes available to me.

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

Executed on February 14, 2019.

By:



Patrick McDaniel

APPENDIX 1: CURRICULUM VITAE OF PATRICK MCDANIEL

PATRICK DREW MCDANIEL

William L. Weiss Professor of Information and Communications Technology
School of Electrical Engineering and Computer Science ◊ Pennsylvania State University
Office : W329 Westgate Building ◊ University Park, PA 16802 ◊ (814) 863-3599
email: mcdaniel@cse.psu.edu ◊ Homepage: <http://www.patrickmcdaniel.org/>

ACADEMIC AND RESEARCH APPOINTMENTS

William L. Weiss Professor of Information and Communications Technology Computer Science and Engineering, Pennsylvania State University, University Park, PA	2017-present
Distinguished Professor Computer Science and Engineering, Pennsylvania State University, University Park, PA	2016-2017
Professor Computer Science and Engineering, Pennsylvania State University, University Park, PA	2011-2015
Associate Professor Computer Science and Engineering, Pennsylvania State University, University Park, PA	2007-2011
Hartz Family Career Development Assistant Professor Computer Science and Engineering, Pennsylvania State University, University Park, PA	2004-2007
Adjunct Professor Stern School of Business, New York University, New York, NY	2003-2009
Senior Research Staff Member AT&T Labs - Research, Florham Park, NJ	2001-2004

RESEARCH LEADERSHIP APPOINTMENTS

Director, Institute for Networking and Security Research College of Engineering, Pennsylvania State University, University Park, Pennsylvania	2016-present
Director, National Science Foundation, Center for Trustworthy Machine Learning Participants: Penn State, Stanford, UC Berkeley, UC San Diego, Univ. of Wisconsin, Univ. of Virginia	2018-present
Program Manager, Cyber-Security Collaborative Research Alliance (CRA) College of Engineering, Pennsylvania State University, University Park, Pennsylvania	2013-2018
Co-Director, Systems and Internet Infrastructure Laboratory College of Engineering, Pennsylvania State University, University Park, Pennsylvania	2004-present

EDUCATION

University of Michigan , Ann Arbor, MI • Ph.D. , Computer Science and Engineering • Dissertation: <i>Policy Management in Secure Group Communication</i> • Advisor: Dr. Atul Prakash	2001
Ball State University , Muncie, IN • M.S. , Computer Science • Thesis: The Analysis of D_i , a Detailed Design Metric on Large Scale Software	1991
Ohio University , Athens, OH • B.S. , Computer Science	1989

STUDENTS

Past Post-Docs

- *Vaibhav Rastogi*, graduated Northwestern University, joined
- *Robert Walls*, graduated University of Massachusetts, Amherst, joined

Past PhD Students

- *Z. Berkay Celik*, Spring 2019, now **Assistant Professor**, Purdue University
- *Nicolas Papernot*, Spring 2018, now **Assistant Professor**, University of Toronto
- *Wenhui Hu*, Fall 2016, now **Senior Member of Technical Staff**, Oracle
- *Devin Pohly*, Spring 2016, now **Assistant Professor**, Wheaton College
- *Damien Octeau*, Summer 2014, now **Software Engineer in Security**, Google
- *Steve McLaughlin*, Spring 2014, now **Senior Software Engineer**, Samsung Research America
- *Thomas Moyer*, Summer 2011, now **Assistant Professor**, University of North Carolina-Charlotte
- *William Enck*, Spring 2011, now **Associate Professor**, North Carolina State University
- *Kevin Butler*, Summer 2010, now **Associate Professor**, University of Florida
- *Machigar Ongtang*, Summer 2010, now **Assistant Professor**, Dhurakij Pundit University
- *Patrick Traynor*, Spring 2008, now **Professor**, University of Florida, co-advisor
- *Fr. Boniface Hicks*, Fall 2007, now **Assistant Professor**, St. Vincent College

Current PhD Students : *Ryan Sheatsley*, Pennsylvania State University, Spring 2022 ◊ *Bolor-Erdene Zolbayar*, Pennsylvania State University, Spring 2022

Past Masters Students : *Sushrut Shringarputale*, Pennsylvania State University, Fall 2019 ◊ *Raquel Alvarez*, Pennsylvania State University, Spring 2019 ◊ *Valentin Vie*, Pennsylvania State University, Spring 2019 ◊ *Ryan Sheatsley*, Pennsylvania State University, Fall 2018 ◊ *Eric Kilmer*, Pennsylvania State University, Spring 2016 ◊ *Nathan Lagerman*, Pennsylvania State University, Spring 2016 ◊ *Matthew Dering*, Pennsylvania State University, Spring 2014 ◊ *Phil Koshy*, M.S. Pennsylvania State University, Fall 2013 ◊ *Diana Koshy*, M.S. Pennsylvania State University, Fall 2013 ◊ *Steve McLaughlin*, M.S. Pennsylvania State University, Spring 2011 ◊ *Sergei Miadzvezhanka*, M.S. Pennsylvania State University, Spring 2011 ◊ *Adam Delozier*, M.S. Pennsylvania State University, Spring 2011 ◊ *Juliet Uhloft*, M.Eng. Pennsylvania State University, Fall 2010 ◊ *Damien Octeau*, M.S. Pennsylvania State University, Spring 2010 ◊ *Thomas Moyer*, M.S. Pennsylvania State University, Spring 2009 ◊ *Luke St. Clair*, M.S. Pennsylvania State University, Summer 2008 ◊ *Lisa Johansen*, M.S. Pennsylvania State University, Spring 2008 ◊ *Sunam Ryu*, M.S. Pennsylvania State University, Spring 2007 ◊ *Dhananjay Bapat*, M.S. Pennsylvania State University (Electrical Engineering), Fall 2006 ◊ *Jennifer Plasterr*, M.Eng. Pennsylvania State University, Summer 2006 ◊ *Adam Kerr*, M.Eng. Pennsylvania State University, Fall 2006 ◊ *William Enck*, M.S. Pennsylvania State University, Spring 2006 ◊ *Wesam Lootah*, M.S. Pennsylvania State University, Spring 2006 ◊ *Jon Hansford*, M.Eng. Pennsylvania State University, Fall 2005 ◊ *John van Bremer*, M.Eng. Pennsylvania State University, Spring 2005

TEACHING

- **CMPSC311 - Introduction to Systems Programming**
Fall 2013, Fall 2014, Fall 2015, Fall 2016, Summer 2019
- **CMPSC443 - Introduction to Computer and Network Security**
Spring 2006, Spring 2009, Fall 2017, Fall 2018
- **CSE543 - Computer and Network Security**
Fall 2004, Fall 2005, Fall 2008, Fall 2009, Fall 2011, Fall 2014
- **CSE544 - Advanced System Security**
Spring 2005, Spring 2007
- **CSE545 - Advanced Network Security**
Spring 2006, Spring 2008, Spring 2011
- **CSE597g - Principles, Analysis, and Applications of Computer Security**
Fall 2015
- **Security and Privacy of Machine Learning**
Fall 2016

-
- **Advanced Topics in the Security and Privacy of Machine Learning**
Spring 2017
 - **CSE598 - Cell Phone Operating Systems**
Spring 2009
 - **CSE598i - Web 2.0 Security**
Spring 2010
 - **CSE598d - Topics in Applied Systems Security**
Fall 2010
 - **CSE598e - Critical Infrastructure Security**
Fall 2011

INDUSTRIAL EXPERIENCE

Software Developer Applied Innovation, Inc., Columbus, OH	1994-1995
Project Manager Primary Access Corporation, San Diego, CA	1993-1994
Software Developer Primary Access Corporation, San Diego, CA	1991-1993
Software Developer Integrated Technologies, Inc., Muncie, IN	1989

AFFILIATIONS

Association for Computing Machinery (ACM), *Fellow*
The Institute of Electrical and Electronics Engineers (IEEE), *Fellow*
USENIX Advanced Computing Systems Association (USENIX)
American Association for the Advancement of Science (AAAS)

HONORS, AWARDS, AND KEYNOTE ADDRESSES

Penn State Engineering Society Outstanding Advising Award, *highly selective award by the Penn State Engineering Society given to faculty in the College of Engineering who have made significant contributions as advisor, October 2018*

Best Paper, 2017 EAI SECURECOMM 2018, *with Sayed M. Saghaian, Tom La Porta, Trent Jaeger and Z. Berkay Celik, August 2018*

Best Student Paper, 2017 ACM Symposium on SDN Research (SOSR), *with Stefan Achleitner, Thomas La Porta and Trent Jaeger, April 2017*

IEEE Technical Committee on Security and Privacy Outstanding Community Service Award, *in recognition for leadership of the Technical Committee on Security and Privacy, May 2016*

ACM Fellow, *for contributions to computer and mobile systems security, December 2015*

Science of Security Index of Significant Research in Cyber Security, *acknowledging paper 'Toward a Science of Secure Environments', Science of Security Virtual Organization (SOS-VO), August 2015*

IEEE Fellow, *for contributions to the security of mobile communications, November 2014*

Best Artifact Award, 20th International Symposium on the Foundations of Software Engineering (FSE), *with advisee Damien Ocateau and collaborator Somesh Jha, November 2012*

Best Paper, 25th Annual Computer Security Applications Conference, *with advisees Machigar Ongtang, Stephen McLaughlin, and William Enck, December 2009*

Faculty Marshal, College of Engineering, *selected by student marshals for contributions to undergraduate education, leads procession into graduation ceremony, May 2009*

Penn State Engineering Society Outstanding Research Award, *highly selective award by the Penn State Engineering Society given to faculty in the College of Engineering who have made significant contributions to knowledge in their field, March 2009*

Google Security and Product Safety Acknowledgement, *in recognition of efforts in improving the security of Google Android cellular phone operating system, 2008*

Commendation for Exceptional Leadership and Achievement, *in recognition of efforts as PI of the EVEREST study, from Ohio Secretary of State Jennifer Brunner, August 2008*

IEEE Technical Committee on Security and Privacy Outstanding Community Service Award, *in recognition for technical program management of the IEEE Security and Privacy symposia, August 2008*

National Science Foundation CAREER Award, *Faculty early career development grant, August 2007*

Penn State Computer Science and Engineering Outstanding Teaching Award, *Given to best teacher in the department as selected by students, March 2007*

ACM Certificate of Meritorious Service, *Certificate acknowledging exemplary service as associate editor of ACM Transactions on Internet Technologies, April 2007*

Best Student Paper, 22nd Annual Computer Security Applications Conference, *as advisor, with Boniface Hicks and Kiyan Ahmadizadeh, December 2006*

Best Paper, Innovations and Commercial Applications of Distributed Sensor Networks Symposia, *Awarded for best paper in conference, October 2005.*

Hartz Family Career Development Professor, *Endowed Professorship, Pennsylvania State University, Fall 2004-2007*

Bang for the Buck Award, DARPA Dynamic Coalitions Program, *Award for most feature-rich/useful software system, April 2002*

National Aeronautics and Space Administration, Kennedy Space Center Fellowship, *Research Fellowship, September 1997 - August 2000*

Electrical Engineering and Computer Science Summer Fellowship Award, *University of Michigan, June 1997*

Dean's Citation for Perfect Academic Record, *Ball State University, June 1991*

Keynote Addresses

1. The Challenges of Machine Learning in Adversarial Settings. Triangle Area Privacy and Security Day, Durham, NC, October, 2019.
2. The Challenges of Machine Learning in Adversarial Settings. 2019 Subversion and Assurance of AI Workshop, US National Reconnaissance Office, Washington, DC, March, 2019.
3. Attacks, Defenses, and Impacts of Machine Learning in Adversarial Settings. 2017 Conference on Security and Privacy in Communication Networks (SecureComm), Niagara Falls, Canada, October, 2017.
4. Tracing the Arc of Smartphone Application Security. 2017 ACM on International Workshop on Security And Privacy Analytics. Scottsdale, AZ, March, 2017.
5. Tracing the Arc of Smartphone Application Security. 12th International Conference on Information Systems Security, Jaipur, India, December, 2016.
6. The 25th International Conference on Computer Communication and Networks (ICCCN 2016), August, 2016, Waikoloa, Hawaii.
7. Learning from Ourselves: Where are we and where can we go in mobile systems security?. Mobile Security Technologies (MOST) 2016 Workshop, IEEE Computer Society Security and Privacy Workshops, San Jose, CA, May, 2016.

-
8. Eight Years of Mobile Smartphone Security. Center for Secure and Dependable Systems (CSDS) Cybersecurity Symposium, Coeur d'Alene, April, 2016.
 9. The Importance of Measurement and Decision Making to a Science of Security, 2015 IEEE Conference on Communications and Network Security (CNS), September 2015, Florence, Italy.
 10. The Importance of Measurement and Decision Making to a Science of Security. 3rd International Symposium on Resilient Cyber Systems, Philadelphia, PA, August, 2015.
 11. The Importance of Measurement and Decision Making to a Science of Security, 2015 Symposium And Bootcamp on the Science of Security (Hotsos), April 2015, University of Illinois at Urbana-Champaign
 12. Security and Science of Agility, ACM Workshop on Moving Target Defense (MTD 2014), November 2014, Scottsdale, AZ
 13. A Secondary Internet Revolution: How the Smart Device has Changed the Information Security Landscape, IEEE New Technology Industry Seminar (NTIS '13), Everett WA, August, 2013
 14. Permission-based Application Governance; A Step Forward or Backward?, 26th Annual WG 11.3 Conference on Data and Applications Security and Privacy (DBSec'12), Paris, France, July 2012.
 15. Scalable Integrity-Guaranteed AJAX, The 14th Asia-Pacific Web Conference (APWeb), Kunming, China, April 2012.
 16. Security Challenges and Solutions in Mobile Smartphone Applications, IEEE Computer Security Foundations (CSF 2011), Abbaye des Vaux de Cernay, France, June 2012.
 17. Password Exhaustion: Predicting the End of Password Usefulness. 2nd International Conference on Information Systems Security, Kolkata India, December, 2006.
 18. Physical and Digital Convergence: Where the Internet is the Enemy. Eighth International Conference on Information and Communications Security (ICICS '06), Raleigh, NC, December, 2006.

Distinguished Lectures

1. The Challenges of Machine Learning in Adversarial Settings: A Systems Perspective. Computer Science Department, University of Wisconsin-Madison, Madison, WI, February, 2020.
2. Shutterstock Distinguished Lecture: The Challenges of Machine Learning in Adversarial Settings. Computer Science Department, Stonybrook University, Stonybrook, NY, December, 2019.
3. Distinguished Blockchain Lecture: The Challenges of Machine Learning in Adversarial Settings. Cylab Security and Privacy Institute, Carnegie Mellon University, Pittsburgh, PA, December, 2019.
4. Distinguished Speaker Series: The Challenges of Machine Learning in Adversarial Settings. Department of Computer Science, University at Buffalo, Buffalo, NY, November, 2018.
5. Samuel D. Conte Distinguished Lecture Series: The Challenges of Machine Learning in Adversarial Settings. Department of Computer Science, Purdue University, West Lafayette, Indiana, November, 2018.
6. The Challenges of Machine Learning in Adversarial Settings. Department of Software and Information Systems, University of North Carolina at Charlotte, Charlotte, NC, February, 2018.
7. Tracing the Arc of Smartphone Application Security. Celebrating 50 Years of Computer Science @ NC State, North Carolina State University, Raleigh, NC, October, 2017.
8. Tracing the Arc of Smartphone Application Security. Computer Science Department and the Electrical and Computer Engineering Department Seminar Series, Colorado State University, Fort Collins, CO, October, 2017.
9. Tracing the Arc of Smartphone Application Security. Rochester Institute of Technology, College of Computing and Information Sciences, Rochester, NY, September, 2017.

-
10. Tracing the Arc of Smartphone Application Security, University of Texas-Dallas Department of Computer Science, Dallas, TX, May 2017.
 11. Tracing the Arc of Smartphone Application Security. Rochester Institute of Technology, College of Computing and Information Sciences, Rochester, NY, May 2017.
 12. Tracing the Arc of Smartphone Application Security. University of California-Irvine, Computer Science Department, Irvine CA, March, 2017.
 13. Tracing the Arc of Smartphone Application Security. The Ohio State University, Department of Computer Science and Engineering, Columbus, OH, March, 2017.
 14. Tracing the Arc of Smartphone Application Security. Virginia Technical University, Department of Computer Science, Blacksburg, VA, March, 2017.
 15. Six Years of Mobile Smartphone Security, CISP A Distinguished Lecture Series, Max Planck Institute/Saarland University, Saarbrücken Germany, July, 2015.
 16. Six Years of Mobile Smartphone Security. Technische Universität Darmstadt, Darmstadt Germany, July, 2015.
 17. Security Challenges and Solutions in Mobile Smartphone Applications. Computer and Information Science Department, University of Oregon, Eugene, OR, April, 2011.
 18. Security Challenges and Solutions in Mobile Smartphone Applications. Department of Software Information Systems College of Computing and Informatics, UNC Charlotte, Charlotte, NC, December, 2010.

PATENTS

U.S. Patent 8,732,293, System and method for tracking individuals on a data network using communities of interest, Patrick McDaniel, Subhabrata Sen, Oliver Spatschek, Jacobus E. Van de Merwe, May 20, 2014.

U.S. Patent 8,453,227, Reverse firewall with self-provisioning, William A. Aiello, Charles Robert Kalmanek, Jr., William J. Leighton, III, Patrick McDaniel, Subhabrata Sen, Oliver Spatschek, Jacobus E. Van der Merwe, May 28, 2013.

U.S. Patent 7,975,044, Automated disambiguation of fixed-serverport-based applications from ephemeral applications, Oliver Spatschek, Subhabrata Sen, Jacobus E. Van der Merwe, Patrick McDaniel, May 28, 2013.

U.S. Patent 7,873,350, End-to-end secure wireless communication for requesting a more secure channel, Patrick Drew McDaniel, Martin Joel Strauss, January 18, 2011.

RESEARCH SUPPORT

PI, *SaTC CORE: Frontier: Collaborative: End-to-End Trustworthiness of Machine-Learning Systems*, NSF (CNS), \$9,649,366 (PSU award \$2,044,550), 8/15/2016-3/31/2017, Collaborators: Boneh (Stanford), Chaudhuri (UCSD), Evans (Virginia), Jha (Wisconsin), Liang (Stanford), Song (Berkeley).

PI, *2017 SaTC PI Meeting*, NSF (CNS), \$99,999 (PSU award \$50,230), 8/15/2016-3/31/2017, Collaborators: Antonakakis (GaTech), Mason (UIUC).

