

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

SECURITY FIRST INNOVATIONS, LLC,
Patent Owner.

Case IPR2024-00214
Patent 11,068,609

PATENT OWNER'S RESPONSE

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. AS THE BOARD HAS TWICE FOUND, THE ORSINI GROUND FAILS (ALL CLAIMS, GROUND 1).	1
A. The '609 Patent Claims Fundamentally Different Technology Than That Disclosed In Orsini.	1
B. The Board Correctly Found That Petitioner Fails To Show That Orsini Anticipates The Claims.	6
1. Orsini Does Not Inherently Disclose Claim 1.	11
2. Petitioner's Other Arguments Cannot Establish Anticipation Of Claim 1.....	15
C. Petitioner Fails To Show That Orsini Renders The Claims Obvious.	17
1. Petitioner Cannot Rely On Common Sense To Supply Missing Limitation 1[e].	18
2. Petitioner Fails To Show That A POSITA Would Have Been Motivated To Modify Orsini to Add Limitation 1[e].....	20
III. THE FOSTER GROUND FAILS (ALL CLAIMS, GROUND 2).....	32
A. The '609 And Foster Fundamentally Differ.....	32
B. Petitioner Fails To Show Foster Discloses "Generating A Plurality Of Data Chunks Based On A Data Set" (Limitation 1[c][1]).	34
C. Petitioner Fails To Show That Foster Discloses Or Renders Obvious "Executing Code By A Processor To Perform ... Storing ... At Least One Data Chunk ... On At Least One Storage Device" (Limitations 1[a] & 1[f]).	42

D.	Petitioner Fails To Show That Foster Discloses Or Renders Obvious “Storing, In A Memory Coupled To The Processor, At Least One Data Chunk . . . On At Least One Storage Device” (Limitation 1[f]).	48
1.	Petitioner Fails To Demonstrate That Foster Discloses Limitation 1[f].	49
2.	Petitioner Fails To Demonstrate A POSITA Would Modify Foster As Proposed.	52
IV.	CONCLUSION	56

TABLE OF AUTHORITIES

Page(s)

COURT DECISIONS

<i>Arctic Cat Inc. v. Bombardier Rec. Prods.</i> , 876 F.3d 1350 (Fed. Cir. 2017)	24
<i>Arctic Cat Inc. v. Polaris Indus.</i> , 795 Fed. App'x. 827 (Fed. Cir. 2019)	24, 56
<i>Arendi S.A.R.L. v. Apple, Inc.</i> , 832 F.3d 1355 (Fed. Cir. 2016)	18
<i>Ariosa Diagnostics v. Verinata Health, Inc.</i> , 805 F.3d 1359 (Fed. Cir. 2015)	28
<i>Baran v. Med. Device Techs., Inc.</i> , 616 F.3d 1309, 1316 (Fed. Cir. 2010)	41
<i>BlephEx, LLC v. Myco Indus.</i> , 24 F.4th 1391 (Fed. Cir. 2022)	17
<i>D Three Enters., LLC v. SunModo Corp.</i> , 890 F.3d 1042 (Fed. Cir. 2018)	12
<i>DSS Tech. Mgm't v. Apple Inc.</i> , 885 F.3d 1367 (Fed. Cir. 2018)	18
<i>Enplas Display Device Corp. v. Seoul Semiconductor Co., Ltd.</i> , 909 F.3d 398 (Fed. Cir. 2018)	11
<i>Finisar Corp. v. DirecTV Group, Inc.</i> , 523 F.3d 1323 (Fed. Cir. 2008)	17
<i>Forum US, Inc. v. Flow Valve, Ltd. Liab. Co.</i> , 926 F.3d 1346 (Fed. Cir. 2019)	12

<i>Haddad v. United States</i> , 164 Fed. Cl. 28, 66 (2023)	41
<i>Henny Penny Corp. v. Frymaster LLC</i> , 938 F.3d 1324 (Fed. Cir. 2019)	24, 56
<i>In re Benno</i> , 768 F.2d 1340 (Fed. Cir. 1985)	12
<i>In re Conrad</i> , 759 Fed. App'x. 982 (Fed. Cir. 2019)	31
<i>In re Kahn</i> , 441 F.3d 977 (Fed. Cir. 2006)	37, 48
<i>In re Magnum Oil Tools Int'l, Ltd.</i> , 829 F.3d 1364 (Fed. Cir. 2016)	37, 48
<i>In re NTP, Inc.</i> , 654 F.3d 1279 (Fed. Cir. 2011)	22
<i>In re Ratti</i> , 270 F.2d 810 (CCPA 1959)	56
<i>Intel Corp. v. PACT XPP Schweiz AG</i> , 61 F.4th 1373 (Fed. Cir. 2023)	29
<i>InTouch Techs., Inc. v. VGo Commc'ns., Inc.</i> , 751 F.3d 1327 (Fed. Cir. 2014)	31
<i>KSR Int'l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007)	29, 30, 37, 47
<i>LBT IP I LLC v. Apple Inc.</i> , 2023 U.S. App. LEXIS 14376 (Fed. Cir. 2023)	13
<i>Mantech Envtl. Corp. v. Hudson Envtl. Servs.</i> , 152 F.3d 1368 (Fed. Cir. 1998)	5
<i>Microsoft Corp. v. Biscotti, Inc.</i> , 878 F.3d 1052 (Fed. Cir. 2017)	13

<i>Net MoneyIN, Inc. v. VeriSign, Inc.</i> , 545 F.3d 1359 (Fed. Cir. 2008)	17
<i>Otsuka Pharm. Co., v. Sandoz, Inc.</i> , 678 F.3d 1280 (Fed. Cir. 2012)	31
<i>PAR Pharm. v. TWi Pharms., Inc.</i> , 773 F.3d 1186, 1195–96 (Fed. Cir. 2014).....	13
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303, 1313 (Fed. Cir. 2005)	42
<i>Plas-Pak Indus., Inc. v. Sulzer Mixpac AG</i> , 600 F. App'x 755 (Fed. Cir. 2015)	56
<i>Polaris Indus., Inc. v. Arctic Cat, Inc.</i> , 882 F.3d 1056 (Fed. Cir. 2018)	55
<i>Ruiz v. A.B. Chance Co.</i> , 357 F.3d 1270 (Fed. Cir. 2004)	30
<i>Schumer v. Lab. Computer Sys., Inc.</i> , 308 F.3d 1304 (Fed. Cir. 2002)	13
<i>Sci. Applications Int'l Corp. v. U.S.</i> , 154 Fed. Cl. 594, 623 (2021)	42
<i>Trintec Indus., Inc. v. Top U.S.A. Corp.</i> , 295 F.3d 1292, 1295 (Fed. Cir. 2002);	13
<i>Univ. of Strathclyde v. Clear-Vu Lighting LLC</i> , 17 F.4th 155 (Fed. Cir. 2021)	31
<i>Virtek Vision Int'l ULC v. Assembly Guidance Sys., Inc.</i> , 97 F.4th 882 (Fed. Cir. 2024)	28
<i>Vitronics Corp. v. Conceptronic</i> , 90 F.3d 1576, 1582 (Fed. Cir. 1996)	42

