

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

REPLY DECLARATION OF PATRICK MCDANIEL, PH.D.

I, Patrick McDaniel, declare as follows:

1. I am the same Patrick McDaniel that previously submitted a declaration in this proceeding which includes the scope of my work, qualifications, and background (EX1003). I have been asked to respond to Dr. Nielson's Declaration (EX2012). I have reviewed and considered each of the materials listed in the attached Appendix 1. 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. A POSITA would have been able to modify a standard file format to use partial encryption.

2. Dr. Nielson argues that "it was beyond the skills of a POSITA to modify the standard file formats to implement partial frame encryption..." EX2012, ¶80. This is incorrect.

3. In my opening declaration, I explained that Chen-924 provided express teaching, suggestion, and motivation to combine adaptive bitrate streaming, for example as taught by Chen, with partial encryption, for example as taught by Lindahl. EX1003, ¶117. Moreover, Chen-924 expressly teaches partially encrypting video frames in an adaptive bitrate streaming system *using 3GPP or MP4 file formats*. See EX1010 (Chen-924), 5:63-6:9 ("In one embodiment, the containers may be selectively encrypted...Selective encryption may further include selectively *encrypting at least a portion of an I-frame, P-frame, B-frame*, and any

combination of P, B, and I frames within the container stream.”), 14:51-55 (“downloading of the container may be performed using any of a variety of mechanisms, including, but not limited to random access, *adaptive streaming*...”), 3:2-6, 4:14-15 (“A non-limiting and non-exhaustive list of examples of a container is: MPEG2-TS, FLV, H.264, MOV, **MP4**, **3GP**, and ASF.”). Therefore, Chen-924 confirms that actual POSITAs knew how to apply partial encryption to the 3GPP and MP4 file formats. Chen-924 provides express motivation to combine the prior art in the manner explained in my opening declaration while demonstrating that POSITAs had a reasonable expectation of success in doing so.

4. Moreover, Lindahl expressly teaches that its partial encryption can be applied to popular file formats, including the MPEG-4 format. EX1007, ¶[0067] (“*The file format is typically dependent on the type of media.* For example, when the type of media is video, one suitable file format is a H.264 format. Other suitable formats include QuickTime movies, MPEG-1 format, MPEG-2 format and **MPEG-4 format.**”). Therefore, a POSITA would have understood Lindahl’s teachings to include the use of known file formats such as the MPEG-4 file format. A POSITA would have had a motivation and reasonable expectation when applying Lindahl’s partial encryption to known file formats, such as MPEG-4, because it is taught by Lindahl and other prior art references.

5. In my opening declaration, I noted that a POSITA would have been able to use known file formats including the MPEG-4 file format and 3GPP extensions. EX1003, ¶¶68. Dr. Nielson argues that I did not include the word “modify” in my description. EX2012, ¶¶80. However, the use of 3GPP and MPEG-4 file formats encompasses the ability to extend those formats because their architectures were extensible by design. A key aspect of these file formats was their extensibility—a fact that Dr. Nielson did not consider. Indeed, 3GPP and MP4 are both derived from the ISO base media file format.¹ EX1006, ¶¶[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.”); EX1003, ¶¶165. Chen explains that various specifications have been derived from the MPEG-4 file format, including the 3GPP file format, and that “*derived specifications* can map all described features to the more general ISO base media file formats defined in ISO/IEC 14496-12...” EX1006, ¶¶[0070]. Therefore, Chen confirms that modification of file formats was common in the art, and as discussed in this section, the prior art is replete with examples that confirm Chen’s teaching.

¹ MPEG-4 Part 12 is also referred to as the ISO base media file format or ISO/IEC 14496-12. MPEG-4 Part 14 (sometimes called the MP4 file format) is an extension of the ISO base media format.