PI, *TWC: Medium: Collaborative: Scaling and Prioritizing Market-Sized Application Analysis*, NSF (CNS), \$1,147,213 (PSU award \$547,213), 7/01/2016-6/30/2020, Collaborators: Jha (Wisconsin).

PI, *Student Travel Support for Symposium on Security and Privacy 2014*, Army Research Office, \$10,000, 5/1/14-5/1/15.

PI, *Models for Enabling Continuous Reconfigurability of Secure Missions (MACRO) Cyber-Security Collaborative Research Alliance (CRA)*, Army Research Laboratory, \$24.1 million (\$48.2 million with renewal at 12/17), 9/20/2013-9/19/2023 (renewed at 5 years), Collaborators: PSU, Carnegie Mellon, Indiana, UC Davis, UC Riverside, ARL, CERDEC.

PI, *Google Faculty Research Award, Plotting a Map of Android Inter-App Communication*, Google, \$50,000, 3/1/2012-2/28/2013, Collaborators: PSU (McDaniel), TU Darmstadt (Bodden), University of Luxembourg (Traon), .

PI, *Battelle BGP Security Study (Phase 2)*, Battelle, \$102,815, 10/1/2012-9/30/2013, Collaborators: PSU (McDaniel), Oregon (Butler).

PI, *TWC: Medium: Collaborative: Extending Smart-Phone Application Analysis*, NSF (CNS), \$1,386,518 (plus 16k REU supplement) (PSU award \$534,748), 8/1/2012-7/31/2016, Collaborators: PSU (McDaniel), Wisconsin (Jha).

PI, *Battelle BGP Security Study (Phase 1)*, Battelle, \$94,400, 2/15/2012-9/30/2012, Collaborators: PSU (McDaniel).

co-PI, *TC: Medium: Collaborative Research: Building Trustworthy Applications for Mobile Devices*, NSF (CNS), \$1,386,518 (PSU award \$350,000), 8/1/2011-7/31/2014, Collaborators: PSU (McDaniel), Wisconsin (Banerjee, Jha, Swift).

PI, *Closing the Loop on Security Testing and Security Requirements*, Security and Software Engineering Research Center, \$31,000, 8/1/2011-7/31/2012.

co-PI, *Managing Security and Vulnerability Risks in the Smart Grid*, Institute for CyberScience and The Penn State Institutes of Energy and the Environment, \$31,000, 08/1/09-12/16/09, Collaborators: PSU (Blumsack, McDaniel).

PI, *Smart Grid Cyber Security Research*, Lockheed Martin, \$250,000, 1/1/10-12/16/10.

PI, *NSF HECURA: Collaborative Research: Secure Provenance in High-End Computing Systems*, NSF (CCF), \$1,000,000 (PSU award \$307,073), 08/1/09-8/31/13, Collaborators: PSU (McDaniel), UIUC (Winslett), Stonybrook (Sion, Zadok).

PI, *TC: Medium: Collaborative Research: Security Services in Open Telecommunications Networks*, NSF (CNS), \$1,386,518 (PSU award \$594,941), 08/01/09-08/01/12, Collaborators: PSU (McDaniel, La Porta), UPenn (Blaze), Columbia (Schulzrinne).

PI, *Characterizing and Mitigating Wireless Systems Vulnerabilities*, Defense University Research Instrumentation Program (DURIP), Army Research Office (ARO), \$150,000, 05/22/09-02/28/11, Collaborators: PSU (La Porta, McDaniel).

co-PI, *Integrity Management for ICT Development*, Bell Labs Network Reliability and Security Office, Alcatel-Lucent, \$100,000, 11/30/08-11/30/09, Collaborators: PSU (La Porta, McDaniel).

PI, *Utility Grid Automation and Risk Management*, Lockheed Martin, \$400,000, 11/30/08-12/16/09.

PI, *EVEREST: Evaluation and Validation of Election-Related Equipment, Standards, and Testing*, The State of Ohio, \$716,336 (PSU award \$332,066), 10/01/07-01/07/08, Collaborators: PSU (McDaniel), UPenn (Blaze), UCSB (Kemmerer, Vigna), Berkeley (Hall, Quilter).

Co-PI, *Protecting Services for Emerging Wireless Telecommunications Infrastructure*, NSF (CNS), \$658,032, 09/01/07-08/31/11, Collaborators: PSU (La Porta, Jaeger, McDaniel).

Co-PI, *Security for Internet/IMS Convergence*, Cisco, \$100,000, 9/1/07-8/31/08, Collaborators: PSU (La Porta, McDaniel).

Co-PI, *System-Wide Information Flow Enforcement*, BAA 06-11-IFKA, "National Intelligence Community Enterprise Cyber Assurance Program", \$496,000, 2/1/07-8/1/08, Collaborators: PSU (Jaeger, McDaniel).

PI, *CAREER: Realizing Practical High Assurance through Security-Typed Information Flow Systems*, NSF (CNS), \$400,000, 1/2/07-1/1/12.

Co-PI, *CT-IS: Shamon: Systems Approaches for Constructing Distributed Trust*, NSF (CNS), \$400,000, 9/1/06-8/31/10, Collaborators: PSU (Jaeger, McDaniel).

Co-PI, *Center of Excellence*, Ben Franklin Technology Partners, \$75,000, 01/01/07-07/01/07, Collaborators: PSU (Cao, Jaeger, La Porta, McDaniel, Smith).

Co-PI, *Exploiting Asymmetry in Performance and Security Requirements for I/O in High-end Computing*, NSF (CFF), \$699,690, 9/1/06-8/31/10, Collaborators: PSU (McDaniel, Sivasubramaniam).

PI, *Automated Configuration with the PRESTO Network Management Platform*, AT&T, \$100,000, 6/1/06-5/31/07.

PI, *Testbed for Network-Scale Countermeasure Evaluation*, Cisco, \$45,938, 9/1/05-8/31/06.

PI, *Collaborative Research: CT-T: Flexible, Decentralized Information-flow Control for Dynamic Environments*, NSF (CFF), \$1,057,427 (PSU award \$234,585), 8/1/05-7/31/08, Collaborators: PSU (McDaniel), UPenn (Zdancewic), Maryland (Hicks), GMU (Winsborough).

PI, *Extending Developer Tools for Security-typed Languages*, Software Engineering Research Center, Sponsor: Motorola, \$23,200, 7/1/05-6/30/06.

PI, *Student Travel Support for ACM SIGCOMM 2005 Conference*, NSF, \$19,620, 4/1/05-3/31/06.

Co-PI, *NSF CyberTrust: Collaborative Research: Testing and Benchmarking Methodologies for Future Network Security Mechanisms (EMIST)*, NSF/DHS, \$5,344,459 (PSU award \$2,533,447), 8/1/04-8/31/06, Collaborators: PSU (Kesidis, Miller, Liu), Purdue (Fahmy, Rosenberg, Spafford, Shroff, Brodley), UCDavis (Wu, Levitt, Bishop, Rowe), ICSI/Berkeley (Paxson, Floyd, Weaver).

PROFESSIONAL ACTIVITIES

Editorial Positions, Panels, and Boards

IEEE Technical Committee on Security and Privacy

- *Chair*–January 2014–January 2016
- *Vice Chair*–January 2012–December 2014

ACM Transactions on Internet Technology (TOIT)

- *Editor in Chief*–September 2007–December 2012
- *Associate Editor*–April 2004–August 2007

IEEE Security and Privacy Magazine

- *Area Editor, Secure Systems*–January 2009–2015

IEEE Transactions on Computers (TC)

- *Associate Editor*–August 2008–2014

ACM Transactions on Information and System Security (TISSEC)

- *Associate Editor*–May 2007–May 2012

IEEE Transactions on Software Engineering (TSE)

- *Associate Editor*–January 2007–April 2012
- *Guest Editor, Special Issue on Topics in Security*–Fall 2006–April 2012

IEEE Transactions on Parallel and Distributed Systems (TPDS)

- *Guest Editor, Special Issue on Trust, Security and Privacy in Parallel and Distributed Systems*–Fall 2012

Elsevier Journal of Computer Networks

- *Guest Editor, Special Issue on Web Security*–Fall 2003–Spring 2005

Encyclopedia of Cryptography and Security

- *Editorial Board Member*–Fall 2002–Spring 2005

Journal of Defense Modeling and Simulation

- *Guest Editor, Special Issue on Cyber Risk and Vulnerability Estimation*–Winter 2018–

Other Professional Activities

Helmholtz Center for Information Security (CISPA), Scientific Advisory Board

- *Member*–2019–present

IEEE Computer Society's Technical Committee on Security and Privacy

- *Chair*–2014–2016
- *Vice Chair*–2012–2014

Member, Technical Guideline Development Committee, U.S. Election Assistance Commission

- *Member*–2010–2011

Natural Sciences and Engineering Research Council of Canada, Internetworked Systems Security Network

- *Scientific Advisory Board*–2008–2013

Technology for Cyber Physical System Security Forum, Cyber Security Research and Development, (Senators Joseph I. Lieberman and Susan Collins, Chairs)

- *Speaker and Participant*–September 2008

ACM Student Organization Advisor

- *Penn State Computer Science and Engineering Department*–2006–2012

The Technology Collaborative

- *Penn State Representative (Pennsylvania economic development consortium)*–2007–2008

President's National Security Telecommunications Advisory Panel

- *Member, Next Generation Networks Task Force*–2005–2006

Abusable Technologies Awareness Center (ATAC)

-
- *Panelist*–October 2003-2010
- AT&T IP Services Security Council**
- *Member*–June 2003-August 2004
- AT&T Internet Intellectual Property Review Team**
- *Member*–September 2001-May 2002
- ACM SIGCOMM Student Travel Grant Committee**
- *Member*–August 2005
- National Science Foundation, Grant Review Panel**
- *Member*–2003, 2004, 2006, 2007, 2009, 2010, 2011, 2012, 2014, 2015, 2016, 2017
- Department of Energy SciDAC Review Panel**
- *Member*–2001

Conference and Workshop Participation

- IEEE Symposium on Security and Privacy**
- *Technical Program Co-Chair*–2007, 2008
 - *Program Committee*–2011, 2012, 2013
- IEEE European Symposium on Security and Privacy**
- *Steering Committee*–2015–
 - *Program Committee*–2016, 2017
- USENIX Security Symposium**
- *Program Chair*–2005
 - *Invited Talks Chair*–2006, 2009
 - *Program Committee*–2001, 2003, 2004, 2007, 2014, 2018, 2019, 2020
- ACM Conference on Computer and Communications Security (CCS)**
- *Program Committee*–2006, 2008, 2009, 2010, 2012, 2018, 2019, 2020
 - *Industry and Government Track Chair*–2004, 2007
 - *Program Committee-Industry and Government Track*–2003, 2005, 2006
- International Conference on Privacy, Security and Trust (PST)**
- *Steering Committee*–2019–
- ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)**
- *Program Committee*–2017
- Network and Distributed System Security Symposium (NDSS)**
- *Program Committee*–2009, 2012, 2013, 2017
- Annual Computer Security Applications Conference (ACSAC)**
- *Program Committee*–2004, 2005, 2006, 2007, 2010, 2011
- Financial Cryptography**
- *General Chair*–2006
 - *Program Committee*–2007, 2008, 2012
- Computer Security Foundations Symposium (CSF)**
- *Program Committee*–2011, 2021
- European Symposium on Research in Computer Security (ESORICS)**
- *Program Committee*–2004, 2005
- International Symposium on Engineering Secure Software and Systems (ICISSP)**
- *Program Committee*–2015
- IEEE Conference on Communications and Network Security (CNS)**
- *Program Committee*–2015, 2017
- ACM Annual International Conference on Mobile Computing and Networking (MobiCom)**
- *Program Committee*–2010, 2011, 2012

ACM Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)
• *Program Committee*–2012

ACM Symposium on Access Control Models and Technologies (SACMAT)
• *Program Committee*–2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2011

ACM Conference on ASIA Computer and Communications Security (ASIA CCS)
• *Program Committee*–2008

ACM Conference on Electronic Commerce (ACM EC)
• *Program Committee*–2005

International Conference on Applied Cryptography and Network Security (ACNS)
• *Program Committee*–2006

ACM Annual Digital Forensics Conference
• *Program Committee*–2012

IEEE Workshop on the Internet of Safe Things
• *Program Committee*–2019

ACM Workshop on Moving Target Defense (MTD)
• *Program Committee*–2015, 2016

IEEE ICNP Workshop on Secure Network Protocols (NPSec)
• *Program Committee*–2005, 2006

Conference on Decision and Game Theory for Security (GameSec)
• *Program Committee*–2012, 2018

ACM Symposium on Applied Computing (SAC)
• *Program Committee, Information Security Research and Applications* –2010

USENIX Annual Technical Conference
• *Program Committee*–2002, 2003

World Wide Web Conference (WWW)
• *Security and Privacy Track Vice-Chair*–2005
• *Security and Privacy Track Deputy Vice-Chair*–2004
• *Program Committee*–2003, 2007, 2010, 2011

International Conference on Emerging Trends in Information and Communication Security (ET-RICS)
• *Program Committee*–2006

International Conference On Distributed Computing Systems (ICDCS)
• *Program Committee*–2011

IEEE INFOCOM
• *Program Committee*–2007

IEEE GLOBECOM
• *Program Committee*–2010

MILCOM
• *Program Committee*–2015, 2016, 2017, 2018, 2019

The Five Nines Workshop on Designing and Managing High Availability Internet Services (INM 2007)
• *Program Committee*–2007

International Conference on Information Systems Security (ICISS)
• *Steering Committee*–2007
• *Program Co-Chair*–2007
• *Program Committee*–2005, 2006, 2009, 2011

International Conference on Parallel Processing
• *Program Committee-Network Security*–2003

USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)

-
- *Program Committee*–2010
- ACM Workshop on Networking, Systems, Applications on Mobile Handhelds (MobiHand)**
- *Program Committee*–2009
- ACM Workshop on Cloud Computing Security**
- *Program Committee*–2009, 2010
- ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM)**
- *Program Committee*–2011, 2012, 2013
- International Workshop on Security in Software Engineering**
- *Founding General Co-Chair*–2007
- USENIX Workshop On Offensive Technology (WOOT)**
- *Program Committee*–2007
- ACM Storage Security and Survivability Workshop**
- *Program Committee*–2006
- ACM SIGCOMM Workshop on Internet Network Management**
- *Program Committee*–2006, 2007
- Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec)**
- *Program Committee*–2006, 2007, 2008
- Workshop on Workshop on Telecommunications Infrastructure Protection and Security (TIPS)**
- *General Chair*–2009
- USENIX Workshop on Hot Topics in Security (HotSec)**
- *Program Chair*–2011
 - *Program Committee*–2007, 2008, 2009, 2010, 2012
- ACM Workshop TPC on Security and Privacy in Smartphones and Mobile Devices**
- *Program Committee*–2011
- International Workshop on Security (IWSEC)**
- *Program Committee*–2006
- International Workshop on Systems and Network Security (SNS)**
- *Program Committee*–2005, 2006

COLLABORATORS (Last 48 Months)

Yasemin Acar, Abbas Acar, Stefan Achleitner, Gail-Joon Ahn, Hidayet Aksu, Alexander Alexeev, Raquel Alvarez, Manos Antonakakis, Ahmed Atya, Leonardo Babun, Michael Backes, Alexandre Bartel, Novella Bartolini, Z. Berkay Celik, Dan Boneh, Thomas Bowen, Scott Brown, Sven Bugiel, Kevin Butler, Hasan Cam, Berkay Celik, Ritu Chada, Ritu Chadha, Joseph Choi, David Dagon, Drew Davidson, Lorenzo De Carli, Matthew Dering, Erik Derr, Daniel E. Krych, Sascha Fahl, Reuben Feinman, Earlene Fernandes, Matt Fredrikson, Ian Goodfellow, Kathrin Grosse, Jens Grossklags, Richard Harang, Diane Henshel, Yunfeng Hong, Nan Hu, Yongjian Hu, Charles Huber, Rauf Izmailov, Edward J. M. Colbert, Devin J. Pohly, Trent Jaeger, Somesh Jha, Eric Kilmer, Jacques Klein, Srikanth Krishnamurthy, Padma Krishnaswamy, Amit Kumar Sikder, Alexey Kurakin, Thomas La Porta, Nathaniel Lageman, Chun-Ming Lai, Yves Le Traon, Chaz Lever, Karl Levitt, Li Li, Yun Li, Yu-Cheng Lin, David Lopez-Paz, Daniel Luchaup, Sayed M. Saghaian, Praveen Manoharan, Lisa Marvel, Yacin Nadjji, Iulian Neamtiu, Damien Ocateau, Nicolas Papernot, Yu Paul, Eric Pauley, Giuseppe Petracca, Alexander Poylisher, Zhiyun Qian, Vaibhav Rastogi, Ahmad-Atamli Reineh, Brian Rivera, Jeff Rowe, A. Selcuk Uluagac, Constantin Serban, Ryan Sheatsley, Arunesh Sinha, Matthew Smith, Chengyu Song, Shridatt Sugrim, Yuqiong Sun, Ananthram Swami, Gang Tan, Steve Templeton, Dave Tian, Florian Tramer, Dang Tu Nguyen, Selcuk Uluagac, Srikanth V. Krishnamurthy, Robert Walls, Xiaoyun Wang, Sebastian Weisgerber, Michael Weisman, Michael Wellman, Xi Wu, Felix Wu

CASES (Expert Witness)

Rimini Street, Inc. v. Oracle International Corporation, Expert for Expert for Oracle, United States District Court, District of Nevada, Case no. Case No. 2:14-cv-01699.

Inter Partes Review, Expert for Expert for Google Inc., United States Patent and Trademark Office, Case no. Patent No. 9,444,812.

Good Technology Corporation et al., v. AirWatch, LLC, Expert for defense, United States District Court for the Northern District of California, Case no. 5-12-cv-05827.

Inter Partes Review, Expert for Duo Security Inc., United States Patent and Trademark Office, Case no. IPR2017-01041.

Inter Partes Review, Expert for Duo Security Inc., United States Patent and Trademark Office, Case no. IPR2017-01064.

Frederick Whalen, et al., v. SEI/AARON'S, INC., Expert for defense, United States District Court for the Northern District of Georgia, Case no. 1:2014-cv-01209.

Certain Portable Electronic Communications Devices, Including Mobile Phones and Components Thereof, Expert for plaintiff, International Trade Commission.

Secure Access, LLC v. Bank of America Corp., et al., Expert for defense, United States District Court for the Eastern District of Texas, Tyler Division, Case no. 6-10-cv-00670.

Intellectual Ventures I LLC, v. Check Point et al., Expert for plaintiff, United States District Court for the District of Delaware, Case no. 1-10-cv-01067.