AGENCY DECISIONS

Apple Inc. v. Cellular Comm’ns Equipment, LLC,
IPR2015-00576, Paper 7 (June 12, 2015).....22

Edwards Lifesciences Corp. v. Aortic Innovations LLC,
IPR2023-01232, Paper 10 (Feb. 7, 2024).....29

F5 Networks, Inc. v. Radware, Ltd.,
IPR2017-00124, Paper 48 (Apr. 23, 2018).....23

RPX Corp. v. Parity Networks, LLC,
IPR2018-00097, Paper 7 (Apr. 24, 2018).....28

REGULATIONS

37 C.F.R. § 42.6(a)(3) 13

EXHIBIT LIST	
2001	Declaration of Nathan Lowenstein in Support of Motion for <i>Pro Hac Vice</i> Admission
2002	Declaration of Colette Woo in Support of Motion for <i>Pro Hac Vice</i> Admission
2003	Declaration of Aviel D. Rubin, Ph.D.
2004	Prosecution History of 10,452,854 Patent ['854-FH]
2005	Bruce Schneier Curriculum Vitae [Schneier-CV]
2006	Bruce Schneier, <i>Why Cryptography Is Harder Than It Looks</i> , SCHNEIER ON SECURITY, 1997, https://www.schneier.com/essays/archives/1997/01/why_cryptography_is.html [Schneier-Why-Cryptography-Is-Harder-Than-It-Looks]
2007	Excerpts from Jonathan Katz et al., <i>Introduction to Modern Cryptography</i> (1st ed. 2007) [Katz]
2008	Bruce Schneier, <i>Security Pitfalls in Cryptography</i> , SCHNEIER ON SECURITY, 1998, https://www.schneier.com/essays/archives/1998/01/security_pitfalls_in.html [Schneier-Security-Pitfalls-in-Cryptography]
2009	Excerpts from Ross Anderson, <i>Security Engineering: A Guide to Building Dependable Distributed Systems</i> (1st ed. 2001) [Anderson]
2010	Excerpts from Bruce Schneier, <i>Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C</i> (2nd ed. 2015) [Schneier-Applied-Cryptography]
2011	Patent Application Publication to Tsai US 2005/0195857 A1 [Tsai]
2012	Edgar Pek et al., <i>Formal Verification of Communication Protocols in Distributed Systems</i> , SYMPOSIUM ON OPERATING SYS. PRINCIPLES, 2003 [Pek]

2013	Jan Friso Groote et al., <i>A Bounded Retransmission Protocol for Large Data Packets: A Case Study in Computer Checked Algebraic Verification</i> , 1996 [Groote]
2014	Microsoft, <i>Overview of MPEG-2 Systems</i> , WWW.LEARN.MICROSOFT.COM , July 27, 2023, https://learn.microsoft.com/en-us/windows/win32/directshow/overview-of-mpeg-2-systems [Overview-of-MPEG-2-Systems]
2015	Excerpts from Harry Newton, <i>Newton's Telecom Dictionary</i> (16th ed. 2000) [Newton's-Telecom-Dictionary]
2016	Excerpts from <i>Oxford Dictionary of Computing</i> (1996) [Oxford-Dictionary-of-Computing]
2017	IBM, <i>Transmission Control Protocol (TCP)</i> , WWW.IBM.COM , https://www.ibm.com/docs/en/zos-basic-skills?topic=4-transmission-control-protocol-tcp [IBM]
2018	<i>Security First Innovations, LLC v. Google LLC</i> , 2:23-cv-00091 (JKW-RJK), Google's Opening Claim Construction Brief (December 6, 2023) [Google's-Opening-Claim-Construction-Brief]
2019	Excerpts from Microsoft Press, <i>Microsoft Computer Dictionary</i> (5th ed. 2002) [Microsoft-Computer-Dictionary]
2020	<i>Security First Innovations, LLC v. Google LLC</i> , 2:23-cv-00091 (JKW-RJK), Google's Responsive Claim Construction Brief (Non-Confidential Redacted Version) (Jan. 8, 2024) [Google-Responsive-Claim-Construction-Brief]
2021	Cambridge Dictionary, <i>Set-Top Box</i> , https://dictionary.cambridge.org/us/dictionary/english/set-top-box (last visited Feb. 27, 2024)

2022	Jim Connors, INFOWORLD, <i>The evolution of cable television to interactive communications service provider</i> (Oct. 1, 1996), https://www.infoworld.com/article/2077255/the-evolution-of-cable-television-to-interactive-communications-service-provider.html [InfoWorld]
2023-2027	Reserved
2028	Declaration of Colette Woo [SERVED, NOT FILED]
2029	Jonathan Katz et al., <i>Introduction to Modern Cryptography</i> (1st ed. 2007) [SERVED, NOT FILED]
2030	Reserved
2031	Ross Anderson, <i>Security Engineering: A Guide to Building Dependable Distributed Systems</i> (1st ed. 2001) [SERVED, NOT FILED]
2032	Bruce Schneier, <i>Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C</i> (2nd ed. 2015) (E-Book) [SERVED, NOT FILED]
2033	Second Declaration of Aviel D. Rubin, Ph.D. [Rubin-2 nd -Decl.]
2034	Deposition Transcript of Dr. Samrat Bhattacharjee [Bhattacharjee-Depo.]
2035	Reserved
2036	J. Hur et al., <i>Packet Chunking for File Detection</i> , IEEE/ACM Transactions on Networking Vol. 31 (2023). [Hur]
2037	R. Li et al., <i>A Framework for Qualitative Communications Using Big Packet Protocol</i> , 2019, https://doi.org/10.48550/arXiv.1906.10766 . [Li-I]
2038	N. McKeown et al., <i>The Tiny Tera: A Packet Switch Core</i> , IEEE Micro 17, 1998. [McKeown]

2039	Reserved
2040	Reserved
2041	R. Li et al., <i>New IP: A Data Packet Framework to Evolve the Internet</i> , 2020, https://www.researchgate.net/publication/341582478 [Li-II]
2042	Vimal Kumar et al., <i>POP: Privacy and Integrity Preserving Data Aggregation in Wireless Sensor Networks</i> , PROCEEDINGS OF THE IEEE SYMPOSIUM ON RELIABLE DISTRIBUTED SYSTEMS, September 2013 [Kumar]
2043	Yuxin Mao et al., <i>A Feedback-Based Secure Path Approach for Wireless Sensor Network Data Collection</i> , SENSORS, October 22, 2010 [Mao]
2044	Institution Decision in IPR2024-00215 [215-ID]

I. INTRODUCTION

Petitioner fails to prove that any of the challenged claims are invalid. As the Board has twice found, Petitioner fails to show that Orsini anticipates or renders obvious the claims. *See* Section II. Petitioner also fails to show that Foster anticipates or renders obvious the claims. *See* Section III.