6. Multiple references teach modifying standard file formats to implement partial encryption, confirming that a POSITA would have known how to do so. For example, Shamoon teaches partially encrypting video frames in an MPEG-4 file and modifying the MPEG-4 file format by adding an Initial Object Descriptor that contains a DigiBox. EX1018, 22:26-38, 2:5-9 (“The digital bit stream includes content, which is encrypted at least in part...”), 23:50-56 (“Shamoon teaches that this DigiBox is a created box that contains keys and rules for decrypting content in the MPEG-4 file.”). EX1018, 22:7-45. As another example, Shi teaches modifying the early MPEG-1 standard to implement VEA partial encryption on a modified video stream format. EX1017, 83 (“VEA, a selective encryption algorithm”). Shi teaches that modification of video stream processing to implement partial encryption improves performance, a motivation I discuss in my opening declaration. EX1017, 85 (“VEA ... only selectively encrypts a small number of bits of the MPEG compressed video.”). Thus, prior art teachings confirm the fact that a POSITA would have been able to modify standard file formats to implement partial frame encryption.

7. Beyond encryption, various prior art references teach the modification of standard file formats, confirming that a POSITA would have known how to modify standard file formats to suit their particular applications. For example, Frojdh teaches extending the 3GP file format by adding a new box with a top-level

file tree within a 3GP file, for multimedia presentations. *See* EX1020, 4:19-28, 5:32-45, FIGS. 8, 9.

8. It was common to modify MPEG standards for data security enhancements, including for example for steganographic watermarking of digital content, where a hidden encoding of data was added into the content to allow a content owner to trace the origin of pirated data. From a technical perspective, steganography is closely related to encryption, and it often involved modifications to MPEG files. For example, in an early watermarking system, the authors of “Steganography in Compressed Video Stream” modified the MPEG standards to place “control information to facilitate data extraction was embedded in I frame, in P frames and B frames.” EX1021, Abstract. This information was used to track and recover the content protecting watermarks.

9. In summary, as explained above, the prior art includes express teaching, suggestion, and motivation to apply partial encryption of video frames to the MP4 and 3GPP file formats. The MP4 and 3GPP file formats were extensible by design, and the prior art includes express teaching, suggestion, and motivation to extend those file formats. A POSITA would have known how to use popular file formats, which includes extending them. EX1003, ¶68. This is confirmed by the teachings of the prior art discussed above, as well as my own personal experience. For example, as a graduate student, I myself worked directly with the MPEG

standards in developing a video conferencing system that distributed encrypted content to members of a multicast group (i.e., broadcasting data to a set of host computers). EX1022, 8 (“We added support for Sun hardware MPEG-1 encoding, multiple platform software MPEG-1 decoding, and WIN32 hardware decoding.”). Additionally, I have advised and worked with several graduate students that have modified MPEG and other video software as part of their research. As a practical matter, a POSITA would not have found it technically challenging to change the file format for MPEG or other content standards. Modifying the meta-data or content data formats to include partial encryption parameters would simply have involved modification of the encoding software, to write the data, and playback software, to read it. Therefore, for the reasons explained above, a POSITA would have had a reasonable expectation of success in extending known file formats, such as the MP4 and 3GPP file formats, for partial encryption, including for the combination of Chen, Lindahl, and Hurst.

II. FEC is compatible with partial encryption.

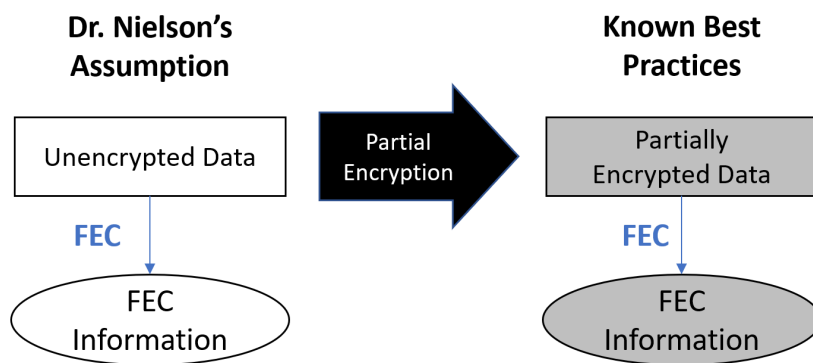
10. Dr. Nielson argues that the combination of Forward Error Correction (“FEC”) with partial encryption has the potential to compromise the security of encryption. *See* EX2012, ¶¶68-69. Dr. Nielson argues that “FEC provides correction information for packets” that can be used to decrypt encrypted portions of packets based on earlier packets. *Id.* This is incorrect because it assumes,