NetMonitor LLC, v. Compuware et al., Expert for plaintiff, United States District Court for the District of Delaware, Case no. 1-10-cv-01061.

Amdocs Ltd. v. Openet Telecom Ltd., Expert for defense, United States District Court for the Eastern District of Virginia, Alexandria Division, Case no. 1-10-cv-00910.

PSI Systems Inc. v. Stamps.com, Expert for defense, United States District Court for the Central District of California, Case no. 2-08-cv-05233.

Stamps.com v. Endicia, Expert for plaintiff, United States District Court for the Central District of California, Case no. 2-06-cv-07499.

Kara Technologies v. Stamps.com, Expert for defense, United States District Court for the Central District of California, Case no. 2-05-cv-01890.

Coinstar Inc. v. Coinxchange, Expert for defense, United States District Court for the Eastern District of Virginia, Richmond Division, Case no. 3-06-cv-00299.

VCode Holdings Inc. v. Stamps.com, Expert for defense, United States District Court for the Central District of California.

Inter Partes Review, Expert for Google Inc., United States Patent and Trademark Office, Case no. (Patent No. 8,489,868).

Inter Partes Review, Expert for Cisco Systems, Inc., , Case no. (Patent No. 7,536,598).

Vir2us v. Cisco Systems, INC and Sourcefire, LLC, Expert for defense, United States District Court for the Eastern District of Virginia, Case no. 51:16cv1095.

Zito Vault, LLC V. International Business Machines Corporation, and Softlayer Technologies, Inc., Expert for defense, , Case no. 3:16-CV-962-M.

PUBLICATIONS

Books

Patrick Traynor, Patrick McDaniel, and Thomas La Porta, Security for Telecommunications Networks. Springer, Series: Advances in Information Security, July, 2008. ISBN: 978-0-387-72441-6.

Book Chapters

Kevin Butler, William Enck, Patrick Traynor, Jennifer Plasterr, and Patrick McDaniel. Privacy Preserving Web-Based Email. *Algorithms, Architectures and Information Systems Security, Statistical Science and Interdisciplinary Research*, World Scientific Computing, pages 349-371. November 2008. *Editor*: Bhargab Bhattacharya, Susmita Sur-Kolay, Subhas Nandy and Aditya Bagchi. (extends iciss06b)

Patrick McDaniel. Authentication. *Handbook of Computer Networks, Volume II, Chapter 171*, John Wiley and Sons. May 2007. *Editor*: Hossein Bidgoli.

Patrick McDaniel. Computer and Network Authentication. *Handbook of Information Security*, John Wiley and Sons. September 2004. *Editor*: Hossein Bidgoli.

Patrick McDaniel. IPsec. *Encyclopedia of Information Security*, Kluwer. 2003. *Editor*: Hossein Bidgoli.

Patrick McDaniel. Policy. *Encyclopedia of Information Security*, Kluwer. 2003. *Editor*: Hossein Bidgoli.

Patrick McDaniel. Authentication. *The Internet Encyclopedia*, John Wiley and Sons. 2002.

Columns

Patrick McDaniel, Nicolas Papernot, and Berkay Celik, Machine Learning in Adversarial Settings. *IEEE Security & Privacy Magazine*, 14(3), May/June, 2016.

Patrick McDaniel, Brian Rivera, and Ananthram Swami, Toward a Science of Secure Environments. *IEEE Security & Privacy Magazine*, 12(4), July/August, 2014.

Patrick McDaniel, Bloatware Comes to the Smartphone. *IEEE Security & Privacy Magazine*, 10(4), July/August, 2011.

Patrick McDaniel, Data Provenance and Security. *IEEE Security & Privacy Magazine*, 9(3), March/April, 2011.

Patrick McDaniel and William Enck, Not So Great Expectations: Why Application Markets Haven't Failed Security. *IEEE Security & Privacy Magazine*, 8(5):76-78, September/October, 2010.

Patrick McDaniel and Stephen McLaughlin, Security and Privacy Challenges in the Smart Grid. *IEEE Security & Privacy Magazine (Secure Systems issue column)*, 7(3):75-77, May/June, 2009.

Journal Publications

Z. Berkay Celik, Patrick McDaniel, and Thomas Bowen, Malware Modeling and Experimentation through Parameterized Behavior. *Journal of Defense Modeling and Simulation (JDMS)*, 15(1):31-48, 2017.

Z. Berkay Celik, Earlene Fernandes, Eric Pauley, Gang Tan, and Patrick McDaniel, Program Analysis of Commodity IoT Applications for Security and Privacy: Opportunities and Challenges. *ACM Computing Surveys (CSUR)*, ACM, 42(4), 2019.

Z. Berkay Celik, Patrick McDaniel, Gang Tan, Leonardo Babun, and Selcuk Uluagac, Verifying IoT Safety and Security in Physical Spaces. *IEEE Security & Privacy Magazine*, IEEE, 17(5):30-37, 2019.

Ahmed Atya, Zhiyun Qian, Srikanth Krishnamurthy, Thomas La Porta, Patrick McDaniel, and Lisa Marvel, Catch Me if You Can: Malicious Co-Residency on the Cloud. *IEEE/ACM Transactions on Networking*, 27(2), April, 2019.

Daniel E. Krych and Patrick McDaniel, Exposing Android Social Applications: Linking Data Leakage to Privacy Policies. *Journal of Cyber Security Technology*, Taylor & Francis, 2019.

Ian Goodfellow, Patrick McDaniel, and Nicolas Papernot, Making machine learning robust against adversarial inputs. *Communications of the ACM*, ACM, 61(7):56-66, June/July, 2018.

Dave Tian, Kevin Butler, Joseph Choi, Patrick McDaniel, and Padma Krishnaswamy, Securing ARP/NDP From the Ground Up. *IEEE Transactions on Information Forensics and Security*, April, 2017.

Stefan Achleitner, Thomas La Porta, Patrick McDaniel, Shridatt Sugrim, Srikanth Krishnamurthy, and Ritu Chada, Deceiving Network Reconnaissance Using SDN-based Virtual Topologies. *IEEE Transactions on Network and Service Management, Special Issue on Advances in Management of Softwarized Networks*, 14(4), July, 2017.

Chaz Lever, Robert Walls, Yacin Nadji, David Dagon, Patrick McDaniel, and Manos Antonakakis, Dawn of the Dead Domain: Measuring the Exploitation of Residual Trust in Domains. *IEEE Security & Privacy Magazine (Secure Systems issue column)*, April, 2017.

Patrick McDaniel and Ananthram Swami, The Cyber Security Collaborative Research Alliance: Unifying Detection, Agility, and Risk in Mission-Oriented Cyber Decision Making. *CSIAAC Journal, Army Research Laboratory (ARL) Cyber Science and Technology*, 5(1), January, 2017.

Damien Oceau, Daniel Luchaup, Somesh Jha, and Patrick McDaniel, Composite Constant Propagation and its Application to Android Program Analysis. *IEEE Transactions on Software Engineering*, 42(11):999-1014, 2016.

Alexander Kott, Ananthram Swami, and Patrick Drew McDaniel, Security Outlook: Six Cyber Game Changers for the Next 15 Years. *IEEE Computer*, 47(12):104-106, 2014.

Zhenfu Cao, Keqiu Li, Xu Li, Patrick McDaniel, Radha Poovendran, Guojun Wang, and Yang Xiang, Guest Editors' Introduction: Special Issue on Trust, Security, and Privacy in Parallel and Distributed Systems. *IEEE Transactions on Parallel and Distributed Systems*, 25(2):279-282, 2014.

William Enck, Peter Gilbert, Seungyeop Han, Vasant Tendulkar, Byung-Gon Chun, Landon Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol Sheth, TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones. *ACM Transactions on Computer Systems (TOCS)*, 32(2), June, 2014. (extends egc+10)

William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol N. Sheth, TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones. *Communications of the ACM*, 57(3), March, 2014. Research Highlight.

Machigar Ongtang, Stephen McLaughlin, William Enck, and Patrick McDaniel, Semantically Rich Application-Centric Security in Android. *Security and Communication Networks*, 5(6):658-673, 2012.

Thomas Moyer, Kevin Butler, Joshua Schiffman, Patrick McDaniel, and Trent Jaeger, Scalable Web Content Attestation. *IEEE Transactions on Computers*, 61(5):686-699, April, 2011. (extends mbs+09)

Joshua Schiffman, Thomas Moyer, Trent Jaeger, and Patrick McDaniel, Network-based Root of Trust for Installation. *IEEE Security & Privacy Magazine*, pages 40-48, Jan/Feb, 2011.

Kevin Butler, Stephen McLaughlin, Thomas Moyer, and Patrick McDaniel, New Security Architectures Based on Emerging Disk Functionality. *IEEE Security and Privacy Magazine*, 8(5), October, 2010.

Patrick Traynor, Chaitrali Amrutkar, Vikhyath Rao, Trent Jaeger, Patrick McDaniel, and Thomas La Porta, From Mobile Phones to Responsible Devices. *Journal of Security and Communication Networks (SCN)*, 4(6):719-726, June, 2011.

Patrick Traynor, Kevin Butler, William Enck, Kevin Borders, and Patrick McDaniel, malnets: Large-Scale Malicious Networks via Compromised Wireless Access Points. *Journal of Security and Communication Networks (SCN)*, 2(3):102-113, March, 2010.

Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Brent Waters, Secure Attribute-Based Systems. *Journal of Computer Security (JCS)*, 18(5):799-837, 2010. (extends ptmw06)

Boniface Hicks, Sandra Rueda, Luke St. Clair, Trent Jaeger, and Patrick McDaniel, A Logical Specification and Analysis for SELinux MLS Policy. *ACM Transactions on Information and System Security (TISSEC)*, 13(26), 2010. (extends hrs+07)

Kevin Butler, Toni Farley, Patrick McDaniel, and J. Rexford, A Survey of BGP Security Issues and Solutions. *Proceedings of the IEEE*, 2010(1):100-122, January, 2010.

Kevin Butler, Sunam Ryu, Patrick Traynor, and Patrick McDaniel, Leveraging Identity-based Cryptography for Node ID Assignment in Structured P2P Systems. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 20(12):1803-1815, December, 2009. (extends rbtm07)

Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta, Mitigating Attacks on Open Functionality in SMS-Capable Cellular Networks. *IEEE/ACM Transactions on Networking (TON)*, 17(1):40-53, February, 2009.

William Enck, Machigar Ongtang, and Patrick McDaniel, Understanding Android Security. *IEEE Security & Privacy Magazine*, 7(1):50–57, January/February, 2009.

William Enck, Thomas Moyer, Patrick McDaniel, Shubho Sen, Panagiotis Sebos, Sylke Spoerel, Albert Greenberg, Yu-Wei Sung, Sanjay Rao, and William Aiello, Configuration Management at Massive Scale: System Design and Experience. *IEEE Journal on Selected Areas in Communications (JSAC)*, 27(3):323–335, 2009. (extends ems+07)

Heesook Choi, William Enck, Jaesheung Shin, Patrick McDaniel, and Thomas La Porta, ASR: Anonymous and Secure Reporting of Traffic Forwarding Activity in Mobile Ad Hoc Networks. *Wireless Networks (WINET)*, ACM/Kluwer, 15(4):525–539, MAY, 2009. (extends ces+05)

Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta, Exploiting Open Functionality in SMS-Capable Cellular Networks. *Journal of Computer Security*, 16(6):713–742, Febraury, 2009. (extends etml05)

Patrick Traynor, Michael Chien, Scott Weaver, Boniface Hicks, and Patrick McDaniel, Non-Invasive Methods for Host Certification. *ACM Transactions on Information and System Security (TISSEC)*, 11(3), 2008. (extends tcw+06)

Patrick McDaniel and Bashar Nuseibeh, *Guest Editorial: Special Issue on Software Engineering for Secure Systems*. *IEEE Transactions on Software Engineering*, 34(1):3–4, 2008.

Wesam Lootah, William Enck, and Patrick McDaniel, TARP: Ticket-based Address Resolution Protocol. *Computer Networks*, Elsevier, 51(15):4322–4337, October, 2007. (extends lem05)

Patrick McDaniel and Avi Rubin, *Guest Editorial: Special Issue on Web Security*. *Computer Networks*, Elsevier, 22(2), 2005.

Patrick McDaniel and Atul Prakash, Enforcing Provisioning and Authorization Policy in the Antigone System. *Journal of Computer Security*, 14(6):483–511, November, 2006.

Patrick McDaniel and Atul Prakash, Methods and Limitations of Security Policy Reconciliation. *ACM Transactions on Information and System Security (TISSEC)*, Association for Computing Machinery, 9(3):259–291, August, 2006. (extends mp02)

Patrick McDaniel, William Aiello, Kevin Butler, and John Ioannidis, Origin Authentication in Interdomain Routing. *Journal of Communication Networks*, Elsevier, 50(16):2953–2980, November, 2006. (extends aim03)

Matthew Pirretti, Sencun Zhu, Vijaykrishnan Narayanan, Patrick McDaniel, Mahmut Kandemir, and Richard Brooks, The Sleep Deprivation Attack in Sensor Networks: Analysis and Methods of Defense. *International Journal of Distributed Sensor Networks*, 2(3):267–287, June, 2005. (extends pzn+05)

Simon Byers, Lorrie Cranor, Eric Cronin, Dave Kormann, and Patrick McDaniel, Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *Telecommunications Policy*, 28(8):619–644, August, 2004. (extends bcc+03)

Conference Publications

Flow Table Security in SDN: Adversarial Reconnaissance and Intelligent Attacks. *IEEE INFOCOM*, IEEE Conference on Computer Communications 2020. Beijing, China.

Giuseppe Petracca, Yuqiong Sun, Ahmad-Atamli Reineh, Jens Grossklags, Patrick McDaniel, and Trent Jaeger. EnTrust: Regulating Sensor Access by Cooperating Programs via Delegation Graphs. *Proceedings of the 28th USENIX Security Symposium*, August 2019. Santa Clara, CA.

Z. Berkay Celik, Abbas Acar, Hidayet Aksu, Abbas Acar, Ryan Sheatsley, A. Selcuk Uluagac, and Patrick McDaniel. Curie: Policy-based Secure Data Exchange. *ACM Conference on Data and Applications Security (CODASPY)*, March 2019. Dallas, TX.

Z. Berkay Celik, Gang Tan, and Patrick McDaniel. IoTGuard: Dynamic Enforcement of Security and Safety Policy in Commodity IoT. *Network and Distributed System Security Symposium (NDSS)*, February 2019. San Diego, CA.

Dang Tu Nguyen, Chengyu Song, Zhiyun Qian, Srikanth V. Krishnamurthy, Edward J. M. Colbert, and Patrick McDaniel. IoTSan: Fortifying the Safety of IoT Systems. *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies (CoNEXT '18)*, December 2018.

Z. Berkay Celik, Leonardo Babun, Amit Kumar Sikder, Hidayet Aksu, Gang Tan, Patrick McDaniel, and A. Selcuk Uluagac. Sensitive Information Tracking in Commodity IoT. *Proceedings of the 27th USENIX Security Symposium*, August 2018. Baltimore, MD.

Rauf Izmailov, Shridatt Sugrim, Ritu Chadha, Patrick McDaniel, and Ananthram Swami. Enablers Of Adversarial Attacks in Machine Learning. *Proceedings of the Military Communications Conference (MILCOM)*, IEEE, October 2018.

Sayed M. Saghaian, Thomas La Porta, Trent Jaeger, Z. Berkay Celik, and Patrick McDaniel. Mission-oriented Security Model, Incorporating Security Risk, Cost and Payout. *Proceedings of EAI SECURECOMM 2018*, August 2018. **(best paper award)**.

Z. Berkay Celik, Patrick McDaniel, and Gang Tan. Soteria: Automated IoT Safety and Security Analysis. *USENIX Annual Technical Conference (USENIX ATC)*, July 2018. Boston, MA.

Z. Berkay Celik, Patrick McDaniel, Rauf Izmailov, Nicolas Papernot, Ryan Sheatsley, Raquel Alvarez, and Ananthram Swami. Detection under Privileged Information. *ACM Asia Conference on Computer and Communications Security (ASIACCS)*, June 2018.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble Adversarial Training: Attacks and Defenses. *International Conference on Learning Representations (ICLR)* 2018. Vancouver, Canada. accepted as poster.

Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. SoK: Security and Privacy in Machine Learning. *Security and Privacy (EuroS&P)*, 2018 IEEE European Symposium on on Security and Privacy (EuroS&P), IEEE, April 2018. London, UK.

Chun-Ming Lai, Xiaoyun Wang, Yunfeng Hong, Yu-Cheng Lin, Felix Wu, Patrick McDaniel, and Hasan Cam. Attacking Strategies and Temporal Analysis Involving Facebook Discussion Groups. *13th IEEE International Conference on Network and Service Management*, November 2017. Tokyo, Japan.

Kathrin Grosse, Nicolas Papernot, Praveen Manoharan, Michael Backes, and Patrick McDaniel. Adversarial Examples for Malware Detection. *22nd European Symposium on Research in Computer Security (ESORICS '17)*, September 2017. Oslo, Norway.

Z. Berkay Celik, David Lopez-Paz, and Patrick McDaniel. Patient-Driven Privacy Control through Generalized Distillation. *Proceedings of the Privacy-Aware Computing (PAC)*, IEEE 2017.

Abbas Acar, Z. Berkay Celik, Hidayet Aksu, A. Selcuk Uluagac, and Patrick McDaniel. Achieving Secure and Differentially Private Computations in Multiparty Settings. *Proceedings of the Privacy-Aware Computing (PAC)*, IEEE 2017.

Vaibhav Rastogi, Drew Davidson, Lorenzo De Carli, Somesh Jha, and Patrick McDaniel. Cimplifier: Automatically Debloating Containers. *11Th Joint Meeting of the European Software Engineering Conference and the Acm Sigsoft Symposium on the Foundations of Software Engineering*, September 2017. Paderborn, Germany. (acceptance rate=24%)

Yunfeng Hong, Yongjian Hu, Chun-Ming Lai, Felix Wu, Iulian Neamtii, Yu Paul, Patrick McDaniel, Hasan Cam, and Gail-Joon Ahn. Defining and Detecting Environment Discrimination in Android Apps. *The 13th EAI International Conference on Security and Privacy in Communication Networks (SecureComm 17)*, October 2017.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical Black-Box Attacks against Machine Learning. *ACM Asia Conference on Computer and Communications Security (ASIACCS) 2017*, April 2017. (acceptance rate=209%)

Stefan Achleitner, Thomas La Porta, Trent Jaeger, and Patrick McDaniel. Adversarial Network Forensics in Software Defined Networking. *ACM Symposium on SDN Research (SOSR)*, ACM, April 2017. **(best student paper award)**.