II. AS THE BOARD HAS TWICE FOUND, THE ORSINI GROUND FAILS (ALL CLAIMS, GROUND 1).

Petitioner’s first ground (anticipation/obviousness over Orsini) has been twice rejected by the Board. In its institution decisions (“IDs”) both here and in IPR2024-00215, the Board found that “Petitioner has not established a reasonable likelihood that it would prevail in showing that any of claims 1–10 is unpatentable as anticipated by or obvious over [Orsini].” ID, 33; Ex. 2044 [215-ID] 39 (similar). Even prior to the IDs, the claims issued over Orsini in prosecution. *See* Section II.A. Petitioner’s anticipation (Section II.B) and obviousness (Section II.C) arguments again fail.

A. The ’609 Patent Claims Fundamentally Different Technology Than That Disclosed In Orsini.

Orsini is Patent Owner’s application. Figures 1-25 and paragraphs 1-423 of Orsini are included in the ’609’s specification. The ’609, however, includes ***17 new figures and over eighteen columns of additional material*** (~10,500 words) which contain significantly different technology than that disclosed in Orsini. *See*

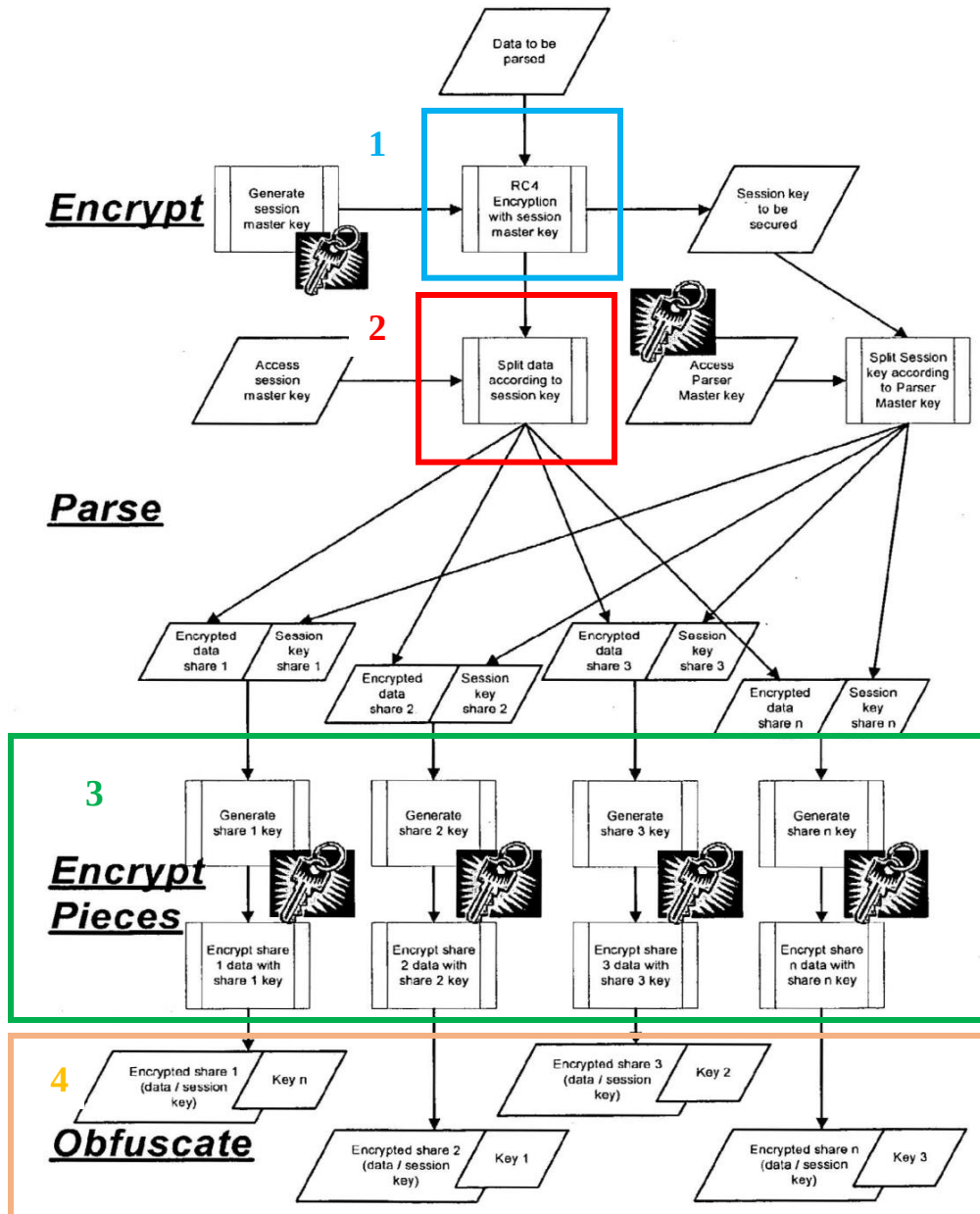
Ex. 1001 ['609] Figs. 26-41, 6:44-6:62, 8:10-8:61, 65:56-83:3. The claims-at-issue rely on this new matter.

Petitioner's Ground 1 relies, almost entirely, on Orsini's Figure 21. *E.g.*, Pet., 9-16, 18, 27-28, 29, 43. Orsini's Figure 21 and the '609's claim 1 are each directed to methods for securing a "dataset." Both methods rely, in part, on dividing the dataset into pieces (called "chunks" or "shares" in the '609) and encrypting the chunks.

From here, however, the methods diverge. As Dr. Aviel Rubin explains, Figure 21's method of securing data is centered around encrypting the dataset with a key called the "session master key," parsing the dataset and the session master key into shares, applying a share key to each of the shares, and then non-cryptographically storing the share keys with a share it was not used to encrypt:

As can be seen below, Figure 21 teaches **(1) encrypting the entire data set with the session master key**, **(2) parsing the data set and session master key into shares**, then **(3) encrypting those shares and split session master key with a set of share keys**, followed by a **(4) non-cryptographic** step:

Figure 21



Ex. 1005 [Orsini] Fig. 21 (annotated), [0293].

Ex. 2033 [Rubin-2nd-Decl.] ¶ 36.

The '609 discloses and claims a materially different and improved method for securing a dataset from that disclosed in Orsini that addresses a potential weakness in Figure 21's method. Ex. 2033 [Rubin-2nd-Decl.] ¶ 42. The '609's innovation is embodied in claim 1's limitation requiring "performing a cryptographic operation based on the first key to further secure the plurality of data chunks." *Id.*, ¶ 41. Rather than non-cryptographically securing the keys, a "first" key (which corresponds to the "workgroup key" of the '609's disclosure) further **cryptographically** secures the shares/chunks. This step, which is neither disclosed nor suggested in Orsini, is rooted in the inventors' insight that Orsini's method potentially allows an attacker to reconstruct the dataset if the attacker has access to each of the shares; the '609's claimed inventions eliminate that potential vulnerability. *Id.*, ¶¶ 39, 42.