contrary to known best practices, that the correction information contains information about the packets *before* they were encrypted, i.e., that FEC was applied before encryption and thus reveals information about unencrypted data.

11. In his testimony, Dr. Nielson appears to be criticizing his own strawman. A POSITA would understand no reasonable system would provide FEC on unencrypted data in the way suggested for at least two reasons. First, as suggested by Dr. Nielson, such a design would provide theoretical information on unencrypted data—flying in the face of security best practices (and common sense). Second, the FEC would not work—the data would have to be decrypted first in order to apply the error correction, but the lack of access to the ciphertext (because of errors or packet loss) means that the data could not be decrypted. Simply put, a POSITA would have known that such an FEC-before-encryption approach would be unworkable and would discard it out of hand.

12. Various prior art references (and known practices) teach the application of FEC-*after*-encryption, particularly for partial encryption, because “[a]pplying FEC to data which already has been partially encrypted ***does no harm*** with respect to the data security.” EX1023, 217 (emphasis added). When used in this manner, FEC correction information only contains information about partially encrypted data—not unencrypted data—and therefore cannot be used to discern information about unencrypted data, as illustrated below. Based on my decades of

experience in computer security and streaming video, this was a well-known, and well-established practice in the art.



III. The combination teaches alternative streams that are encrypted using a set of common keys

A. The combination with Hurst teaches alternative streams that are encrypted using a set of common keys

13. As an initial matter, I note that Dr. Nielsen admits that Hurst teaches this claim limitation. *See* EX2012, ¶117 (“The streamlets 304 of different bitrates may be encrypted with the same key.”) (citing EX1008, 18:66-67); *see also* EX2012, ¶120 (“Hurst... disclos[es] the possibility of using the same key”). Dr. Nielson argues that “there is no disclosure in Hurst to suggest to a POSITA that use of the same key to encrypt alternate streams of protected video provides adequate security in a system that also uses partial frame encryption as disclosed by Lindahl—as using both the same key and partial encryption could negatively impact security.” EX2012, ¶118.

14. In his argument, Dr Nielsen seems to restrict the combined system to an undisclosed and overly restrictive threat model for the combined system. As a practical matter, situations where a legitimate user must be specifically prevented from viewing an alternate stream were uncommon. There were other (more common) situations where a single key was sufficient. For example, if I were to use adaptive streaming to provide a recorded video of my course lecture to legitimate students, I would not care which of the alternate streams they watch because the alternate streams vary in bitrate and video quality but contain the same content. Using a single key for all alternate streams would provide a way to provide access to all streams to the students while still retaining confidentiality to only students in my class. Indeed, I used this single key solution to encrypt multiple streams in the security conferencing systems discussed above. EX1022, 3 (“All participants use the same data encryption key for the video and audio streams.”). In that system, a single key (called a session or group key) was used to encrypted all streams at any given time.