Ahmed Atya, Zhiyun Qian, Srikanth V. Krishnamurthy, Thomas La Porta, Patrick McDaniel, and Lisa Marvel. Stealth Migration: Hiding Virtual Machines on the Network. *IEEE International Conference on Computer Communications (INFOCOM)*, IEEE 2017.

Stefan Achleitner, Thomas La Porta, Patrick McDaniel, Srikanth V. Krishnamurthy, Alexander Poylisher, and Constantin Serban. Malicious Co-Residency on the Cloud: Attacks and Defense. *IEEE International Conference on Computer Communications (INFOCOM)*, IEEE 2017.

Nathaniel Lageman, Eric Kilmer, Robert Walls, and Patrick McDaniel. BinDNN: Resilient Function Matching Using Deep Learning. *2016 International Conference on Security and Privacy in Communication Networks (SECURECOMM)*, October 2016.

Z. Berkay Celik, Nan Hu, Yun Li, Nicolas Papernot, Patrick McDaniel, Jeff Rowe, Robert Walls, Karl Levitt,

Novella Bartolini, Thomas La Porta, and Ritu Chadha. Mapping Sample Scenarios to Operational Models. *Proceedings of the Military Communications Conference (MILCOM)*, IEEE 2016.

Nicolas Papernot, Patrick McDaniel, Ananthram Swami, and Richard Harang. Crafting Adversarial Input Sequences for Recurrent Neural Networks. *Proceedings of the Military Communications Conference (MILCOM)*, IEEE 2016.

Michael Backes, Sven Bugiel, Erik Derr, Patrick McDaniel, Damien Ocateau, and Sebastian Weisgerber. On Demystifying the Android Application Framework: Re-Visiting Android Permission Specification Analysis. *Proceedings of the 25th USENIX Security Symposium*, August 2016.

Devin J. Pohly and Patrick McDaniel. Modeling Privacy and Tradeoffs in Multichannel Secret Sharing Protocols. *46th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, June 2016.

Chaz Lever, Robert Walls, Yacin Nadji, David Dagon, Patrick McDaniel, and Manos Antonakakis. Domain-Z: 28 Registrations Later. *Proceedings of the 37th IEEE Symposium on Security and Privacy*, May 2016. San Francisco, CA.

Yasemin Acar, Michael Backes, Sven Bugiel, Sascha Fahl, Patrick McDaniel, and Matthew Smith. SoK: Lessons Learned From Android Security Research For Appified Software Platforms. *Proceedings of the 37th IEEE Symposium on Security and Privacy*, May 2016. San Francisco, CA.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks. *Proceedings of the 37th IEEE Symposium on Security and Privacy*, May 2016. San Francisco, CA.

Charles Huber, Scott Brown, Patrick McDaniel, and Lisa Marvel. Cyber Fighter Associate: A Decision Support System for Cyber Agility. *Proceedings of the 50th Annual Conference on Information Sciences and Systems (CISS)*, March 2016. Princeton, NJ.

Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The Limitations of Deep Learning in Adversarial Settings. *Proceedings of the 1st IEEE European Symposium on Security and Privacy*, IEEE 2016. Saarbrücken, Germany.

Damien Ocateau, Somesh Jha, Matthew Dering, Patrick McDaniel, Alexandre Bartel, Li Li, Jacques Klein, and Yves Le Traon. Combining Static Analysis with Probabilistic Models to Enable Market-Scale Android Inter-Component Analysis. *Proceedings of the 43rd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL)*, January 2016. St. Petersburg, Florida, USA.

Devin J. Pohly and Patrick McDaniel. MICSS: A Realistic Multichannel Secrecy Protocol. *IEEE Global Communications Conference (GLOBECOM)*, December 2015. San Diego, CA.

Robert Walls, Eric Kilmer, Nathaniel Lageman, and Patrick D. McDaniel. Measuring the Impact and Perception of Acceptable Advertisements. *Proceedings of the ACM 2015 Internet Measurement Conference (IMC)*, October 2015. Tokyo, Japan.

Nicolas Papernot, Patrick McDaniel, and Robert Walls. Enforcing Agile Access Control Policies in Relational Databases using Views. *Proceedings of the Military Communications Conference (MILCOM)*, October 2015. Tampa, FL.

Alessandro Oltramari, Lorrie Cranor, Robert Walls, and Patrick McDaniel. Computational Ontology of Network Operations. *Proceedings of the Military Communications Conference (MILCOM)*, October 2015. Tampa, FL.

Berkay Celik, Robert Walls, Patrick McDaniel, and Ananthram Swami. Malware Traffic Detection using Tamper Resistant Features. *Proceedings of the Military Communications Conference (MILCOM)*, October 2015. Tampa, FL.

Devin Pohly, Charles Sestito, and Patrick McDaniel. Adaptive Protocol Switching Using Dynamically Insertable Bumps in the Stack. *Proceedings of the Military Communications Conference (MILCOM)*, October 2015. Tampa, FL.

Azeem Aqil, Ahmed Atya, Trent Jaeger, Srikanth Krishnamurthy, Karl Levitt, Patrick McDaniel, Jeff Rowe, and Ananthram Swami. Detection of Stealthy TCP-based DoS Attacks. *Proceedings of the Military Communications Conference (MILCOM)*, October 2015. Tampa, FL.

Daniel E. Krych, Stephen Lange-Maney, Patrick McDaniel, and William Glodek. Investigating Weaknesses in Android Certificate Security. *SPIE 9478, Modeling and Simulation for Defense Systems and Applications X*, May 2015.

Damien Octeau, Daniel Luchaup, Matthew Dering, Somesh Jha, and Patrick McDaniel. Composite Constant Propagation: Application to Android Inter-Component Communication Analysis. *Proceedings of the 37th International Conference on Software Engineering (ICSE)*, May 2015. Florence, Italy.

Li Li, Alexandre Bartel, Tegawende Bissyande, Jacques Klein, Yves Le Traon, Steven Arzt, Siegfried Rasthofer, Eric Bodden, Damien Octeau, and Patrick McDaniel. IccTA: Detecting Inter-Component Privacy Leaks in Android Apps. *Proceedings of the 37th International Conference on Software Engineering (ICSE)*, May 2015. Florence, Italy.

Jing Tian, Kevin Butler, Patrick McDaniel, and Padma Krishnaswamy. Securing ARP From the Ground Up. *CODASPY '15: Proceedings of the 5th ACM Conference on Data Application and Security and Privacy*, March 2015. San Antonio, TX, USA.

Alessandro Oltramari, Lorrie Cranor, Robert Walls, and Patrick McDaniel. Building an Ontology of Cyber Security. *Proc. Intl. Conference on Semantic Technologies for Intelligence, Defense, and Security (STIDS)*, November 2014.

Matthew Dering and Patrick McDaniel. Android Market Reconstruction and Analysis. *Proceedings of the Military Communications Conference (MILCOM)*, October 2014. Baltimore, MD.

Wenhui Hu, Damien Octeau, Patrick McDaniel, and Peng Liu. Duet: Library Integrity Verification for Android Applications. *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, July 2014. Oxford, United Kingdom.

Steven Arzt, Siegfried Rasthofer, Christian Fritz, Eric Bodden, Alexandre Bartel, Jacques Klein, Yves Le Traon, Damien Octeau, and Patrick McDaniel. FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps. *Proceedings of the 35th Conference on Programming Language Design and Implementation (PLDI)*, June 2014. Edinburgh, UK.

Phillip Koshy, Diana Koshy, and Patrick McDaniel. An Analysis of Anonymity in Bitcoin Using P2P Network Traffic. *Proceedings of Financial Cryptography 2014, International Financial Cryptography Association (IFCA)*, February 2014. Christ Church, Barbados.

Stephen McLaughlin, Devin Pohly, Patrick McDaniel, and Saman Zonouz. A Trusted Safety Verifier for Process Controller Code. *Proc. ISOC Network and Distributed Systems Security Symposium (NDSS)*, February 2014. San Diego, CA.

Damien Octeau, Patrick McDaniel, Somesh Jha, Alexandre Bartel, Eric Bodden, Jacques Klein, and Yves Le Traon. Effective Inter-Component Communication Mapping in Android with Epicc: An Essential Step Towards Holistic Security Analysis. *Proceedings of the 22th USENIX Security Symposium*, August 2013. Washington, DC. (acceptance rate=16.2%)

Devin J. Pohly, Stephen McLaughlin, Patrick McDaniel, and Kevin Butler. Hi-Fi: Collecting High-Fidelity Whole-System Provenance. *Proceedings of the 28th Annual Computer Security Applications Conference (ACSAC)*, December 2012. Orlando, Florida.

Stephen McLaughlin and Patrick McDaniel. SABOT: Specification-based Payload Generation for Programmable Logic Controllers. *19th ACM Conference on Computer and Communications Security (CCS)*, October 2012. (acceptance rate=18.9%)

Weining Yang, Ninghui Li, Yuan Qi, Wahbeh Qardaji, Stephen McLaughlin, and Patrick McDaniel. Minimizing Private Data Disclosures in the Smart Grid. *19th ACM Conference on Computer and Communications Security (CCS)*, October 2012. (acceptance rate=18.9%)

Eun Kyoung Kim, Patrick McDaniel, and Thomas La Porta. A Detection Mechanism for SMS Flooding Attacks in Cellular Networks. *Proceedings of the 8th International Conference on Security and Privacy in Communication Networks (SECURECOMM 2012)*, September 2012. Padua, Italy. (acceptance rate=30%)

Damien Octeau, Somesh Jha, and Patrick McDaniel. Retargeting Android Applications to Java Bytecode. *20th International Symposium on the Foundations of Software Engineering (FSE)*, November 2012. Research Triangle Park, NC. (**best artifact award**). (acceptance rate=17.4%)

Thomas Moyer, Trent Jaeger, and Patrick McDaniel. Scalable Integrity-Guaranteed AJAX. *Proceedings of the 14th Asia-Pacific Web Conference (APWeb)*, April 2012. Kunming, China. *Invited Paper*.

Patrick McDaniel and Stephen McLaughlin. Structured Security Testing in the Smartgrid. *Proceedings of 5th International Symposium on Communications, Control, and Signal Processing*, May 2012. Rome, Italy. *Invited Paper*. (extends mdd+10)

Stephen McLaughlin, Patrick McDaniel, and William Aiello. Protecting Consumer Privacy from Electric Load Monitoring. *The 18th ACM Conference on Computer and Communications Security (CCS)*, October 2011. Chicago, IL. (acceptance rate=13.9%)

William Enck, Damien Ocateau, Patrick McDaniel, and Swarat Chaudhuri. A Study of Android Application Security. *Proceedings of the 20th USENIX Security Symposium*, August 2011. San Francisco, CA. (acceptance rate=17.2%)

Kevin Butler, Stephen McLaughlin, and Patrick McDaniel. Kells: A Protection Framework for Portable Data. *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC)*, December 2010. Austin, TX. (acceptance rate=16.3%)

Stephen McLaughlin, Dmitry Podkuiko, Adam Delozier, Sergei Miadzezhanka, and Patrick McDaniel. Multi-vendor Penetration Testing in the Advanced Metering Infrastructure. *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC)*, December 2010. Austin, TX. (acceptance rate=16.3%)

Machigar Ongtang, Kevin Butler, and Patrick McDaniel. Porscha: Policy Oriented Secure Content Handling in Android. *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC)*, December 2010. Austin, TX. (acceptance rate=16.3%)

Patrick Traynor, Joshua Schiffman, Thomas La Porta, Patrick McDaniel, Abhrajit Ghosh, and Farooq Anjum. Constructing Secure Localization Systems with Adjustable Granularity. *IEEE Global Communications Conference (GLOBECOM)*, December 2010. Miami, FL. (acceptance rate=36%)

William Enck, Peter Gilbert, Byung-gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol N. Sheth. TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones. *Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, October 2010. Vancouver, BC. (acceptance rate=16.1%)

Toby Ehrenkrantz, Jun Li, and Patrick McDaniel. Realizing A Source Authentic Internet. *Proceedings of the 6th International ICST Conference on Security and Privacy in Communications Networks (Securecomm)*, September 2010. Singapore. (acceptance rate=25%)

Boniface Hicks, Sandra Rueda, David King, Thomas Moyer, Joshua Schiffman, Yogesh Sreenivasan, Patrick McDaniel, and Trent Jaeger. An Architecture for Enforcing End-to-End Access Control over Web Applications. *Proceedings of the Fifteenth ACM Symposium on Access Control Models and Technologies (SACMAT 2010)*, pages 163-172, June 2010. Pittsburgh, PA.

Kevin Butler, Stephen McLaughlin, and Patrick McDaniel. Disk-Enabled Authenticated Encryption. *Proceedings of the 26th IEEE Symposium on Massive Storage Systems and Technologies (MSST)*, May 2010. (acceptance rate=45.5%)

Machigar Ongtang, Stephen McLaughlin, William Enck, and Patrick McDaniel. Semantically Rich Application-Centric Security in Android. *Proceedings of the 25th Annual Computer Security Applications Conference (ACSAC)*, pages 340-349, December 2009. Honolulu, Hawaii. (**best paper**). (acceptance rate=19.0%)

Thomas Moyer, Kevin Butler, Joshua Schiffman, Patrick McDaniel, and Trent Jaeger. Scalable Asynchronous Web Content Attestation. *Proceedings of the 25th Annual Computer Security Applications Conference (ACSAC)*, pages 95-104, December 2009. Honolulu, Hawaii. (acceptance rate=19.0%)

Joshua Schiffman, Thomas Moyer, Christopher Shal, Trent Jaeger, and Patrick McDaniel. Justifying Integrity Using a Virtual Machine Verifier. *Proceedings of the 25th Annual Computer Security Applications Conference (ACSAC)*, pages 83-92, December 2009. Honolulu, Hawaii. (acceptance rate=19.0%)

William Enck, Machigar Ongtang, and Patrick McDaniel. On Lightweight Mobile Phone App Certification. *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS)*, pages 235-245, November 2009. (acceptance rate=18.4%)

Patrick Traynor, Michael Lin, Machigar Ongtang, Vikhyath Rao, Trent Jaeger, Thomas La Porta, and Patrick McDaniel. On Cellular Botnets: Measuring the Impact of Malicious Devices on a Cellular Network Core. *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS)*, pages 223-234, November 2009. (acceptance rate=18.4%)

William Enck, Patrick McDaniel, and Trent Jaeger. PinUP: Pinning User Files to Known Applications. *Proceedings of the 24th Annual Computer Security Applications Conference (ACSAC)*, December 2008. (acceptance rate=24.3%)

William Enck, Kevin Butler, Thomas Richardson, Patrick McDaniel, and Adam Smith. Defending Against Attacks on Main Memory Persistence. *Proceedings of the 24th Annual Computer Security Applications Conference (ACSAC)*, December 2008. (acceptance rate=24.3%)

Kevin Butler, Stephen McLaughlin, and Patrick McDaniel. Rootkit-Resistant Disks. *Proceedings of the 15th ACM Conference on Computer and Communications Security (CCS)*, November 2008. Alexandria, VA. (acceptance rate=18.1%%)

Patrick Traynor, Kevin Butler, William Enck, and Patrick McDaniel. Realizing Massive-Scale Conditional Access Systems Through Attribute-Based Cryptosystems. *ISOC Network & Distributed System Security Symposium (NDSS)*, February 2008. San Diego, CA. (acceptance rate=17.7%%)

Luke St. Clair, Joshua Schiffman, Trent Jaeger, and Patrick McDaniel. Establishing and Sustaining System Integrity via Root of Trust Installation. *23rd Annual Computer Security Applications Conference (ACSAC)*, pages 19-29, December 2007. Miami, FL. (acceptance rate=21.9%)

Boniface Hicks, Tim Misiak, and Patrick McDaniel. Channels: Runtime System Infrastructure for Security-typed Languages. *23rd Annual Computer Security Applications Conference (ACSAC)*, pages 443-452, December 2007. Miami, FL. (acceptance rate=21.9%)

Dhananjay Bapat, Kevin Butler, and Patrick McDaniel. Towards Automated Privilege Separation. *Proceedings of 2nd International Conference on Information Systems Security (short paper)*, December 2007. Delhi, India. (acceptance rate=31.5%)

Lisa Johansen, Kevin Butler, Michael Rowell, and Patrick McDaniel. Email Communities of Interest. *Fourth Conference on Email and Anti-Spam (CEAS 2007)*, August 2007. Mountain View, California. (acceptance rate=18.9%)

Patrick Traynor, Patrick McDaniel, and Thomas La Porta. On Attack Causality in Internet-Connected Cellular Networks. *Proceedings of the 16th USENIX Security Symposium*, pages 1–16, August 2007. Boston, MA. (acceptance rate=12.28%)

Boniface Hicks, Sandra Rueda, Trent Jaeger, and Patrick McDaniel. From Trusted to Secure: Building and Executing Applications that Enforce System Security. *Proceedings of the USENIX Annual Technical Conference*, June 2007. Santa Clara, CA. (acceptance rate=23.8%)

William Enck, Patrick McDaniel, Shubho Sen, Panagiotis Sebos, Sylke Spoerel, Albert Greenberg, Sanjay Rao, and William Aiello. Configuration Management at Massive Scale: System Design and Experience. *Proceedings of the USENIX Annual Technical Conference*, June 2007. Santa Clara, CA. (acceptance rate=23.8%)

Boniface Hicks, Sandra Rueda, Luke St. Clair, Trent Jaeger, and Patrick McDaniel. A Logical Specification and Analysis for SELinux MLS. *12th ACM Symposium on Access Control Models and Technologies (SACMAT)*, ACM, June 2007. Sophia Antipolis, France. (acceptance rate=23.75%)

Anusha Sriraman, Kevin Butler, Patrick McDaniel, and Padma Raghavan. Analysis of IPv4 Address Space Delegation Structure. *12th IEEE Symposium on Computers and Communications (ISCC)*, July 2007. Aveiro, Portugal. (acceptance rate=40%)

Heesook Choi, Thomas La Porta, and Patrick McDaniel. Privacy Preserving Communication in MANETs. *Proceedings of Fourth Annual IEEE Communications Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks (SECON 07)*, June 2007. San Diego, CA. (acceptance rate=20%)

Sophie Qiu, Patrick McDaniel, and Fabian Monrose. Toward Valley-Free Inter-domain Routing. *Proceedings of 2007 IEEE International Conference on Communications (ICC 2007)*, June 2007. Glasgow, Scotland.