Beyond being different, Figure 21's method is simply incompatible with claim 1 and thus a POSITA would have no motivation or reasonable expectation of success in modifying Orsini to achieve claim 1. Petitioner equates the claimed "first key" (*i.e.*, the workgroup key) with the "session master key" of Orsini's Figure 21. Claim 1 requires the generation of data chunks from a data set, and that the data chunks must first be encrypted with a respective second key before the plurality of data chunks is "**further** secur[ed]" with a "**cryptographic** operation based on the first key." Pet., 15, 30-36. Petitioner contends that a POSITA would

modify Orsini to use the session master key to “further secure” the dataset after encrypting the shares. But, as discussed, in Figure 21, prior to this “further securing” step, the session master key was already divided into pieces, the pieces were integrated into the shares, and the session master key/share combinations (each containing a different piece of the session master key) were encrypted with share keys. Thus, by the time the dataset is “further secured” as required by claim 1, the session master key in Orsini is buried within encrypted shares, and any further use of it would entirely negate the prior steps in Orsini’s method. Ex. 2033 [Rubin-2nd-Decl.] ¶ 43.¹ Critically, to use the already parsed and encrypted session master key in a “cryptographic operation” to “further secure the plurality of data chunks” as Petitioner suggests, a priori knowledge of the session master key would

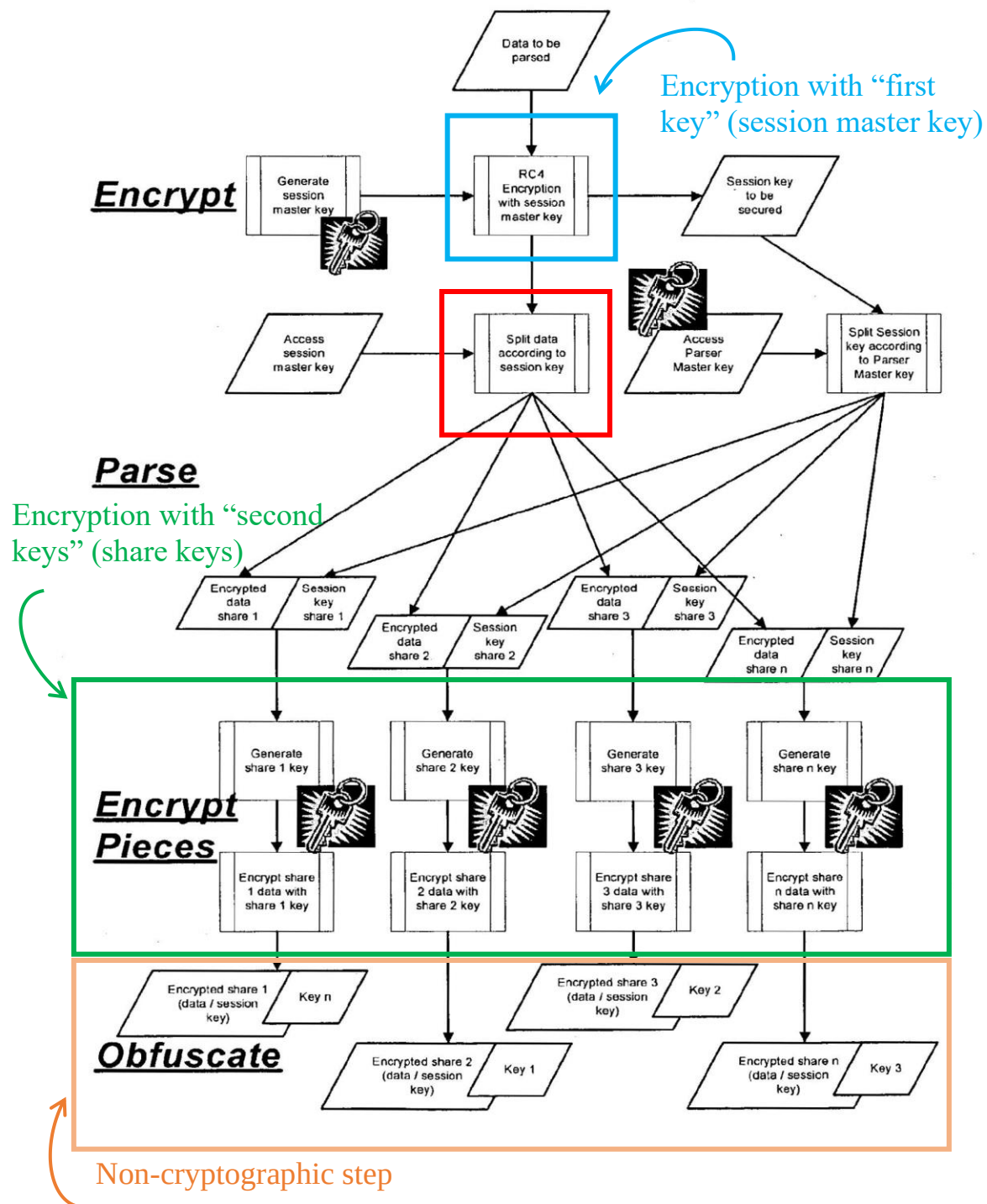
¹ There is no dispute that these steps must be performed in the claimed order. Ex. 1064 [Google-Proposed-Claim-Construction] 16 (claim 1 steps “must be performed in order”); *see also Mantech Envtl. Corp. v. Hudson Envtl. Servs.*, 152 F.3d 1368, 1375–76 (Fed. Cir. 1998) (“steps must be performed in order” where “the sequential nature of the claim steps is apparent from the plain meaning of the claim language”). Here, without being already “encrypt[ed] ... with a respective second key,” the plurality of data chunks could not be “**further** secure[d]” with a “cryptographic operation based on the first key.” Ex. 2033 [Rubin-2nd-Decl.] ¶ 45.

be needed to unpack the encrypted shares. This would require complex and undisclosed changes to Figure 21's method due to the need for separate storage of the session master key and would render entirely unnecessary the steps Figure 21 does disclose to parse and store the session master key within the encrypted shares. *Id.*; ID, 44 (“both versions of Petitioner’s Two-Encrypt-Phases rationale go beyond mere repetition.”); *id.* (“We find Dr. Rubin’s testimony (Ex. 2003 ¶ 42) more persuasive than Dr. Bhattacharjee’s (Ex. 1003 ¶¶ 106-108) at least because it is better supported by Orsini’s Figure 21.”).