15. Moreover, partial encryption was often used for video applications with lower security requirements, where content protection only needed to provide enough security to create a deterrent effect, e.g., against potential piracy. *See* EX1003, ¶84; EX1017, 83 (“Multimedia applications appeal [to] those encryption algorithms which require less computation and low implementation cost. For

multimedia data, we only need to provide sufficient security level such that the cost to break the security system is much more expensive than to buy the program. Hence, light-weight and cost-effective multimedia data encryption algorithms should be developed.”) Even if using the same key had less security than using different keys, it still would have provided sufficient protection for many video applications. *See id.*

16. Additionally, my opening declaration explained that using the same set of keys aligns with the known usage model for adaptive bitrate streaming. When adaptively streaming a video, such as a movie, users are typically granted access to the alternative streams of the video, as a group, because they are expected to switch between the streams as streaming conditions change. EX1003, ¶81. Here, a POSITA would also recognize the basic utility of a single key solution. In such a system the user could simply switch between alternate streams as network and computational conditions change. EX1006, ¶¶[0063-0064]; EX1003, ¶116. There would be no need to acquire new keys or access and external DRM server. Indeed, this would have simplified the DRM and made the system more portable to diverse network and computing environments because the DRM system would be able to provide access without pre-assigning some particular alternate stream.

17. As I explained in my opening declaration, using the same set of keys is one of two known ways to encrypt the streams, and Hurst expressly teaches it.

EX1003, ¶¶146-148. Therefore, a POSITA would have been motivated to apply Hurst's teaching because it is one of a finite number of known ways for handling keys in an adaptive streaming system. EX1003, ¶¶146-148. Hurst teaches a natural, simple, efficient, and obvious approach to DRM for adaptive streaming.

B. The combination of Lindahl and Chen teaches alternative streams that are encrypted using a set of common keys.

18. Dr. Nielson argues that “Lindahl only teaches that metadata associated with a media file, but not the media data within the file, may be encrypted with a global key.” EX2012, ¶107. This is incorrect.

19. As I explained in my opening declaration, Lindahl teaches that “the *media file* can then be decrypted” using the cryptographic key or “*the global key*.” EX1007, ¶[0066]. EX1003, ¶144 (emphasis added). This teaching is directed to the media itself, not metadata—indeed the following sentence makes it clear that the global key can be used to decrypt both the media and metadata. “If the metadata of the media file itself is encrypted, the metadata can *also* be decrypted using the global key.” EX1007, ¶[0066] (emphasis added). Therefore Lindahl teaches or suggests that media data can all be encrypted and decrypted using the same global key, when the media data is not that sensitive. *See* EX1007, ¶[0048].

20. Dr. Nielson argues that “there is no disclosure cited from Lindahl that would require Lindahl to use a symmetric key system” and suggests that Lindahl teaches encrypting video content using asymmetric encryption. EX2012, ¶111.

This is incorrect. As I explained in my opening declaration, Lindahl teaches using symmetric encryption algorithms, such as Blowfish, AES and DES, meaning the same key was used for encryption and decryption. EX1003, ¶144. Content such as video was encrypted using symmetric key encryption because the alternative, asymmetric encryption, was far more computationally intensive. EX1003, ¶82. Therefore, a basic principle in cryptography, ranging back decades, was to use asymmetric encryption to exchange symmetric keys, which were then used to encrypt/decrypt data using a symmetric algorithm. *See* EX1023, 209 (“Public key methods usually are 100 to 1000 times slower than symmetric key methods. So they are merely used for a secure symmetric key distribution”). From my decades of experience in computer security, I am not aware of any system that used asymmetric encryption on video data itself, nor would not make sense to do so. Therefore, as I also explained in my deposition, when Lindahl teaches decrypting media using a global key (EX1007, ¶[0066]), a POSITA would have found it obvious that the media was encrypted using that same (symmetric) global key. EX2013, 79:8-13.