Sunam Ryu, Kevin Butler, Patrick Traynor, and Patrick McDaniel. Leveraging Identity-based Cryptography for Node ID Assignment in Structured P2P Systems. *Proceedings of the 3rd IEEE International Symposium on Security in Networks and Distributed Systems (SSNDS-07)*, June 2007. Niagra Falls, Canada. (acceptance rate=40%)

Hosam Rowaihy, William Enck, Patrick McDaniel, and Thomas La Porta. Limiting Sybil Attacks in Structured Peer-to-Peer Networks. *Proceedings of IEEE INFOCOM 2007 MiniSymposium*, May 2007. Anchorage, AK. (acceptance rate=25%)

Boniface Hicks, Kiyan Ahmadizadeh, and Patrick McDaniel. Understanding Practical Application Development in Security-Typed Languages. *22st Annual Computer Security Applications Conference (ACSAC)*, pages 153–164, December 2006. Miami, FL. (**best student paper**). (acceptance rate=30.3%)

Luke St. Clair, Lisa Johansen, William Enck, Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Trent Jaeger. Password Exhaustion: Predicting the End of Password Usefulness. *Proceedings of 2nd International Conference on Information Systems Security (ICISS)*, pages 37–55, December 2006. Kolkata, India.

Kevin Butler, William Enck, Jennifer Plasterr, Patrick Traynor, and Patrick McDaniel. Privacy-Preserving Web-Based Email. *Proceedings of 2nd International Conference on Information Systems Security (ICISS)*, pages 116–131, December 2006. Kolkata, India. (acceptance rate=30.4%)

Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Brent Waters. Secure Attribute-Based Systems. *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS)*, pages 99–112, November 2006. Alexandria, VA. (acceptance rate=14.7%)

Kevin Butler, William Aiello, and Patrick McDaniel. Optimizing BGP Security by Exploiting Path Stability. *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS)*, pages 298–310, November 2006. Alexandria, VA. (acceptance rate=14.7%)

Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta. Mitigating Attacks on Open Functionality in SMS-Capable Cellular Networks. *Proceedings of the Twelfth Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 182–193, September 2006. Los Angeles, CA. (acceptance rate=11.7%)

Patrick Traynor, Michael Chien, Scott Weaver, Boniface Hicks, and Patrick McDaniel. Non-Invasive Methods for Host Certification. *Proceedings of the Second IEEE Communications Society/CreateNet International Conference on Security and Privacy in Communication Networks (SecureComm)*, August 2006. Baltimore, MD. (acceptance rate=25.4%)

Sophie Qiu, Patrick McDaniel, Fabian Monrose, and Avi Rubin. Characterizing Address Use Structure and Stability of Origin Advertisement in Interdomain Routing. *11th IEEE Symposium on Computers and Communications*, pages 489–496, June 2006. Pula-Cagliari, Sardinia, Italy. (acceptance rate=49.4%)

Patrick McDaniel, Shubho Sen, Oliver Spatscheck, Jacobus Van der Merwe, William Aiello, and Charles Kalmanek. Enterprise Security: A Community of Interest Based Approach. *Proceedings of Network and Distributed Systems Security 2006 (NDSS)*, February 2006. San Diego, CA. (acceptance rate=13.6%)

Kevin Butler and Patrick McDaniel. Understanding Mutable Internet Pathogens, or How I Learned to Stop Worrying and Love Parasitic Behavior. *Proceedings of 1st International Conference on Information Systems Security (ICISS)*, Springer-Verlag Lecture Notes in Computer Science, volume 3803, pages 36–48, December 2005. Kolkata, India. *Invited Paper*.

Wesam Lootah, William Enck, and Patrick McDaniel. TARP: Ticket-Based Address Resolution Protocol. *21st Annual Computer Security Applications Conference (ACSAC)*, pages 95–103, December 2005. Tuscon, AZ. (acceptance rate=19.2%)

William Enck, Patrick Traynor, Patrick McDaniel, and Thomas La Porta. Exploiting Open Functionality in SMS-Capable Cellular Networks. *Proceedings of the 12th ACM Conference on Computer and Communications Security (CCS)*, pages 393–404, November 2005. Alexandria, VA. (acceptance rate=15.0%)

Matthew Pirretti, Sencun Zhu, Vijaykrishnan Narayanan, Patrick McDaniel, Mahmut Kandemir, and Richard Brooks. The Sleep Deprivation Attack in Sensor Networks: Analysis and Methods of Defense. *Proceedings of the Innovations and Commercial Applications of Distributed Sensor Networks Symposia*, October 2005. Bethesda, Maryland. (**best paper**). (acceptance rate=27.2%)

Louis Kruger, Somesh Jha, and Patrick McDaniel. Privacy Preserving Clustering. *10th European Symposium on Research in Computer Security (ESORICS '05)*, September 2005. Milan, Italy. (acceptance rate=16.4%)

Heesook Choi, William Enck, Jaesheung Shin, Patrick McDaniel, and Thomas La Porta. Secure Reporting of Traffic Forwarding Activity in Mobile Ad Hoc Networks. *MobiQuitous 2005*, July 2005. San Diego, CA. (acceptance rate=35%)

Simon Byers, Lorrie Cranor, Eric Cronin, Dave Kormann, and Patrick McDaniel. Exposing Digital Content Piracy: Approaches, Issues and Experiences. *Thirty-Eighth Conference on Signals, Systems, and Computers*, pages 45–50, Nov 2004. Monterey, CA. *Invited paper*.

William Aiello, John Ioannidis, and Patrick McDaniel. Origin Authentication in Interdomain Routing. *Proceedings of 10th ACM Conference on Computer and Communications Security (CCS)*, ACM, pages 165–178, October 2003. Washington, DC. (acceptance rate=13.8%)

Eric Cronin, Sugih Jamin, Tal Malkin, and Patrick McDaniel. On the Performance, Feasibility, and Use of Forward Secure Signatures. *Proceedings of 10th ACM Conference on Computer and Communications Security (CCS)*, ACM, pages 131-144, October 2003. Washington, DC. (acceptance rate=13.8%)

Patrick McDaniel. On Context in Authorization Policy. *8th ACM Symposium on Access Control Models and Technologies (SACMAT)*, ACM, pages 80-89, June 2003. Como, Italy. (acceptance rate=37.0%)

Geoff Goodell, William Aiello, Tim Griffin, John Ioannidis, Patrick McDaniel, and Avi Rubin. Working Around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing. *Proceedings of Network and Distributed Systems Security 2003 (NDSS)*, Internet Society, pages 75-85, February 2003. San Diego, CA. (acceptance rate=20.5%)

Patrick McDaniel and Atul Prakash. Methods and Limitations of Security Policy Reconciliation. *2002 IEEE Symposium on Security and Privacy*, IEEE Computer Society Press, pages 73-87, May 2002. Oakland, CA. (acceptance rate=17.8%)

Patrick McDaniel, Atul Prakash, Jim Irrer, Sharad Mittal, and Thai-Chuin Thuang. Flexibly Constructing Secure Groups in Antigone 2.0. *Proceedings of DARPA Information Survivability Conference and Exposition II*, IEEE Computer Society Press, pages 55-67, June 2001. Los Angeles, CA.

Hugh Harney, Andrea Colegrove, and Patrick McDaniel. Principles of Policy in Secure Groups. *Proceedings of Network and Distributed Systems Security 2001 (NDSS)*, Internet Society, pages 125-135, February 2001. San Diego, CA. (acceptance rate=24.2%)

Patrick McDaniel and Sugih Jamin. Windowed Certificate Revocation. *Proceedings of IEEE INFOCOM 2000*, IEEE, pages 1406-1414, March 2000. Tel Aviv, Israel. (acceptance rate=26%)

Patrick McDaniel and Avi Rubin. A Response to ‘Can We Eliminate Certificate Revocation Lists?’. *Proceedings of Financial Cryptography 2000*, International Financial Cryptography Association (IFCA), February 2000. Anguilla, British West Indies. (acceptance rate=26.1%)

Andrwe Adamson, C.J. Antonelli, Kevin Coffman, Patrick McDaniel, and Jim Rees. Secure Distributed Virtual Conferencing. *Proceedings of Communications and Multimedia Security (CMS '99)*, pages 176-190, September 1999. Katholieke Universiteit, Leuven, Belgium. (acceptance rate=70%)

Patrick McDaniel, Atul Prakash, and Peter Honeyman. Antigone: A Flexible Framework for Secure Group Communication. *Proceedings of the 8th USENIX Security Symposium*, pages 99-114, August 1999. Washington, DC. (acceptance rate=26.8%)

Workshop Publications

Z. Berkay Celik and Patrick McDaniel. Extending Detection with Privileged Information via Generalized Distillation. *IEEE Security & Privacy Workshop on Deep Learning and Security (IEEE S&P DLS)* 2018.

Alexander Alexeev, Diane Henshel, Karl Levitt, Patrick McDaniel, Brian Rivera, Steve Templeton, and Michael Weisman. Constructing a Science of Cyber-Resilience for Military Systems. *Information Systems and Technology (IST) Panel, IST-153/RWS-21, CEUR Workshop Proceedings*, pages 30-42, October 2017.

Berkay Celik, Patrick McDaniel, and Rauf Izmailov. Feature Cultivation in Privileged Information-augmented Detection. *Proceedings of the International Workshop on Security And Privacy Analytics (IWSPA 2017)* 2017. *Invited Paper*.

Stefan Achleitner, Thomas La Porta, Patrick McDaniel, Shridatt Sugrim, Srikanth Krishnamurthy, and Ritu Chadha. Cyber Deception: Virtual Networks to Defend Insider Reconnaissance. *Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats*, ACM 2016.

Patrick McDaniel, Trent Jaeger, Thomas La Porta, Nicolas Papernot, Robert Walls, Alexander Kott, Lisa Marvel, Ananthram Swami, Prasant Mohapatra, Srikanth V. Krishnamurthy, and Iulian Neamtiu. Security and Science of Agility. *First ACM Workshop on Moving Target Defense (MTD 2014)*, November 2014. Scottsdale, AZ.

Joshua Schiffman, Thomas Moyer, Hayawardh Vijayakumar, Trent Jaeger, and Patrick McDaniel. Seeding Clouds with Trust Anchors. *Proceedings of CCSW 2010: The ACM Cloud Computing Security Workshop*, October 2010. Chicago, IL.

Stephen McLaughlin, Dmitry Podkuiko, Adam Delozier, Sergei Miadzvezhanka, and Patrick McDaniel. Embedded Firmware Diversity for Smart Electric Meters. *Proceedings of the 5th Workshop on Hot Topics in Security (HotSec '10)*, August 2010. Washington, DC. (acceptance rate=19.6%)

Patrick McDaniel, Kevin Butler, Stephen McLaughlin, Radu Sion, Erez Zadok, and Marianne Winslett. Towards a Secure and Efficient System for End-to-End Provenance. *the 2nd USENIX Workshop on the Theory and Practice of Provenance*, February 2010. San Jose, CA.

Thomas La Porta, Patrick McDaniel, Karl Rauscher, and Jun Shu. The Impact of Supply Chain on Information and Communications Technology Security. *In the 1st Workshop on Workshop on Telecommunications Infrastructure Protection and Security*, December 2009. Honolulu, Hawaii.

Stephen McLaughlin, Dmitry Podkuiko, and Patrick McDaniel. Energy Theft in the Advanced Metering Infrastructure. *In the 4th International Workshop on Critical Information Infrastructure Security*, September 2009. Bonn, Germany.

Matt Blaze and Patrick McDaniel. Below the Salt: The Dangers of Unfulfilled Physical Media Assumptions. *In Proceedings of Seventeenth International Workshop on Security Protocols*, April 2009. Cambridge, England.

Kevin Butler, William Enck, Harri Hursti, Stephen McLaughlin, Patrick Traynor, and Patrick McDaniel. Systemic Issues in the Hart InterCivic and Premier Voting Systems: Reflections Following Project EVEREST. *In Proceedings of the 3rd USENIX/ACCURATE Electronic Voting Technology Workshop (EVT '08)*, July 2008.

Kevin Butler, Stephen McLaughlin, and Patrick McDaniel. Non-Volatile Memory and Disks: Avenues for Policy Architectures. *Proceedings of the 1st ACM Computer Security Architectures Workshop*, November 2007. Alexandria, VA. (acceptance rate=30%)

William Enck, Sandra Rueda, Yogesh Sreenivasan, Joshua Schiffman, Luke St. Clair, Trent Jaeger, and Patrick McDaniel. Protecting Users from "Themselves". *Proceedings of the 1st ACM Computer Security Architectures Workshop*, November 2007. Alexandria, VA. (acceptance rate=30%)

Boniface Hicks, David King, and Patrick McDaniel. Jifclipse: Development Tools for Security-Typed Applications. *Proceedings of the 2nd ACM SIGPLAN Workshop on Programming Languages and Analysis for Security (PLAS '07)*, ACM Press, June 2007. San Diego, CA. (acceptance rate=39%)

Boniface Hicks, Sandra Rueda, Trent Jaeger, and Patrick McDaniel. Integration of SELinux and Security-typed Languages. *Proceedings of the 2007 Security-Enhanced Linux Workshop*, March 2007. Baltimore, MD. (acceptance rate=75%)

Sophie Qiu, Fabian Monrose, Andreas Terzis, and Patrick McDaniel. Efficient Techniques for Detecting False Origin Advertisements in Inter-domain Routing. *Proceedings of The Second Workshop on Secure Network Protocols (NPSec)*, November 2006. Santa Barbara. (acceptance rate=40%)

Shiva Chaitanya, Kevin Butler, Patrick McDaniel, and Anand Sivasubramaniam. Design, Implementation and Evaluation of Security in iSCSI-based Network Storage Systems. *Proceedings of 2nd International Workshop on Storage Security and Survivability (StorageSS 2006)*, October 2006. Alexandria, Virginia. (acceptance rate=68.7%)

Trent Jaeger, Patrick McDaniel, Luke St. Clair, Ramon Caceres, and Reiner Sailer. Shame on Trust in Distributed Systems. *Proceedings of the First Workshop on Hot Topics in Security (HotSec '06)*, July 2006. Vancouver, B.C., Canada. (acceptance rate=19.6%)

Kevin Butler, Patrick McDaniel, and Sophie Qiu. BGPRV: A Library for Fast and Efficient Routing Data Manipulation. *Proceedings of DETER Community Workshop*, June 2006. Arlington, VA.

Kevin Butler and Patrick McDaniel. Testing Large Scale BGP Security in Replayable Network Environments. *Proceedings of DETER Community Workshop*, June 2006. Arlington, VA.

Boniface Hicks, David King, Patrick McDaniel, and Michael Hicks. Trusted Declassification: High-level Policy for a Security-Typed Language. *Proceedings of ACM SIGPLAN Workshop on Programming Languages and Analysis for Security*, pages 65-74, June 2006. Ottawa, Canada. (acceptance rate=58.8%)

Ali Al-Lawati, Dongwon Lee, and Patrick McDaniel. Blocking in Private Information Matching. *Proceedings of Second International ACM SIGMOD Workshop on Information Quality in Information Systems*, June 2005. Baltimore, MD. (acceptance rate=42.3%)

William Aiello, Charles Kalmanek, Patrick McDaniel, Shubho Sen, Oliver Spatscheck, and Jacobus Van der Merwe. Analysis of Communities Of Interest in Data Networks. *Passive and Active Measurement Workshop 2005*, March 2005. Boston, MA. (acceptance rate=28.5%)

Simon Byers, Lorrie Cranor, David Kormann, and Patrick McDaniel. Searching for Privacy: Design and Implementation of a P3P-Enabled Search Engine. *Proceedings of 2004 Workshop on Privacy Enhancing Technologies*

(PETS), May 2004. Toronto, Canada. (acceptance rate=28.7%)

Hao Wang, Somesh Jha, Patrick McDaniel, and Miron Livny. Security Policy Reconciliation in Distributed Computing Environments. *Proceedings of 5th International Workshop on Policies for Distributed Systems and Networks (Policy 2004)*, IEEE Computer Society Press, pages 137-146, June 2004. Yorktown Heights, NY. (acceptance rate=20.6%)

Simon Byers, Lorrie Cranor, Eric Cronin, Dave Kormann, and Patrick McDaniel. Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *Proceedings of 2003 ACM Workshop on Digital Rights Management*, ACM, October 2003. Washington, DC, also appeared in Telecommunications Policy Research Conference – September 2003. (acceptance rate=40.0%)

Patents

Patrick McDaniel and Martin Strauss, End-to-end secure wireless communication for requesting a more secure channel. U.S. Patent7873350, January. 2011.

Oliver Spatscheck, Subhabrata Sen, Jacobus Van der Merwe, and Patrick McDaniel, Automated disambiguation of fixed-serverport-based applications from ephemeral applications. U.S. Patent7875044, July. 2011.

Patrick McDaniel and Martin Strauss, End-to-end secure wireless communication for requesting a more secure channel. U.S. Patent8175580, May. 2012.

William Aiello, Charles Kalmanek Jr, William Leighton III, Patrick McDaniel, Subhabrata Sen, Oliver Spatscheck, and Jacobus Van der Merwe, Reverse firewall with self-provisioning. U.S. Patent8453227, May. 2013.

Patrick McDaniel, Subhabrata Sen, Oliver Spatscheck, and Jacobus Van der Merwe, System and method for tracking individuals on a data network using communities of interest. U.S. Patent8732293, May. 2014.

William Aiello, Charles Kalmanek Jr, William Leighton III, Patrick McDaniel, Subhabrata Sen, Oliver Spatscheck, and Jacobus Van der Merwe, Reverse firewall with self-provisioning. U.S. Patent8813213, August. 2014.

Other Publications

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow, Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples. *arXiv preprint arXiv:1605.07277*, 2016.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Berkay Celik, and Ananthram Swami, Practical Black-Box Attacks against Deep Learning Systems using Adversarial Examples. *arXiv preprint arXiv:1602.02697*, 2016.

Patrick McDaniel and Avi Rubin. Conference Proceedings. *2008 IEEE Symposium on Security and Privacy*, IEEE. May 2008.

Patrick McDaniel, Kevin Butler, William Enck, Harri Hursti, Stephen McLaughlin, Patrick Traynor, Matt Blaze, Adam Aviv, Pavol Cerny, Sandy Clark, Eric Cronin, Gaurav Shah, Micah Sherr, Giovanni Vigna, Richard Kemmerer, David Balzarotti, Greg Banks, Marco Cova, Viktoria Felmetsger, William Robertson, Fredrik Valeur, Joseph Lorenzo Hall, and Laura Quilter, EVEREST: Evaluation and Validation of Election-Related Equipment, Standards and Testing. Public Report, Ohio Secretary of State 2007.