B. The Board Correctly Found That Petitioner Fails To Show That Orsini Anticipates The Claims.

As the Board correctly found, Petitioner contends that Orsini anticipates claim 1 but fails to show that Orsini expressly or inherently discloses the required ordering. ID, 35-44. As Dr. Rubin explains:

Assuming, as Petitioner does (*e.g.*, Pet., 11-12, 18, 30-31) that Orsini’s “session master key” corresponds to the claimed “first key,” Figure 21 undisputedly teaches **a cryptographic operation performed on the entire data set with the session master key** (a “first key”) **before parsing the data set and session master key into shares** and **encrypting those shares and split session master key with a set of share keys**. And any additional protection of the data chunks is achieved through a **non-cryptographic step**:

Figure 21

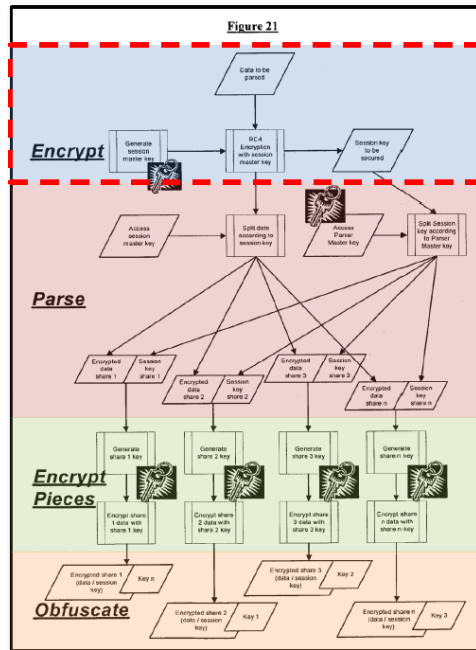
Ex. 1006 [Foster] Fig. 21 (annotated).

Orsini, however, does not disclose performing any “cryptographic operation . . . to further secure the plurality of data chunks” “based on the first [session master] key.” Indeed, Orsini ***cannot disclose this***, because in Figure 21, after the session master key/first key is used to encrypt the dataset as a whole, it is “separat[ed]” along with the data set and those “portions of the session master key” are “contain[ed]” in the chunks/shares of data which are then encrypted. Ex. 1005 [Orsini] [0293], [0297]-[0299]; *compare id. with* Ex. 1001 [’609] 80:24-27 (using workgroup key instead of session master key). Consequently, by the time the data set has been parsed into shares and those shares have been encrypted, the session master key (*i.e.*, the alleged “first key”) cannot be used for a cryptographic operation without removing the benefits provided from Orsini’s splitting and encrypting of that key.

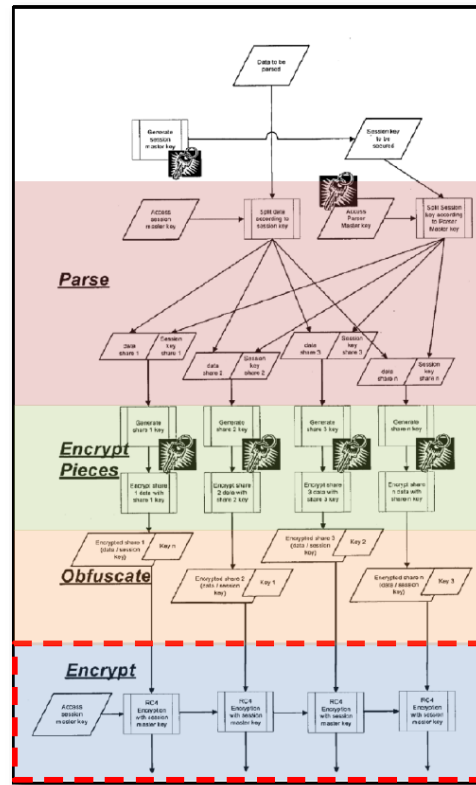
Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 48-49.

Rather than pointing to any actual disclosure in Orsini, Petitioner is forced to ***alter*** Figure 21 to add another, undisclosed, chunk encryption step, this time using the session master key, after the data chunks are generated and encrypted a first time with respective second keys as shown in Petitioner’s figures below:

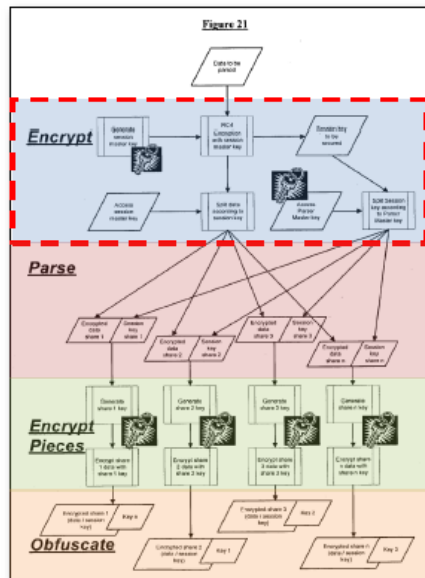
Original



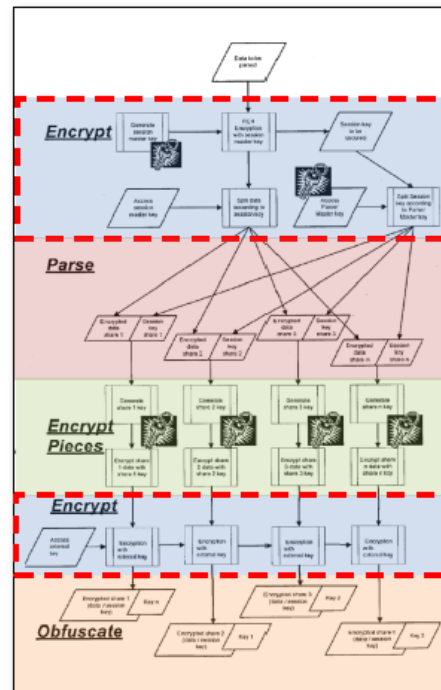
Different-Order Version



Original

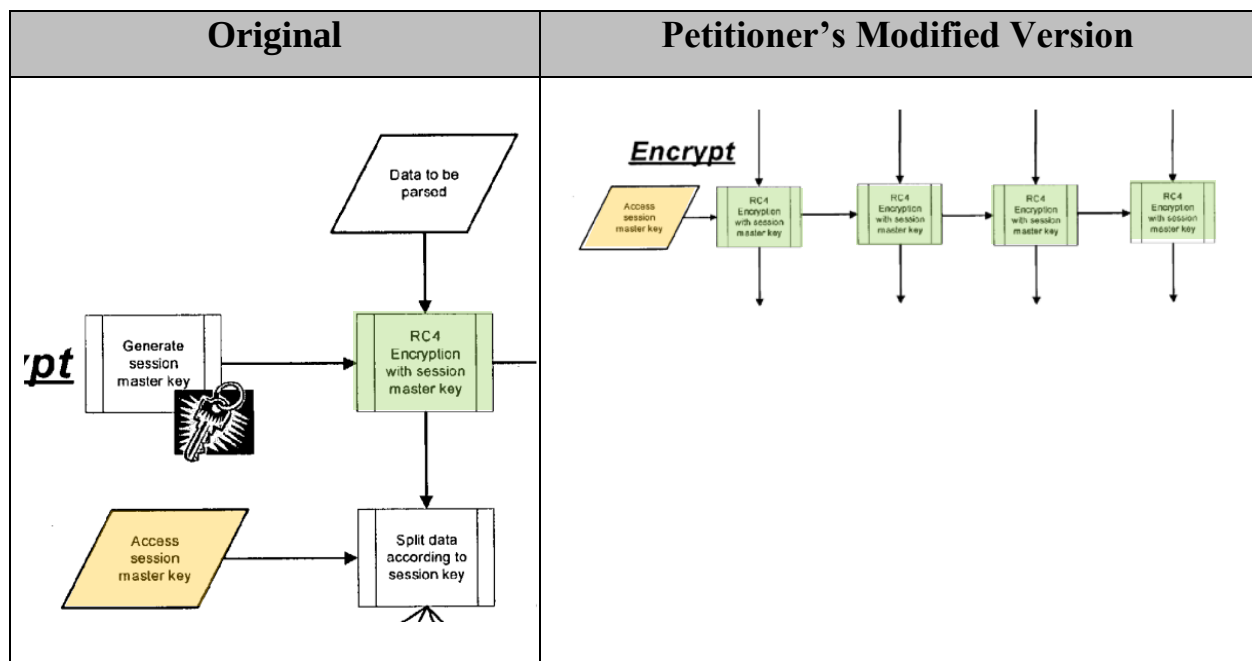


Two "Encrypt" Phases (Version 1)



Pet., 16, 35; *see also id.*, 32-33, 36.

Not only does Petitioner, without explanation, move the dataset encryption step from the beginning of Figure 21 to the end, but it also then entirely re-writes the step (again, without explanation) **to apply to each of the chunks rather than to the dataset as a whole**. Thus, Petitioner does not merely reorder Figure 21, it fundamentally alters what is being encrypted (the dataset versus the chunks/shares). Petitioner additionally moves the step **wherein the session master key is accessed from prior to the data being parsed to after the non-cryptographic step**. Ex. 2033 [Rubin-2nd-Decl.] ¶ 51.



As the Board found, “[u]nlike Orsini’s Original Figure 21 (left), Petitioner’s Different-Order Versions lack an initial encrypt step. Instead, there is an

encryption step after either the ‘obfuscate’ step or the ‘encrypt pieces’ step.” ID, 37 (cleaned up).

Petitioner and Dr. Bhattacharjee’s reliance upon materially altered versions of Figure 21 establishes as a matter of law that Orsini does not anticipate Claim 1. *Enplas Display Device Corp. v. Seoul Semiconductor Co., Ltd.*, 909 F.3d 398, 405 (Fed. Cir. 2018) (“Prior art that must be modified to meet the disputed claim limitation does not anticipate the claim.”). It was for this reason that the Board correctly found that Petitioner had not shown that Orsini expressly discloses the claimed inventions. ID, 38 (“Rather, we preliminarily agree with Patent Owner that Petitioner has not shown that Orsini anticipates claim 1 based on the Different-Order Versions.”); 40 (“Petitioner’s Two-Encrypt-Phases rationale also is deficient for reasons similar to those for the Different-Order rationale: Petitioner does not sufficiently explain or support its arguments about Orsini’s session master key and common key”).

1. Orsini Does Not Inherently Disclose Claim 1.

Lacking an express disclosure of the “further securing” step, Petitioner, nonetheless, contends that Orsini suggests the step to a POSITA in its statement that “[i]t is readily apparent to those of ordinary skill in the art that certain steps of the methods described herein may be performed in different order, or repeated multiple times, as desired.” Pet., 30-36 (citing Ex. 1005 [Orsini] [0301]). But,

such boilerplate language is not sufficient to show actual disclosure of a claim limitation, for “[a] [prior art] patent discloses only that which it describes” *In re Benno*, 768 F.2d 1340, 1346 (Fed. Cir. 1985) (reversing obviousness rejection); *cf. Forum US, Inc. v. Flow Valve, Ltd. Liab. Co.*, 926 F.3d 1346, 1352 (Fed. Cir. 2019) (“the boilerplate language that modifications can be made to the original disclosed invention does not even suggest an arbor-less embodiment of the disclosed invention.”); *D Three Enters., LLC v. SunModo Corp.*, 890 F.3d 1042, 1050-51 (Fed. Cir. 2018) (similar).

Equally important, Petitioner does not identify anything in Orsini that could be “performed in a different order” or “repeated” to further secure the **chunks** (as required by claim 1), in contrast to the dataset as a whole. No matter how many times one repeats the step of encrypting **the original dataset** using the session master key, and regardless of the order in which that step is carried out, it cannot disclose encrypting **the chunks**. Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 52, 55.

Instead, Petitioner asserts that a POSITA would understand that its proposed alterations of Figure 21 were taught by Orsini. Petitioner claims that “POSAs understood that one ‘certain step[]’ capable of being reordered was the first ‘encrypt’ step with the session master key in Orsini’s Figure 21 [] because it was well-known to reorder whether an encryption step is performed before and/or after splitting.” Pet., 30-31 (citing Ex. 1003 [Bhattacharjee-Decl.] ¶¶ 102-103).

It is black-letter law that to anticipate, a reference must disclose each limitation either expressly or inherently. An anticipation argument based on what POSITA might “understand” must meet the requirements for an inherent disclosure. *Microsoft Corp. v. Biscotti, Inc.*, 878 F.3d 1052, 1065 (Fed. Cir. 2017) (statements that “a [POSITA] would understand the [evidence] as disclosing [the limitation]” could support an argument that a limitation was “inherently disclose[d] ...”); *LBT IP I LLC v. Apple Inc.*, 2023 U.S. App. LEXIS 14376, *8 (Fed. Cir. 2023) (similar). Inherency requires demonstrating that a limitation is “‘necessarily present,’ not merely probably or possibly present, in the prior art.” *Trintec Indus., Inc. v. Top U.S.A. Corp.*, 295 F.3d 1292, 1295 (Fed. Cir. 2002); *PAR Pharm. v. TWi Pharms., Inc.*, 773 F.3d 1186, 1195–96 (Fed. Cir. 2014) (similar). Petitioner, however, does not even sufficiently show a POSITA’s understanding, much less that Orsini “necessarily” discloses performing a cryptographic operation to further secure the chunks based on the first/session master key, as required by claim 1.