21. Dr. Nielson also argues that a POSITA would “have little motivation to use a global key to encrypt ‘all files’.” EX2012, ¶115. This is incorrect. As I explained in my opening declaration, a POSITA would have been motivated, for several reasons, to use a global key for content that is less sensitive and where

piracy is less of a concern. *See, e.g.*, EX1003, ¶¶144-145. Again, here Dr. Nielson apparently attempts to restrict the combined system to some (yet again unstated) threat model. In truth, a POSITA would understand that it is beneficial in many systems to provide a single key for all content (sometimes called a library access key). For example, in my lecture example above, I would likely provide a single key to all of my lectures and related course materials. EX2013, 98:12-17 (“So for things that are not very sensitive, ... a single key is a totally appropriate key management strategy...”). I would see no need to restrict students to any one lecture, but provide a single key that gave them access to the entire semester or set of courses. In this way, I could provide a key to the students at the beginning of the semester and not have to perform any other DRM action thereafter. Indeed, I could add lectures, reviews, homework help videos or other content whenever I wanted to without ever having to securely distribute other keys to students. Requiring a separate key for each lecture and video, as suggested by Dr. Nielson, would vastly complicate the process with little meaningful benefit for this type of application.

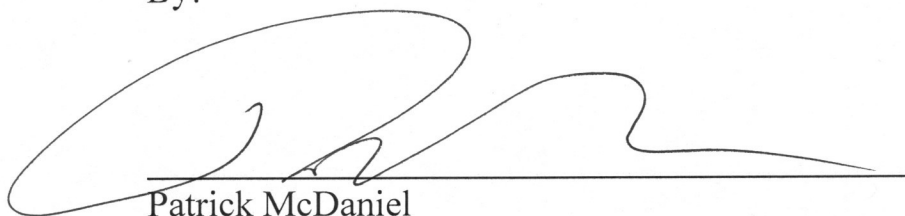
IV. Conclusion

22. 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 March 9, 2021.

By:

A handwritten signature in black ink, appearing to read 'Patrick McDaniel', is written over a horizontal line. The signature is stylized with a large loop at the beginning and a long, sweeping tail.

Patrick McDaniel

APPENDIX 1: MATERIALS CONSIDERED IN THE PREPARATION OF THIS DECLARATION

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)
1017	Changgui Shi and Bharat Bhargava, “A Fast MPEG Video Encryption Algorithm,” Association for Computing Machinery, 1998.
1018	U.S. Patent No. 7,233,948 (“Shamoon”)
1020	U.S. Patent No. 7,979,886 (“Frojd”)
1021	Changyong Xu, Xijian Ping and Tao Zhang, "Steganography in Compressed Video Stream," First International Conference on Innovative Computing, Information and Control - Volume I (ICICIC'06), 2006.
1022	Adamson W.A., Antonelli C.J., Coffman K.W., McDaniel P., and Rees J, “Secure Distributed Virtual Conferencing,” The International Federation for Information Processing, 1999.
1023	Thomas Kunkelmann, Hartmut Vogler, Marie-Luise Moschgarth, and Lars C. Wolf. “Scalable Security Mechanisms in Transport Systems for Enhanced Multimedia Services,” In Proceedings of the Third European Conference on Multimedia Applications, Services and Techniques (ECMAST '98), 1998.
2012	Declaration of Seth Nielson, Ph.D.

Exhibit No.	Description
2013	Deposition of Patrick McDaniel, Nov. 16, 2020
2016	Andy Patrizio, Why the DVD Hack Was a Cinch, WIRED, November, 2, 1999, available at https://www.wired.com/1999/11/why-the-dvd-hack-was-a-cinch/
2017	About Bruce Schneier, Schneier on Security, available at https://www.schneier.com/blog/about/
2018	Bruce Schneier, Why Cryptography Is Harder Than It Looks, Counterpane Systems, 1997, available at http://www.firstnetsecurity.com/library/counterpane/whycrypto.pdf
2019	Kevin Henry et. al., An Overview of the Advanced Access Content System (AACCS), available at http://cacr.uwaterloo.ca/techreports/2007/cacr2007-25.pdf
2020	Free-to-view, Wikipedia, available at https://en.wikipedia.org/wiki/Free-to-view

Paper No.	Description
10	DECISION Granting Institution of <i>Inter Partes</i> Review
17	Patent Owner's Response