Patrick McDaniel and Shyam K. Gupta. Conference Proceedings. *The Third International Conference Information Systems Security*, Springer. December 2007.

Birgit Pfitzmann and Patrick McDaniel. Conference Proceedings. *2007 IEEE Symposium on Security and Privacy*, IEEE. May 2007.

Patrick McDaniel. Conference Proceedings. *The 14th USENIX Security Symposium*, USENIX. August 2005.

Patrick McDaniel, Policy Evolution: Autonomic Environmental Security. Software Engineering Research Center Showcase, USA, December 2004.

Hugh Harney, Uri Meth, Andrea Colegrove, Angela Schuett, Patrick McDaniel, Gavin Kenny, Haitham Cruickshank, and Sunil Iyengar, GSAKMP (*Draft*). Internet Research Task Force, August 2003.

Patrick McDaniel and Atul Prakash. A Flexible Architecture for Security Policy Enforcement. *Proceedings of DARPA Information Survivability Conference and Exposition III, Research Summaries*, pages 234-239, April 2003.

Jim Irrer, Atul Prakash, and Patrick McDaniel. Antigone: Policy-Based Secure Group Communications Systems and AMirD: Antigone-Based Secure File Mirroring System. *Proceedings of DARPA Information Survivability*

Conference and Exposition III, Demo Summaries, pages 44-46, April 2003.

Patrick McDaniel and Sugih Jamin, Windowed Key Revocation in Public Key Infrastructures. *NASA Tech Briefs*, pages 55, August, 2002.

Patrick McDaniel and Atul Prakash, Antigone Secure Group Communication System. *NASA Tech Briefs*, 2001.

Patrick McDaniel, *Policy Management in Secure Group Communication*. PhD Thesis, *University of Michigan, Ann Arbor, MI*, August 2001.

Patrick McDaniel, Hugh Harney, Andrea Colegrove, Atul Prakash, and P. Dinsmore, Multicast Security Policy Requirements and Building Blocks (*Draft*). Internet Research Task Force, Secure Multicast Research Group (SMuG), November 2000.

Tom Hardjono, Hugh Harney, Patrick McDaniel, Andrea Colegrove, and Peter Dinsmore, Group Security Policy Token (*Draft*). Internet Research Task Force, Secure Multicast Research Group (SMuG), November 2001.

Patrick McDaniel, Hugh Harney, Peter Dinsmore, and Atul Prakash, Multicast Security Policy (*Draft*). Internet Research Task Force, Secure Multicast Research Group (SMuG), June 2000.

Patrick McDaniel, 8th USENIX Security Symposium Conference Summaries, Potpourri Session. *USENIX Login Magazine*, pages 9-12, November, 1999.

Patrick McDaniel, *The Analysis of D_i , a Detailed Design Metric on Large Scale Software*. Masters Thesis, *Ball State University, Muncie, IN*, June 1991.

Technical Reports

Nicolas Papernot, Ian Goodfellow, Ryan Sheatsley, Reuben Feinman, and Patrick McDaniel, cleverhans v1.0.0: an adversarial machine learning library. Technical Report arXiv:1610.00768, arXiv preprint 2016.

Devin J. Pohly and Patrick McDaniel, Modeling Privacy and Tradeoffs in Multichannel Secret Sharing Protocols. Technical Report NAS-TR-0188-2016, Institute of Networking and Security Research, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, January 2016.

Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami, The Limitations of Deep Learning in Adversarial Settings. Technical Report NAS-TR-0172-2014, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, October 2015.

Wenhui Hu, Damien Ocateau, Patrick McDaniel, and Peng Liu, Duet: Library Integrity Verification for Android Applications. Technical Report NAS-TR-0172-2014, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, February 2014.

Devin J. Pohly, Stephen McLaughlin, Patrick McDaniel, and Kevin Butler, Hi-Fi: Collecting High-Fidelity Whole-System Provenance. Technical Report NAS-TR-0160-2012, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2012.

Damien Ocateau, Somesh Jha, and Patrick McDaniel, Retargeting Android Applications to Java Bytecode. Technical Report NAS-TR-0150-2011, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, September 2011.

Stephen McLaughlin and Patrick McDaniel, Protecting Consumer Privacy from Electric Load Monitoring. Technical Report NAS-TR-0147-2011, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, March 2011.

William Enck and Patrick McDaniel, Federated Information Flow Control for Mobile Phones. Technical Report NAS-TR-0136-2010, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, July 2010.

Kevin Butler, Stephen McLaughlin, and Patrick McDaniel, Kells: A Protection Framework for Portable Data. Technical Report NAS-TR-0134-2010, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2010.

Stephen McLaughlin, Dmitry Podkuiko, Adam Delozier, Sergei Miadzevzhanka, and Patrick McDaniel, Multi-vendor Penetration Testing in the Advanced Metering Infrastructure. Technical Report NAS-TR-0133-2010, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2010.

Machigar Ongtang, Kevin Butler, and Patrick McDaniel, Porscha: Policy Oriented Secure Content Handling in Android . Technical Report NAS-TR-0132-2010, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2010.

Joshua Schiffman, Thomas Moyer, Hayawardh Vijayakumar, Trent Jaeger, and Patrick McDaniel, Seeding Clouds with Trust Anchors. Technical Report NAS-TR-0127-2010, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, April 2010.

William Enck, Machigar Ongtang, and Patrick McDaniel, TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones. Technical Report NAS-TR-0120-2010, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, February 2010. Updated 13 July 2010.

Joshua Schiffman, Thomas Moyer, Christopher Shal, Trent Jaeger, and Patrick McDaniel, Justifying Integrity Using a Virtual Machine Verifier. Technical Report NAS-TR-0119-2009, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, August 2009.

Machigar Ongtang, Stephen McLaughlin, William Enck, and Patrick McDaniel, Semantically Rich Application-Centric Security in Android. Technical Report NAS-TR-0116-2009, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2009.

Stephen McLaughlin, Dmitry Podkuiko, and Patrick McDaniel, Energy Theft in the Advanced Metering Infrastructure. Technical Report NAS-TR-0115-2009, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2009.

William Enck, Machigar Ongtang, and Patrick McDaniel, On Lightweight Mobile Phone App Certification. Technical Report NAS-TR-0113-2009, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, April 2009.

Patrick Traynor, Michael Lin, Machigar Ongtang, Vikhyath Rao, Thomas La Porta, and Patrick McDaniel, On Cellular Botnets: Measuring the Impact of Malicious Devices on the Network Core. Technical Report NAS-TR-0110-2009, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, March 2009.

Boniface Hicks, Sandra Rueda, Yogesh Sreenivasan, Guruprasad Jakka, David King, Trent Jaeger, and Patrick McDaniel, An Architecture for Enforcing End-to-End Security Over Web Applications. Technical Report NAS-TR-0104-2009, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, January 2009.

Joshua Schiffman, Thomas Moyer, Christopher Shal, Trent Jaeger, and Patrick McDaniel, No Node Is an Island: Shamon Integrity Monitoring Approach. Technical Report NAS-TR-0103-2009, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, February 2009.

Kevin Butler, Stephen McLaughlin, Thomas Moyer, Patrick McDaniel, and Trent Jaeger, SwitchBlade: Policy-Driven Disk Segmentation. Technical Report NAS-TR-0098-2008, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, November 2008.

Thomas Moyer, Kevin Butler, Joshua Schiffman, Patrick McDaniel, and Trent Jaeger, Scalable Asynchronous Web Content Attestation. Technical Report NAS-TR-0095-2008, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, September 2008.

Boniface Hicks, Sandra Rueda, Luke St. Clair, Trent Jaeger, and Patrick McDaniel, A Logical Specification and Analysis for SELinux MLS Policy. Technical Report NAS-TR-0091-2008, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, July 2008.

Kevin Butler, Stephen McLaughlin, and Patrick McDaniel, Rootkit-Resistant Disks. Technical Report NAS-TR-0089-2008, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, April 2008.

Patrick Traynor, Joshua Schiffman, Thomas La Porta, Patrick McDaniel, Abhrajit Ghosh, and Farooq Anjum, Constructing Secure Localization Systems with Adjustable Granularity. Technical Report NAS-TR-0084-2007,

Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, December 2007.

Stephen McLaughlin, Kevin Butler, William Enck, and Patrick McDaniel, Genbd - A Generic Block Device. Technical Report NAS-TR-0082-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, November 2007.

Kevin Butler, Stephen McLaughlin, and Patrick McDaniel, Non-Volatile Memory and Disks: Avenues for Policy Architectures. Technical Report NAS-TR-0074-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2007.

William Enck, Sandra Rueda, Joshua Schiffman, Yogesh Sreenivasan, Luke St. Clair, Trent Jaeger, and Patrick McDaniel, Protecting Users From “Themselves”. Technical Report NAS-TR-0073-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2007.

Dhananjay Bapat, Kevin Butler, and Patrick McDaniel, Towards Automated Privilege Separation. Technical Report NAS-TR-0071-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, May 2007.

Patrick Traynor, Kevin Butler, William Enck, and Patrick McDaniel, Realizing Massive-Scale Conditional Access Systems Through Attribute-Based Cryptosystems. Technical Report NAS-TR-0070-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, May 2007.

Luke St. Clair, Joshua Schiffman, Trent Jaeger, and Patrick McDaniel, Establishing and Sustaining System Integrity via Root of Trust Installation. Technical Report NAS-TR-0067-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, April 2007.

Boniface Hicks, David King, and Patrick McDaniel, Jifclipse: Development Tools for Security-Typed Language. Technical Report NAS-TR-0065-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, April 2007.

William Enck, Patrick McDaniel, and Trent Jaeger, Protecting User Files by Reducing Application Access. Technical Report NAS-TR-0063-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, February 2007.

Boniface Hicks, Sandra Rueda, Trent Jaeger, and Patrick McDaniel, A Logical Specification and Analysis for SELinux MLS Policy. Technical Report NAS-TR-0058-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, January 2007.

Boniface Hicks, Sandra Rueda, Trent Jaeger, and Patrick McDaniel, From Trusted to Secure: Building and Executing Applications that Enforce System Security. Technical Report NAS-TR-0061-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, January 2007.

Lisa Johansen, Kevin Butler, William Enck, Patrick Traynor, and Patrick McDaniel, Grains of SANs: Building Storage Area Networks from Memory Spots. Technical Report NAS-TR-0060-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, January 2007.

Patrick Traynor, Vikhyath Rao, Trent Jaeger, Patrick McDaniel, and Thomas La Porta, From Mobile Phones to Responsible Devices. Technical Report NAS-TR-0059-2007, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, January 2007.

Anusha Sriraman, Kevin Butler, Patrick McDaniel, and Padma Raghavan, Analysis of the IPv4 Address Space Delegation Structure. Technical Report NAS-TR-0057-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, December 2006.

Luke St. Clair, Joshua Schiffman, Trent Jaeger, and Patrick McDaniel, Sum of the Parts: Composing Trust from Validation Primitives. Technical Report NAS-TR-0056-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, November 2006.

Sophie Qiu, Patrick McDaniel, and Fabian Monrose, Toward Valley-Free Inter-domain Routing. Technical Report NAS-TR-0054-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, October 2006.

Boniface Hicks, Sandra Rueda, Trent Jaeger, and Patrick McDaniel, Integrating SELinux with Security-typed Languages. Technical Report NAS-TR-0052-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, October 2006.

Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta, Mitigating Attacks on Open Functionality in SMS-Capable Networks. Technical Report NAS-TR-0051-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, October 2006.

Patrick Traynor, Kevin Butler, William Enck, Kevin Borders, and Patrick McDaniel, *malnets: Large-Scale Malicious Networks via Compromised Wireless Access Points*. Technical Report NAS-TR-0048-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, September 2006.

Boniface Hicks, Sandra Rueda, Trent Jaeger, and Patrick McDaniel, Breaking Down the Walls of Mutual Distrust: Security-typed Email Using Labeled IPsec. Technical Report NAS-TR-0049-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, September 2006.

Sunam Ryu, Kevin Butler, Patrick Traynor, and Patrick McDaniel, Leveraging Identity-based Cryptography for Node ID Assignment in Structured P2P Systems. Technical Report NAS-TR-0043-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, August 2006.

Wesam Lootah, William Enck, and Patrick McDaniel, TARP: Ticket-based Address Resolution Protocol (*extended version*). Technical Report NAS-TR-0046-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, August 2006. (extends lem05)

Patrick McDaniel, Understanding Equivalence in High-Level and Information Flow Policy. Technical Report NAS-TR-0042-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, July 2006.

Trent Jaeger, Patrick McDaniel, Luke St. Clair, Ramon Caceres, and Reiner Sailer, Shame on Trust in Distributed Systems. Technical Report RC239664 (W0605-129), IBM Research Division, Yorktown Heights, NY, May 2006.

Lisa Johansen, Kevin Butler, Michael Rowell, and Patrick McDaniel, Email Communities of Interest. Technical Report NAS-TR-0040-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, May 2006.

Boniface Hicks, Kiyan Ahmadizadeh, and Patrick McDaniel, From Languages to Systems: Understanding Practical Application Development in Security-typed Languages. Technical Report NAS-TR-0035-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, April 2006.

Boniface Hicks, David King, Patrick McDaniel, and Michael Hicks, Trusted Declassification: High-level policy for a security-typed language. Technical Report NAS-TR-0033-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, March 2006.

Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Brent Waters, Secure Attribute-Based Systems. Technical Report NAS-TR-0028-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, February 2006.

William Enck, Kevin Butler, Thomas Richardson, and Patrick McDaniel, Securing Non-Volatile Main Memory. Technical Report NAS-TR-0029-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, February 2006.

Luke St. Clair, Lisa Johansen, William Enck, Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Trent Jaeger, Password Exhaustion: Predicting the End of Password Usefulness. Technical Report NAS-TR-0030-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, February 2006.

Heesook Choi, Patrick McDaniel, and Thomas La Porta, Privacy Preserving Communication in MANETs. Technical Report NAS-TR-0031-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, December 2005.

Matthew Pirretti, Vijaykrishnan Narayanan, Patrick McDaniel, and Bharat Madan, SLAT: Secure Localization

with Attack Tolerance. Technical Report NAS-TR-0024-2005, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, August 2005.

Patrick Traynor, Michael Chien, Scott Weaver, Boniface Hicks, and Patrick McDaniel, Non-Invasive Methods for Host Certification. Technical Report NAS-TR-0025-2005, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, September 2005.

Sophie Qiu, Patrick McDaniel, Fabian Monrose, and Avi Rubin, Characterizing Address Use Structure and Stability of Origin Advertisement in Interdomain Routing. Technical Report NAS-TR-0018-2005, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, July 2005.

Hosam Rowaihy, William Enck, Patrick McDaniel, and Thomas La Porta, Limiting Sybil Attacks in Structured Peer-to-Peer Networks. Technical Report NAS-TR-0017-2005, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, July 2005.

Boniface Hicks, Patrick McDaniel, and Ali Hurson, Information Flow Control in Database Security: A Case Study for Secure Programming with Jif. Technical Report NAS-TR-0011-2005, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, April 2005.

Wesam Lootah, William Enck, and Patrick McDaniel, TARP: Ticket-Based Address Resolution Protocol. Technical Report NAS-TR-0010-2005, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2005.

Patrick Traynor, Kevin Butler, William Enck, Jennifer Plaster, Scott Weaver, John van Bramer, and Patrick McDaniel, Privacy-Preserving Web-Based Email. Technical Report NAS-TR-0009-2005, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, June 2005.

William Enck, Patrick Traynor, Patrick McDaniel, and Thomas La Porta, Exploiting Open Functionality in SMS-Capable Cellular Networks. Technical Report NAS-TR-0007-2005, Network and Security Center, Department of Computer Science, Pennsylvania State University, May 2005.

Boniface Hicks, David King, and Patrick McDaniel, Declassification with Cryptographic Functions in a Security-Typed Language. Technical Report NAS-TR-0004-2005, Network and Security Center, Department of Computer Science, Pennsylvania State University, January 2005. (*updated May 2005*).

William Aiello, Kevin Butler, and Patrick McDaniel, Path Authentication in Interdomain Routing. Technical Report TR NAS-TR-0002-2004, Network and Security Center, Department of Computer Science and Engineering, Penn State University, November 2004.

Dan Pei, William Aiello, Anna Gilbert, and Patrick McDaniel, Origin Disturbances in BGP. Technical Report TD-62TJJF8, AT&T Labs - Research, Florham Park, NJ, July 2004.

Kevin Butler, Toni Farley, Patrick McDaniel, and J. Rexford, A Survey of BGP Security Issues and Solutions. Technical Report TD-5UGJ33, AT&T Labs - Research, Florham Park, NJ, February 2004. (*revised June 2004*).

Patrick McDaniel and Atul Prakash, Securing Distributed Applications Using a Policy-based Approach. Technical Report TD-5UDKVD, AT&T Labs - Research, Florham Park, NJ, December 2003.

William Aiello, Johyn Ioannidis, and Patrick McDaniel, Origin Authentication in Interdomain Routing. Technical Report TD-5QHG2G, AT&T Labs - Research, Florham Park, NJ, August 2003.

Eric Cronin, Sugih Jamin, Tal Malkin, and Patrick McDaniel, On the Performance, Feasibility, and Use of Forward Secure Signatures. Technical Report TD-5QHGBK, AT&T Labs - Research, Florham Park, NJ, August 2003.

Simon Byers, Lorrie Cranor, Eric Cronin, Dave Kormann, and Patrick McDaniel, Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. Technical Report TD-5N6SJ4, AT&T Labs - Research, Florham Park, NJ, August 2003.

Patrick McDaniel, On Context in Authorization Policy. Technical Report TD-5JCJCK, AT&T Labs - Research, Florham Park, NJ, January 2003.

Patrick McDaniel and Atul Prakash, An Architecture for Security Policy Enforcement. Technical Report TD-5C6JFV, AT&T Labs - Research, Florham Park, NJ, July 2002.

Patrick McDaniel and Atul Prakash, Methods and Limitations of Security Policy Reconciliation. Technical Report TD57-PAW, AT&T Labs - Research, Florham Park, NJ, February 2002.

Patrick McDaniel and Atul Prakash, Ismene: Provisioning and Policy Reconciliation in Secure Group Communication. Technical Report CSE-TR-438-00, Electrical Engineering and Computer Science, University of Michigan, December 2000.

Patrick McDaniel and Atul Prakash, Lightweight Failure Detection in Secure Group Communication. Technical Report CSE-TR-428-00, Electrical Engineering and Computer Science, University of Michigan, June 2000.

Patrick McDaniel and Atul Prakash, Antigone: Implementing Policy in Secure Group Communication. Technical Report CSE-TR-426-00, Electrical Engineering and Computer Science, University of Michigan, May 2000.