As to a POSITA’s understanding, Petitioner provides only a conclusory statement with a single citation to its expert declaration, no explanation or analysis, and a string of unexplained citations in its expert’s declaration, none of which is sufficient to show anticipation. See Pet., 30-31; 37 C.F.R. § 42.6(a)(3); *Schumer v. Lab. Computer Sys., Inc.*, 308 F.3d 1304, 1315-16 (Fed. Cir. 2002) (conclusory testimony is “insufficient”). In accord, the Board found that Petitioner’s

conclusions as to the Two-Encrypt-Phases version are insufficiently supported.

ID, 40 (“Petitioner does not identify anything from Orsini that explains how the algorithms are different. ... For at least this reason, we agree with Patent Owner that Petitioner does not sufficiently support its reasoning about repeating Orsini’s steps.”).

But even if it were considered, Orsini’s statement that “certain steps may be performed in different order, or repeated multiple times” does not remotely ***necessarily disclose*** “performing a cryptographic operation based on the first key to ***further secure the plurality of data chunks.***” That “certain” steps can be repeated does not mean that the dataset-encryption step would necessarily be repeated, and certainly does not establish that the chunks, each encrypted with a separate share key, would ***necessarily*** be again encrypted using the session master key, as required by Petitioner’s argument. As Dr. Rubin explains:

Orsini, moreover, only specifies that “***certain*** steps” “may be performed in different order.” Ex. 1005 [Orsini] [0301]. Orsini does not teach that the encryption of the data set step is one such step. Rather, Orsini discloses that “[f]or example, multiple parsing steps may be performed on only one portion of the parsed data,” and that “[e]ach portion of parsed data may be uniquely secured.” *Id.* Orsini, therefore, only identifies the parsing and encryption with the set of second keys as steps which may be repeated or reordered, and on “only one portion of the parsed data” at a time at that. There is simply no disclosure that

would necessitate that the data chunks, after already being individually encrypted with the alleged set of second keys, would be encrypted again, let alone with the alleged first key, or that the “plurality of data chunks” rather than an individual data chunk would be encrypted based on the first key. Indeed, a POSITA would recognize that encryption with the alleged first key would not be an appropriate step to repeat or reorder because it is parsed along with the data set into portions and those portions are then encrypted. Consequently, encrypting the shares with the alleged first key (*i.e.*, the session master key) would render the splitting and encryption of the alleged first key pointless.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 54.

The Board agreed that Petitioner’s “Different-Order Version” and “Two-Encrypt-Phases version” go “beyond mere” “reordering” and “repetition,” respectively, and fundamentally change Figure 21. ID, 37-39, 43-44. The Board “f[ou]nd Dr. Rubin’s testimony ... more persuasive than Dr. Bhattacharjee’s ... on th[ese] issue[s]” and correctly found that “Petitioner has not shown that Orsini discloses” the Different-Order or Two-Encrypt-Phases versions. *Id.*

2. Petitioner’s Other Arguments Cannot Establish Anticipation Of Claim 1.

Petitioner’s expert cites a number of references for the generic proposition that it was known to encrypt data before and/or after fragmenting a dataset but does not contend that its invalidity grounds are based on these references. As such, they are simply irrelevant as to what **Orsini** necessarily discloses. Orsini does not

disclose, implicitly or expressly, “further securing” “data chunks” via a “cryptographic operation” based on a “first key.” See Ex. 1003 [Bhattacharjee-Decl.] ¶ 103 (citing Exs. 1028, 1053, and 1045).

Finally, Petitioner points to certain dependent claims in Orsini that claim encrypting shares/chunks a second time using “a different encryption algorithm.” Pet., 33-34 (citing Ex. 1005 [Orsini] cls. 12, 29, 39). But these claims do not expressly or inherently disclose claim 1’s limitations. As Dr. Rubin explains:

Dependent claims 12, 29, and 39, directed to repeating encryption of shares with a “different encryption algorithm,” do not disclose claim 1, which requires that the encrypted data shares undergo a “cryptographic operation ***based on the first key*** to further secure the plurality of data chunks.” Simply repeating encryption using a “different algorithm” does not even suggest, much less necessarily disclose, that the “different algorithm” would be based on the session master key. See Ex. 1005 [Orsini] [0071] (explaining “a wide number of algorithms [are] available”).

Ex. 2033 [Rubin-2nd-Decl.] ¶ 57; ID, 43 (cited claims “provide little guidance on how the different encryption algorithm works.”).

The dependent claims cited are, moreover, different embodiments from Figure 21. Ex. 2033 [Rubin-2nd-Decl.] ¶ 58. Petitioner may not “mix and match” features from different unconnected embodiments to anticipate claim 1. “[I]t is not enough that the prior art reference . . . includes multiple, distinct teachings that

the artisan might somehow combine to achieve the claimed invention’ to show anticipation.” *BlephEx, LLC v. Myco Indus.*, 24 F.4th 1391, 1400 (Fed. Cir. 2022) (citations omitted); *Net MoneyIN, Inc. v. VeriSign, Inc.*, 545 F.3d 1359, 1371 (Fed. Cir. 2008) (anticipation requires “the limitations [be] arranged or combined in the same way as recited in the claim”); *Finisar Corp. v. DirecTV Group, Inc.*, 523 F.3d 1323, 1334-35 (Fed. Cir. 2008) (similar). As a matter of law, the dependent claims cannot show that the Figure 21 embodiment on which Petitioner relies discloses, expressly or inherently, claim 1’s invention.

C. Petitioner Fails To Show That Orsini Renders The Claims Obvious.

Petitioner alternatively contends that Claim 1 is obvious over Orsini alone, relying heavily upon the purported common sense of a POSITA. As the Board correctly found, Petitioner’s arguments are “unsupported and conclusory,” “circular,” and “based on impermissible hindsight.” ID, 46-47; *id.* (“not credit[ing] Dr. Bhattacharjee’s testimony [] on this issue.”); 48-49 (similar for “[r]e-encrypting with a group key”); 49-50 (similar for “[r]eordering data”); 50-51 (similar for “Encrypting the Distinct Encryption Keys with a Key-Encryption Key”). If more is needed, Petitioner’s argument fails for the additional reasons stated below.

1. Petitioner Cannot Rely On Common Sense To Supply Missing Limitation 1[e].

Instead of combining Orsini with another reference, Petitioner contends that a POSITA “had multiple reasons to implement Orsini to perform” limitation 1[e] (“performing a cryptographic operation based on the first key to further secure the plurality of data chunks”). Pet., 36; 36-42 (describing allegedly “known” cryptographic operations and asserting POSAs “had reason” to implement them in Orsini’s Figure 21 method).

The Federal Circuit has held that “common sense” can be used to supply a missing limitation “**only** when ‘the limitation in question [is] unusually simple and the technology particularly straightforward’”—something that is emphatically not the case here. *DSS Tech. Mgm’t v. Apple Inc.*, 885 F.3d 1367, 1374-75 (Fed. Cir. 2018) (citations omitted). And where, as here, a “missing limitation goes to the heart of an invention” a court’s review of any purported “reasoned basis for resort to common sense” must be especially “searching” *Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1362-63 (Fed. Cir. 2016).