Patrick McDaniel and Sugih Jamin, Windowed Certificate Revocation. Technical Report CSE-TR-413-99, Electrical Engineering and Computer Science, University of Michigan, November 1999.

Patrick McDaniel, Atul Prakash, and Peter Honeyman, Antigone: A Flexible Framework for Secure Group Communication. Technical Report 99-2, Center for Information Technology Integration, September 1999.

Patrick McDaniel and Avi Rubin, A Response to "Can We Eliminate Certificate Revocation Lists?". Technical Report 99.8.1, AT&T Labs - Research, Florham Park, NJ, August 1999.

Andrew Adamson, C.J. Antonelli, Kevin Coffman, Patrick McDaniel, and Jim Rees, Secure Distributed Virtual Conferencing: Multicast or Bust. Technical Report 99-1, Center for Information Technology Integration, January 1999.

Patrick McDaniel and Sugih Jamin, Windowed Key Revocation in Public Key Infrastructures. Technical Report CSE-TR-376-98, Electrical Engineering and Computer Science, University of Michigan 1998.

Patrick McDaniel and Sugih Jamin, A Scalable Key Distribution Hierarchy. Technical Report CSE-TR-366-98, Electrical Engineering and Computer Science, University of Michigan 1998.

Patrick McDaniel, Peter Honeyman, and Atul Prakash, Lightweight Secure Group Communication. Technical Report 98-2, Center for Information Technology Integration, University of Michigan, April 1998.

Wayne Zage, Delores Zage, Patrick McDaniel, and Irshad Khan, Evaluating Design Metrics on Large-Scale Software. Technical Report SERC-TR-106-P, Software Engineering Resource Center, Purdue University, September 1991.

PUBLIC SPEAKING

Invited Talks

Distinguished Lecture: The Challenges of Machine Learning in Adversarial Settings: A Systems Perspective. *Computer Science Department, University of Wisconsin-Madison*, Madison, WI, February, 2020.

Shutterstock Distinguished Lecture: The Challenges of Machine Learning in Adversarial Settings. *Computer Science Department, Stonybrook University*, Stonybrook, NY, December, 2019.

Distinguished Blockchain Lecture: The Challenges of Machine Learning in Adversarial Settings. *Cylab Security and Privacy Institute, Carnegie Mellon University*, Pittsburgh, PA, December, 2019.

The Challenges of Machine Learning in Adversarial Settings. *S2ERC, Ball State University*, Muncie, IN, November, 2019.

Keynote: The Challenges of Machine Learning in Adversarial Settings. *Triangle Area Privacy and Security Day*, Durham, NC, October, 2019.

AI-Cybersecurity Workshop Briefing to the NITRD and MLAI Subcommittees. *NITRD and MLAI Subcommittees Quarterly Meeting*, Washington, DC, July, 2019.

Workshop on the Security and Privacy of Machine Learning. *Workshop on the Security and Privacy of Machine Learning, International Conference on Machine Learning*, Long Beach, CA, June, 2019.

Keynote: The Challenges of Machine Learning in Adversarial Settings. *2019 Subversion and Assurance of AI Workshop, US National Reconnaissance Office*, Washington, DC, March, 2019.

The Challenges of Machine Learning in Adversarial Settings. *National Science Foundation*, Alexandria, VA, March, 2019.

Convergence of AI and IoT. *Intelligence Community Studies Board, Division on Engineering & Physical Sciences, The National Academy of Sciences/Engineering*, Washington, DC, February, 2019.

Tracing the Arc of Smartphone Application Security. *Duke University*, Durham, NC, February, 2019.

Distinguished Speaker Series: The Challenges of Machine Learning in Adversarial Settings. *Department of Computer Science, University at Buffalo*, Buffalo, NY, November, 2018.

Samuel D. Conte Distinguished Lecture Series: The Challenges of Machine Learning in Adversarial Settings. *Department of Computer Science, Purdue University*, West Lafayette, Indiana, November, 2018.

The Challenges of Machine Learning in Adversarial Settings. *Computer Science Department, Indiana University of Pennsylvania*, Indiana, PA, October, 2018.

Huddle with the Faculty: The Challenges of Machine Learning in Adversarial Settings. *Penn State University Alumni Association*, University Park, PA, September, 2018.

Distinguished Lecture: The Challenges of Machine Learning in Adversarial Settings. *Department of Software and Information Systems, University of North Carolina at Charlotte*, Charlotte, NC, February, 2018.

Tracing the Arc of Smartphone Application Security. *School of Electrical and Computer Engineering, Georgia Tech University*, Atlanta, GA, December, 2017.

Tracing the Arc of Smartphone Application Security. *Department of Electrical Engineering and Computer Science, Ohio University*, Athens, OH, October, 2017.

Keynote: Attacks, Defenses, and Impacts of Machine Learning in Adversarial Settings. *2017 Conference on Security and Privacy in Communication Networks (SecureComm)*, Niagara Falls, Canada, October, 2017.

Distinguished Lecture: Attacks, Defenses, and Impacts of Machine Learning in Adversarial Settings. *Celebrating 50 Years of Computer Science @ NC State, North Carolina State University*, Raleigh, NC, October, 2017.

Distinguished Lecture: Tracing the Arc of Smartphone Application Security. *Computer Science Department and the Electrical and Computer Engineering Department Seminar Series, Colorado State University*, Fort Collins, CO, October, 2017.

Distinguished Lecture: Tracing the Arc of Smartphone Application Security. *Rochester Institute of Technology, College of Computing and Information Sciences*, Rochester, NY, September, 2017.

Distinguished Lecture: Tracing the Arc of Smartphone Application Security. *University of Texas-Dallas, Department of Computer Science*, Dallas, TX, May, 2017.

Keynote: Tracing the Arc of Smartphone Application Security. *2017 ACM on International Workshop on Security And Privacy Analytics*, Scottsdale, AZ, March, 2017.

Distinguished Lecture: Tracing the Arc of Smartphone Application Security. *The Ohio State University, Department of Computer Science and Engineering*, Columbus, OH, March, 2017.

Distinguished Lecture: Tracing the Arc of Smartphone Application Security. *University of California-Irvine, Computer Science Department*, Irvine, CA, March, 2017.

Distinguished Lecture: Tracing the Arc of Smartphone Application Security. *Virginia Technical University, Department of Computer Science*, Blacksburg, VA, March, 2017.

Keynote: Tracing the Arc of Smartphone Application Security. *12th International Conference on Information Systems Security*, Jaipur, India, December, 2016.

Tracing the Arc of Smartphone Application Security. *University of Michigan, Ann Arbor*, Ann Arbor, MI, November, 2016.

Machine Intelligence in Adversarial Settings, Developing a Normative Framework for Cyberwarfare. *United States Naval Academy*, Annapolis, MD, September, 2016.

Eight Years of Mobile Smartphone Security. *University of Pittsburgh*, Pittsburgh, PA, September, 2016.

Eight Years of Mobile Smartphone Security. *New Jersey Institute of Technology*, Newark, NJ, September, 2016.

Setting a Cyber-Security Baseline for Physical Systems: Terminology, Technologies, and Goals. *Pacific Northwest Clean Water Association*, webinar, August, 2016.

Keynote: The Limitations of Machine Learning in Adversarial Settings. *25th International Conference on Computer Communication and Networks (ICCCN 2016)*, Waikoloa, HI, August, 2016.

Keynote: Learning from Ourselves: Where are we and where can we go in mobile systems security?. *Mobile Security Technologies (MOST) 2016 Workshop, IEEE Computer Society Security and Privacy Workshops*, San Jose, CA, May, 2016.

Keynote: Eight Years of Mobile Smartphone Security. *Center for Secure and Dependable Systems (CSDS) Cybersecurity Symposium*, Coeur d'Alene, April, 2016.

Eight Years of Mobile Smartphone Security. *University of Idaho*, Moscow, ID, April, 2016.

Army Installation 2035: Cyber Challenges and Opportunities. *US Department of Defense*, Arlington, VA, April, 2016.

The Limitations of Machine Learning in Adversarial Settings. *Florida Institute on National Security Assured Autonomy Workshop*, Fort Myers, FL, February, 2016.

SABOT: Specification-based Payload Generation for Programmable Logic Controllers. *Messaging, Malware and Mobile Anti-Abuse Working Group (M3AAWG)*, San Francisco, CA, February, 2016.

Seven Years of Mobile Smartphone Security. *Computer & Information Sciences Department, Temple University*, Philadelphia, PA, January, 2016.

Seven Years of Mobile Smartphone Security. *Massachusetts Institute of Technology-Lincoln Labs*, Lexington, MA, January, 2016.

Six Years of Mobile Smartphone Security. *Information Trust Institute, University of Illinois at Urbana-Champaign*, Urbana-Champaign, IL, September, 2016.

Keynote: The Importance of Measurement and Decision Making to a Science of Security. *2015 IEEE Conference on Communications and Network Security*, Florence, Italy, September, 2015.

Keynote: The Importance of Measurement and Decision Making to a Science of Security. *3rd International Symposium on Resilient Cyber Systems*, Philadelphia, PA, August, 2015.

Distinguished Lecture Six Years of Mobile Smartphone Security. *CISPA Distinguished Lecture Series, Max Planck Institute/Saarland University*, Saarbrücken Germany, July, 2015.

Distinguished Lecture: Six Years of Mobile Smartphone Security. *Technische Universität Darmstadt*, Darmstadt Germany, July, 2015.

Estimating Attack Intent and Mission Impact From Detection Signals. *Workshop on Cyber Attack Detection, Forensics and Attribution for Assessment of Mission Impact, NATO Science and Technology Organization, Information Systems Technology Panel*, Istanbul, Turkey, June, 2015.

Keynote: The Importance of Measurement and Decision Making to a Science of Security. *2015 Symposium And Bootcamp on the Science of Security (Hotsos)*, University of Illinois at Urbana-Champaign, April, 2015.

Keynote: Security and Science of Agility. *First ACM Workshop on Moving Target Defense (MTD 2014)*, Scottsdale, AZ, November, 2014.

Evaluating Mobile Smartphone Security: The First Five Years. *Computer Science Colloquium Series, Harvard School of Engineering and Applied Sciences, Harvard University*, Boston, MA, October, 2013.

Keynote: A Secondary Internet Revolution: How the Smart Device has Changed the Information Security Landscape. *IEEE New Technology Industry Seminar (NTIS '13)*, Everett, WA, August, 2013.

Geotargeting: Mobile Device Privacy and Security. *National Academy of Sciences*, Washington DC, February, 2013.

Authentication and Web Security. *Security and Privacy in IT-EMTM 604 Guest Lecture, University of Pennsylvania*, Philadelphia, PA, February, 2013.

The Realities of Voting: A Retrospective of Ten Years of Information Security and Electronic Voting Systems. *2012 Information Assurance Day, Computer Science Department, Indiana University of Pennsylvania*, Indiana, PA, November, 2012.

Keynote: Permission-based Application Governance; A Step Forward or Backward?. *26th Annual WG 11.3 Conference on Data and Applications Security and Privacy (DBSec'12)*, Paris, France, July, 2012.

Evaluating Mobile Smartphone Security: The First Four Years. *Carnegie Mellon University*, Pittsburgh, PA, April, 2012.

Keynote: Scalable Integrity-Guaranteed AJAX. *The 14th Asia-Pacific Web Conference (APWeb)*, Kunming, China, April, 2012.

Evaluating Mobile Smartphone Application Security. *Singapore Management University*, Singapore, September, 2011.

Evaluating Mobile Smartphone Application Security. *Computer Security Foundations Symposium*, Florham Park, NJ, July, 2011.

Opening Address: Security Challenges and Solutions in Mobile Smartphone Applications. *Computer Security Foundations Symposium*, Domaine de l'Abbaye des Vaux de Cernay, France, June, 2011.

Distinguished Speaker: Security Challenges and Solutions in Mobile Smartphone Applications. *Computer and Information Science Department, University of Oregon*, Eugene, OR, April, 2011.

Identifying (and Addressing) Security and Privacy Issues in Smart Electric Meters. *Center for Non-Linear Studies*, Los Alamos, NM, February, 2011.

Distinguished Lecture: Security Challenges and Solutions in Mobile Smartphone Applications. *Department of Software Information Systems College of Computing and Informatics, UNC Charlotte*, Charlotte, NC, December, 2010.

Security Challenges and Solutions in Mobile Smartphone Applications. *Computer Science Department, Indiana University of Pennsylvania*, Indiana, PA, December, 2010.

Security Challenges and Solutions in Mobile Smartphone Applications. *Computer Science Department, Georgetown University*, Washington D.C., November, 2010.

Security Challenges and Solutions in Mobile Smartphone Applications. *Networking and Security Research Center, Computer Science and Engineering, Pennsylvania State University*, University Park, PA, October, 2010.

Security Challenges and Solutions in Mobile Smartphone Applications. *Security Day Seminar, Penn State University*, University Park, PA, October, 2010.

The Changing Vulnerability Landscape. *Association for Computing Machinery, Penn State Student Chapter*, University Park, PA, September, 2010.

The Changing Vulnerability Landscape. *ExxonMobil*, Falls Church, VA, March, 2010.

The Impact of Supply Chain on Information and Communications Technology Security. *The 1st Workshop on Telecommunications Infrastructure Protection and Security*, Honolulu, HI, December, 2009.

Energy Theft in the Advanced Metering Infrastructure. *Networking and Security Research Center, Computer Science and Engineering, Pennsylvania State University*, State College, PA, October, 2009.

Secure Provenance in High-End Computing Systems. *NSF HECURA FSIO PI Meeting*, Arlington, VA, August, 2009.

Missing Glue: Architectural Support for Security Annotations. *National Science Foundation Security Driven Architecture Workshop*, Arlington, VA, July, 2009.

Scalable Integrity-Justified Content Provenance. *NSERC ISSNNet Workshop*, Ottawa, Canada, June, 2009.

Utility Grid Automation and Risk Management. *Clean Technology Conference and Expo*, Houston, Texas, May, 2009.

Scalable Integrity-Justified Content Provenance. *Center for Applied Cybersecurity Research, Indiana University*, Bloomington, IN, April, 2009.

Scalable Integrity-Justified Content Provenance. *Department of Electrical Engineering and Computer Science, University of Michigan*, Ann Arbor, MI, April, 2009.

What is Security. *Dickenson Law School, Penn State University*, State College, PA, April, 2009.

Scalable Integrity-Justified Content Provenance. *Department of Computer Science and Engineering, Notre Dame University*, South Bend, IN, April, 2009.

Electronic Voting: The Good, the Bad, and the Reality. *Software Engineering Research Center Showcase*, Muncie, IN, November, 2008.

Ohio Voting Systems Integrity: The EVEREST Report. *Networking and Security Research Center, Computer Science and Engineering, Pennsylvania State University*, State College, PA, October, 2008.

Data Provenance: Challenges and Technology. *Cyber Physical System Security Forum, Cyber Security Research and Development Review*, Washington DC, October, 2008.

System-Wide Information Flow Enforcement. *NICAR PI Meeting*, Washington DC, September, 2008.

Presto: Configuration Management at Massive Scale. *NSF Workshop on Assurable and Usable Security Configuration*, George Mason University, Fairfax, VA, August, 2008.

Asymmetry in Performance and Security Requirements for I/O in High-end Computing. *NSF HECURA FSIO PI Meeting*, Arlington, VA, August, 2008.

Authentication and Web Security. *Security and Privacy in IT-EMTM 604 Guest Lecture, University of Pennsylvania*, Philadelphia, PA, May, 2008.

SPAM and SPAM Mitigation. *Computer Science Department, St. Vincent's University*, Latrobe, PA, April, 2008.

Phones, The Press, Research and Grad School .. or how to make trouble and have fun doing it. *Computer Science Department, St. Vincent's University*, Latrobe, PA, April, 2008.

Applications and Services in Telecommunications Networks. *NSF Wireless Security Workshop, Georgia Institute of Technology*, Atlanta, GA, March, 2008.

Vulnerabilities and Opportunities in SMS-Capable Cellular Networks. *Computer Science Department, Carleton University*, Ottawa, Canada, March, 2008.

Ohio Voting Systems Integrity: The EVEREST Report. *Case-Western Reserve University*, Cleveland, OH, February, 2008.

Ohio Voting Systems Integrity: The EVEREST Report. *Ohio State University*, Columbus, OH, February, 2008.

Ohio Voting Systems Integrity: The EVEREST Report. *Ohio University*, Athens, OH, February, 2008.

Ohio Voting Systems Integrity: The EVEREST Report. *Miami University, Ohio*, Oxford, OH, February, 2008.

Ohio Voting Systems Integrity: The EVEREST Report. *Bowling Green State University*, Bowling Green, OH, February, 2008.

Vulnerabilities and Opportunities in SMS-Capable Cellular Networks. *Computer Science Department, Indiana University of Pennsylvania*, Indiana, PA, September, 2007.

Asymmetry in Performance and Security Requirements for I/O in High-End Computing. *HECIWG FSIO 2007 Workshop*, NSF, Arlington, VA, August, 2007.

Toward Valley-Free Interdomain Routing. *IEEE International Conference on Communications (ICC) 2007*, Glasgow, Scotland, June, 2007.

Extending Developer Tools for Security-Typed Languages. *Software Engineering Research Center Fall Showcase*, West Lafayette, IN, June, 2007.

Open Functionality in SMS/Cellular Networks. *Computer and Information Science*, University of Oregon, Eugene, OR, May, 2007.

Open Functionality in SMS/Cellular Networks. *Computer Security Symposium*, St. Cloud State University, St. Cloud, MN, May, 2007.

Authentication and Web Security. *Security and Privacy in IT-EMTM 604 Guest Lecture*, University of Pennsylvania, Philadelphia, PA, April, 2007.

Grains of SANs: Building Storage Area Networks from Memory Spots. *CISCO Remote Faculty Seminar*, University Park, PA, April, 2007.

Grains of SANs: Building Storage Area Networks from Memory Spots. *2007 IEEE Security and Privacy Crystal Ball Workshop*, Hawthorne, NY, January, 2007.

Keynote Address—Password Exhaustion: Predicting the End of Password Usefulness. *2nd International Conference on Information Systems Security*, Kolkata, India, December, 2006.

Privacy Preserving Web-based Email. *2nd International Conference on Information Systems Security*, Kolkata, India, December, 2006.

Keynote Address—Physical and Digital Convergence: Where the Internet is the Enemy. *Eighth International Conference on Information and Communications Security (ICICS '06)*, Raleigh, NC, December, 2006.

Extending Developer Tools for Security-Typed Languages. *Software Engineering Research Center Fall Showcase*, Muncie, IN, November, 2006.

Open Functionality in SMS/Cellular Networks. *Johns Hopkins University, Computer Science Department*, Baltimore, MD, September, 2006.

Open Functionality in SMS/Cellular Networks. *George Mason University, Computer Science Department*, Fairfax, VA, September, 2006.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *Motorola Security Symposium*, Itasca, IL, September, 2006.

lseb: Testing Large Scale BGP Security in Replayable Network Environments. *NSF/DETER Community Workshop*, Arlington, VA, June, 2006.

BGPRV: A Library for Fast and Efficient Routing Data Manipulation. *NSF/DETER Community Workshop*, Arlington, VA, June, 2006.

JifClipse: Extending Developer Tools for Security-Typed Languages. *Software Engineering Research Center Spring Showcase*, Schaumburg, IL, June, 2006.

Trends in Security: Critical Engineering in the Large. *Schlumberger InnovateIT! 2006*, Cambridge, MA, May, 2006.

Information Flow Revisited: Software Engineering to Provable Security. *Network Center of Excellence, Motorola Labs*, Schaumburg, IL, May, 2006.

Authentication and Web Security. *Security and Privacy in IT-EMTM 604 Guest Lecture*, University of Pennsylvania, Philadelphia, PA, April, 2006.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *InfraGard Pittsburgh Chapter General Meeting*, Pittsburgh, PA, March, 2006.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *Distinguished Lecture, Computer Science Department, University of Virginia*, Charlottesville, VA, January, 2006.

Software Engineering Tools for Security-Typed Languages: Using Eclipse to Make Secure Programming Practical. *Software Engineering Research Center Showcase*, Muncie, IN, November, 2005.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *AT&T IP Services Security Council*, Middletown, NJ, October, 2005.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *Computer Science Department, Carnegie Mellon University*, Pittsburgh, PA, October, 2005.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *Computer Science Department, Yale University*, New Haven, CT, October, 2005.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *Computer Science Department, SUNY-Stony Brook*, Stony Brook, NY, October, 2005.

Exploiting Open Functionality in SMS-Capable Cellular Networks. *Networking and Security Research Center, Computer Science and Engineering, Pennsylvania State University*, State College, PA, October, 2005.

Isab: Trace Driven Modeling of Internet-Scale BGP Attacks and Countermeasures. *2nd Annual DETER/EMIST Workshop*, Newport Beach, CA, September, 2005.

Critical Infrastructure Security through Provably Secure Network Mediation. *2nd Japan/US Workshop on Critical Information Infrastructure Protection (CIIP)*, Tokyo, Japan, June, 2005.

Extending Developer Tools for Security-typed Languages. *Software Engineering Research Center Showcase*, West Lafayette, IN, June, 2005.

Origin Authentication in Interdomain Routing. *2005 IEEE Communications Quality & Reliability (CQR) International Workshop*, St. Petersburg, FL, April, 2005.

Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *Messiah College, Senior Seminar Series*, Grantham, PA, April, 2005.

Origin Authentication in Interdomain Routing. *Intel Research*, Folsom CA, April, 2005.

Key Distribution Strategies For Low-Power Wireless Networks. *Network Center of Excellence, Motorola Labs*, Schaumburg, IL, April, 2005.

Policy Evolution: Autonomic Environmental Security. *Software Engineering Research Center Showcase*, Muncie, IN, December, 2004.

Information Assurance for Enterprise Networks. *BAE Systems, Networking Seminar*, Reston, VA, November, 2004.

Origin Authentication in Interdomain Routing. *Computer Science Department, Purdue University*, West Lafayette, IN, October, 2004.

Origin Authentication in Interdomain Routing. *Electrical Engineering and Computer Science Department, University of Michigan*, Ann Arbor, MI, October, 2004.

Origin Authentication in Interdomain Routing. *Computer Science Department, University of Wisconsin*, Madison, MD, September, 2004.

Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *Computer Science and Engineering Student Organization*, University Park, PA, September, 2004.

Useless Metaphors? Why Specifying Policy is So Hard. *Workshop on Usable Privacy and Security Software, Center for Discrete Mathematics and Theoretical Computer Science (DIMACS)*, New Brunswick, New Jersey, July, 2004.

Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *Information Systems Research Seminar, Stern School of Business, New York University*, New York, NY, April, 2004.

Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *Ball State University*, Muncie, IN, April, 2004. Distinguished Speaker.

Origin Authentication in Interdomain Routing. *Computer Science Department, University of Illinois*, Champaign, IL, March, 2004.

Origin Authentication in Interdomain Routing. *Computer Science and Engineering Department, University of Minnesota*, Minneapolis, MN, March, 2004.

Origin Authentication in Interdomain Routing. *Computer Science Department, Johns Hopkins University*, Baltimore, MD, March, 2004.

Origin Authentication in Interdomain Routing. *Department of Computer Science, University of Massachusetts - Amherst*, Amherst, MA, March, 2004.

Origin Authentication in Interdomain Routing. *Computer Science and Engineering Department, Penn State University*, University Park, PA, March, 2004.

Origin Authentication in Interdomain Routing. *School of Electrical Engineering and Computer Science, Oregon State University*, Corvallis, OR, February, 2004.

Origin Authentication in Interdomain Routing. *Computer Science Department, Northwestern University*, Evanston, IL, February, 2004.

Origin Authentication in Interdomain Routing. *Computer Science Department, SUNY-Stony Brook*, Stony Brook, NY, February, 2004.

Attack Profiling and Simulation in Interdomain Routing. *P2INGS Quarterly Meeting*, Tempe, AZ, February, 2004.

Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *AT&T Finance Lunch*, Morristown, NJ, January, 2004.

Origin Authentication in Interdomain Routing. *AT&T IP Security Conference*, Middletown, NJ, November, 2003.

Origin Authentication in Interdomain Routing. *10th ACM Conference on Computer and Communications Security (CCS)*, Washington, DC, October, 2003.

Origin Authentication in Interdomain Routing. *Computer Science Department, Stevens Institute of Technology*, Hoboken, NJ, October, 2003.

Origin Authentication in Interdomain Routing. *Computer Science Department, Arizona State University*, Mesa, AZ, September, 2003.

Analysis of Security Vulnerabilities in the Movie Production and Distribution Process. *31st Technology Policy Research Conference (TPRC)*, Arlington, VA, September, 2003.

On Context in Authorization Policy. *8th ACM Symposium on Access Control Models and Technologies*, Como, Italy, June, 2003.

The Antigone Project. *DARPA Principal Investigator Meeting*, San Antonio, TX, January, 2003.

Methods and Limitations of Security Policy Reconciliation. *2002 IEEE Symposium on Security and Privacy*, Oakland, CA, May, 2002.

Policy Management in Distributed Systems. *Cigital*, Washington, DC, April, 2002.

Antigone: Policy Management in Secure Group Communication. *School of Computer Science, Carnegie Mellon University*, Pittsburgh, PA, April, 2001.

Antigone: Policy Management in Secure Group Communication. *Computer Science Department, University of Wisconsin*, Madison, MD, April, 2001.

Antigone: Policy Management in Secure Group Communication. *Computer Science Department, University of Maryland*, College Park, MD, April, 2001.

Antigone: Policy Management in Secure Group Communication. *Computer Science Department, University of North Carolina, Chapel Hill*, Chapel Hill, NC, April, 2001.

Antigone: Policy Management in Secure Group Communication. *Computer Science Department, Johns Hopkins University*, Baltimore, MD, March, 2001.

Antigone: Policy Management in Secure Group Communication. *AT&T Shannon Laboratory*, Florham Park, NJ, February, 2001.

Antigone: Policy Management in Secure Group Communication. *Telcordia Applied Research Laboratory*, Morristown, NJ, February, 2001.

Policy Problem Area 3 - Overview and Requirements. *Internet Engineering Task Force MSEC BOF*, San Diego, CA, December, 2000.

Multicast Security Policy Requirements and Building Blocks. *Quarterly Secure Multicast Research Group Meeting (SMuG)*, San Diego, CA, December, 2000.

Antigone: Implementing Policy in Secure Multiparty Communication. *Systems Design and Implementation (SDI) / Laboratory for Computer Systems (LCS) seminar series, School of Computer Science, Carnegie Mellon University*, Pittsburgh, PA, November, 2000.

Secure Group Communication in Antigone 2.0. *11th Annual IPoCSE Research Symposium*, Ann Arbor, MI, October, 2000.

Antigone Secure Group Communication. *Bi-Annual DARPA Visit, Software System Research Laboratory*, Ann Arbor, MI, September, 2000.

Problem Area 3: Policy. *Quarterly Secure Multicast Research Group Meeting (SMuG)*, Pittsburgh, PA, July, 2000.

Windowed Certificate Revocation. *IEEE INFOCOM 2000*, Tel Aviv, Israel, March, 2000.

A Response to ‘Can We Eliminate Certificate Revocation Lists?’. *Financial Cryptography 2000*, Anguilla, British West Indies, February, 2000.

Multicast Security Policy Definition. *Quarterly Secure Multicast Research Group Meeting (SMuG)*, Washington, DC, November, 1999.

Antigone: A Flexible Framework for Secure Group Communication. *Quarterly Secure Multicast Research Group Meeting (SMuG)*, NAI Labs, Baltimore, MD, September, 1999.

Antigone: A Flexible Framework for Secure Group Communication. *10th Annual IPoCSE Research Symposium*, Ann Arbor, MI, September, 1999.

Antigone: A Flexible Framework for Secure Group Communication. *8th USENIX Security Symposium*, Washington, DC, August, 1999.

Antigone: A Flexible Framework for Secure Group Communication. *IBM Watson Security Seminar*, Westchester County, NY, July, 1999.

Windowed Revocation in Public Key Infrastructures. *Department of Electrical Engineering and Computer Science, University of Michigan*, Ann Arbor, MI, September, 1998.

Scalable Key Distribution Hierarchy. *9th Annual IPoCSE Research Symposium*, Ann Arbor, MI, March, 1998.

JavaLauncher Applet Platform. *NASA, Kennedy Space Center Security Seminar*, Kennedy Space Center, FL, January, 1998.

Secure High Performance Group Communication, Directed Study Defense. *Department of Electrical Engineering and Computer Science, University of Michigan*, Ann Arbor, MI, September, 1997.

Tutorials

Understanding Android’s Security Framework. *ACM Conference on Computer and Communications Security (CCS)*, New York, NY, October, 2008. Joint tutorial with William Enck.

Web Security. *The Thirteenth International World Wide Web Conference (WWW2004)*, New York, NY, May, 2004.

Network and Information Security. *Regional Laboratory for Network Engineering Research and Training Institute, National Science Foundation, Jackson State University*, Jackson, MS, March, 2003.

APPENDIX 2: MATERIALS CONSIDERED

Exhibit No.	Description
1001	U.S. Patent No. 10,225,588 to Kiefer et al. (“the ’588 patent”)
1002	Prosecution File History for the ’588 patent
1003	Declaration of Patrick McDaniel, Ph.D.
1004	Prosecution File History for U.S. Patent Application No. 13/340,623 to Kiefer et al. (“Kiefer”)
1005	U.S. Provisional Patent Application No. 61/530,305
1006	U.S. Patent Pub. No. 2011/0096828 (“Chen”)
1007	U.S. Patent Pub. No. 2007/0083467 (“Lindahl”)
1008	U.S. Patent No. 8,683,066 (“Hurst”)
1009	U.S. Patent No. 8,849,950 (“Stockhammer”)
1010	U.S. Patent No. 8,243,924 (“Chen-924”)
1011	U.S. Patent No. 6,415,031 (“Colligan”)
1012	U.S. Patent No. 6,957,350 (“Demos”)
1013	U.S. Patent No. 7,295,673 (“Grab-673”)
1014	U.S. Patent No. 5,574,785 (“Ueno”)
1015	Divx, LLC v. Netflix Inc., No. 2:19-cv-1602-PSG, Am. Compl. (C.D. Cal Aug. 21, 2019)

APPENDIX 3: CHALLENGED CLAIMS

1[a]. A playback device for playing protected content from a plurality of alternative streams, comprising:

[b] a set of one or more processors; and

a non-volatile storage containing an application for causing the set of one or more processors to perform the steps of:

[c] obtaining a top level index file identifying a plurality of alternative streams of protected video,

[d] wherein each of the alternative streams of protected video includes partially encrypted video frames

[e] that are encrypted using a set of common keys comprising at least one key,

[f] and wherein the partially encrypted video frames contain encrypted portions and unencrypted portions of data;

[g] obtaining a copy of the set of common keys;

[h] detecting streaming conditions for the playback device;

[i] selecting a stream from the plurality of alternative streams of protected video based on the detected streaming conditions;

[j] receiving a container index that provides byte ranges for portions of the selected stream of protected video within an associated container file;

[k] requesting portions of the selected stream of protected video based on the provided byte ranges;

[l] locating encryption information that identifies encrypted portions of frames of video within the requested portions of the selected stream of protected video;

[m] decrypting each encrypted portion of the frames of video identified within the located encryption information using the set of common keys; and

[n] playing back the decrypted frames of video obtained from the requested portions of the selected stream of protected video.

2[a]. The playback device of claim 1, wherein the step of requesting portions of the selected stream further comprises the step of obtaining a container file containing protected video from at least one of the plurality of alternative streams,

[b] where the container file also includes encryption information that identifies portions of frames of video that are encrypted and

[c] a reference to at least one key from the set of common keys to utilize in accessing the encrypted portions of the frames of video.

3. The playback device of claim 1, wherein the located encryption information comprises a reference to the start of an encrypted block of data.

4. The playback device of claim 3, wherein the located encryption information further comprises the size of the encrypted block of data.

5. The playback device of claim 3, wherein the located encryption information further comprises cryptographic information that can be utilized to access the encrypted portion of the frame.

6. The playback device of claim 5, wherein the cryptographic information is a reference to at least one key from the set of common keys.

7[a]. The playback device of claim 1, wherein the application is further for causing the set of processors to perform the steps of:

[b] detecting a change in the streaming conditions;

[c] identifying a second alternative stream of protected video based on the detected change;

[d] receiving a container index that provides byte ranges for portions of the second alternative stream of protected video within a second associated container file;

requesting portions of the second alternative stream of protected video based on the provided byte ranges;

[e] decrypting each encrypted portion of the frames of video from the requested portions of the second alternative stream of protected video using the set of common keys; and

[f] playing back the decrypted frames of video from the requested portions of the second alternative stream of protected video.

8. The playback device of claim 1, wherein the application is further for causing the set of processors to perform the steps of:

transmitting a request for content to a set of one or more content distribution servers; and

receiving the content from the set of one or more content distribution servers.

9. The playback device of claim 8, wherein the application is further for causing the set of processors to perform the steps of:

transmitting a request for cryptographic information including the set of common keys to a set of one or more DRM servers; and

receiving the cryptographic information from the set of one or more DRM servers.

10. The playback device of claim 1, wherein the application is further for causing the set of processors to perform the step of obtaining the container index

from at least one file selected from the group consisting of: the associated container file containing the selected stream of protected content; and a separate file to the associated container file containing the selected stream of protected content.

11. The playback device of claim 1, wherein the common set of keys comprises a plurality of keys.

12[a]. A method for playing protected content from a plurality of alternative streams on a playback device, the method comprising:

[c] obtaining a top level index file identifying a plurality of alternative streams of protected video,

[d] wherein each of the alternative streams of protected video includes partially encrypted video frames

[e] that are encrypted using a set of common keys comprising at least one key,

[f] and wherein the partially encrypted video frames contain encrypted portions and unencrypted portions of data;

[g] obtaining a copy of the set of common keys;

[h] detecting streaming conditions for the playback device;

[i] selecting a stream from the plurality of alternative streams of protected video based on the detected streaming conditions;

[j] receiving a container index that provides byte ranges for portions of the selected stream of protected video within an associated container file;

[k] requesting portions of the selected stream of protected video based on the provided byte ranges;

[l] locating encryption information that identifies encrypted portions of frames of video within the requested portions of the selected stream of protected video;

[m] decrypting each encrypted portion of the frames of video identified within the located encryption information using the set of common keys; and

[n] playing back the decrypted frames of video obtained from the requested portions of the selected stream of protected video using a decoder.

13[a]. The method of claim 12, wherein requesting portions of the selected stream further comprises obtaining a container file containing protected video from at least one of the plurality of alternative streams,

[b] where the container file also includes encryption information that identifies portions of frames of video that are encrypted and

[c] a reference to at least one key from the set of common keys to utilize in accessing the encrypted portions of the frames of video.

14. The method of claim 12, wherein the located encryption information comprises a reference to the start of an encrypted block of data.

15. The method of claim 14, wherein the located encryption information further comprises the size of the encrypted block of data.

16. The method of claim 14, wherein the located encryption information further comprises cryptographic information that can be utilized to access the encrypted portion of the frame.

17. The method of claim 16, wherein the cryptographic information is a reference to at least one key from the set of common keys.

18[a]. The method of claim 12 further comprising:

[b] detecting a change in the streaming conditions;

[c] identifying a second alternative stream of protected video based on the detected change;

[d] receiving a container index that provides byte ranges for portions of the second alternative stream of protected video within a second associated container file;

requesting portions of the second alternative stream of protected video based on the provided byte ranges;

[e] decrypting each encrypted portion of the frames of video from the requested portions of the second alternative stream of protected video using the set of common keys; and

[f] playing back the decrypted frames of video from the requested portions of the second alternative stream of protected video.

19. The method of claim 12 further comprising:

transmitting a request for content to a set of one or more content distribution servers; and

receiving the content from the set of one or more content distribution servers.

20. The method of claim 19 further comprising:

transmitting a request for cryptographic information including the set of common keys to a set of one or more DRM servers; and

receiving the cryptographic information from the set of one or more DRM servers.

21. The method of claim 12 further comprising obtaining the container index from at least one file selected from the group consisting of: the associated container file containing the selected stream of protected content; and a separate

file to the associated container file containing the selected stream of protected content.

22. The method of claim 12, wherein the common set of keys comprises a plurality of keys.

23. The playback device of claim 1, wherein:
the container index is part of a hierarchical index; and
the application is further for causing the set of processors to perform the step of:

obtaining a lower layer index that identifies the location of frames within a specific requested portion of the selected stream of protected video; and

identify partially encrypted video frames from within the specific requested portion of the selected stream using the lower layer index.

24. The method of claim 12, wherein:
the container index is part of a hierarchical index; and
the method further comprises:
obtaining a lower layer index that identifies the location of frames within a specific requested portion of the selected stream of protected video; and identify partially encrypted video frames from within the specific requested portion of the selected stream using the lower layer index.