In this case, the limitation-at-issue is both at the heart of the invention and involves complex technology, so it is not appropriate to supply it through “common sense.” As Dr. Rubin explains:

The inventions-at-issue concern cryptography, which is a technologically-involved and complex field involving incredibly high

stakes and safeguarding against a wide variety of highly sophisticated and ever-evolving cybersecurity attacks. ...

And, as previously discussed, the '609's new teachings and the claimed inventions arose from the inventor's insight regarding a potential vulnerability in Orsini's Figure 21 approach wherein if an attacker obtains each of the shares, the dataset can be reconstructed. The '609 eliminates this vulnerability by encrypting the encryption keys using a workgroup key—corresponding to the “first key” of claim 1—a technique not disclosed in Orsini. The workgroup key provides “enhanced protection of information assets by encrypting key information stored within the shares” (Ex. 1001 ['609] 79:60-62) adding “an extra level of security to a secured data set” (*id.*, 77:10-14). As the '609 explains “even if all required shares and external keys are discovered, an attacker has no hope of recreating the information without access to the workgroup key.” *Id.*, 79:62-65. Thus, this limitation goes to the heart of the invention. Moreover, given the benefits associated with the “workgroup key” delineated in the '609, it further seems clear that the inventors of the '609 (and Orsini) considered the “workgroup key” concept to be a fundamentally different approach that solved a problem that Orsini did not recognize or address.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 61-62.

Consequently, Petitioner's attempt to rely solely on a POSITA's common sense to add the missing limitation to Orsini should be rejected.

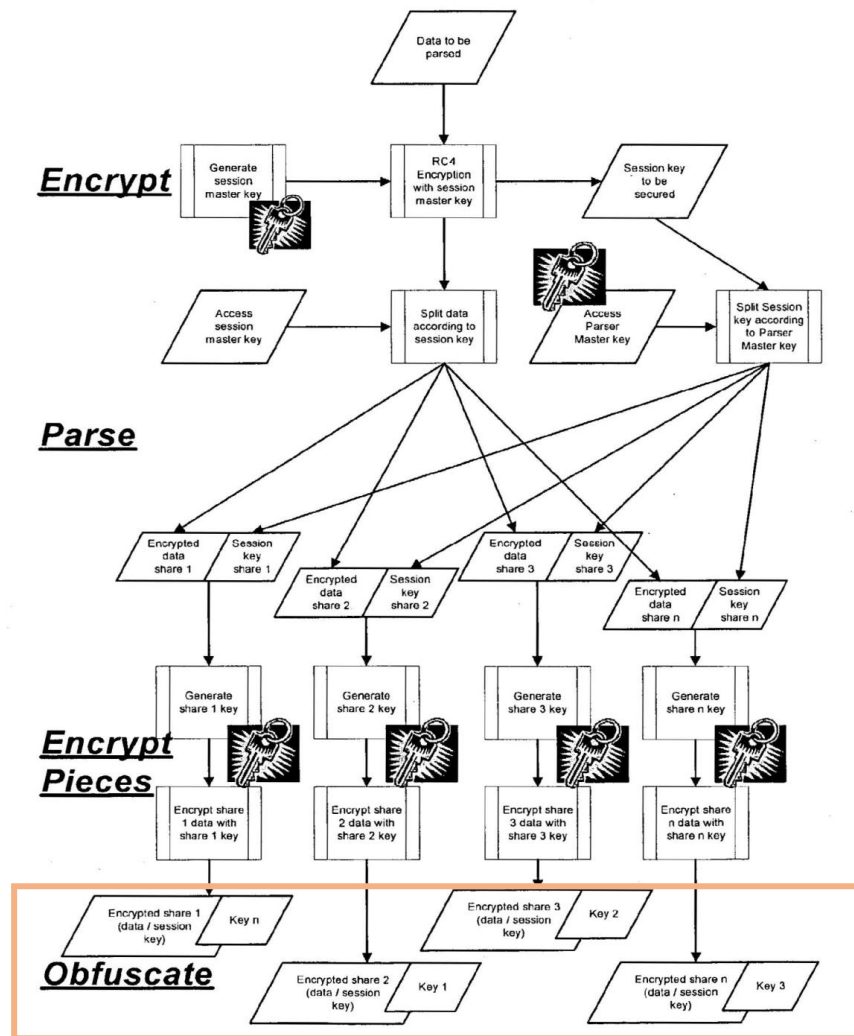
2. Petitioner Fails To Show That A POSITA Would Have Been Motivated To Modify Orsini to Add Limitation 1[e].

As discussed, like claim 1, Orsini includes the steps of generating and encrypting the chunks but does not disclose, expressly or inherently, limitation 1[e]: “performing a cryptographic operation based on the first key to further secure the plurality of data chunks.” See Section II.A. Petitioner suggests that the limitation is obvious because “POSAs would have been motivated by [Orsini’s] disclosures” to add this limitation because they “would have understood that doing so would have provided additional security and efficiency benefits.” Pet., 36-41. Petitioner, however, fails to demonstrate that a POSITA would be motivated to modify Orsini to “perform[] a cryptographic operation based on the first key to further secure the plurality of data chunks.”

Petitioner’s purported motivation should be rejected because adding a “cryptographic operation based on the first key” to Orsini is useless because, unlike the ’609’s claim 1, Orsini **already** uses a **non**-cryptographic operation that is **not based on any key** to further secure the data chunks. Ex. 1005 [Orsini] Fig. 21, [0299]. As Dr. Rubin explains:

Orsini already teaches hiding the encryption key used to encrypt each data chunk with a different chunk through a **non-cryptographic process**, illustrated in Figure 21 below.

Figure 21



Ex. 1005 [Orsini] Fig. 21. As can be seen in Figure 21, in this step, the system “store[s] the encryption keys in different locations from the encrypted data portions or shares ...” *Id.*, [0299]; see also [0353] ...; [0396]; cl. 26.

As Petitioner does not dispute, the step taught in Orsini’s Figure 21 is not a cryptographic operation (much less one involving the first key). See generally *Pet.*; Ex. 1003 [Bhattacharjee-Decl.]. As there is no encryption scheme or “scrambling” or altering of the data or the

alleged respective second keys (Ex. 2007 [Katz] 4), it cannot be a cryptographic operation.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 64-65.

Petitioner fails to explain why a POSITA would be motivated to modify Orsini using other allegedly “known” methods, particularly when it non-cryptographically hides each share key with a share it did not encrypt and teaches that its method provides a “**comprehensive package for data security**.”² Ex. 1005 [Orsini] [0380]; Ex. 2033 [Rubin-2nd-Decl.] ¶ 66.

As a matter of law, it is not obvious to modify a reference that already achieves the benefit supposedly served by the proposed modification. *See, e.g., In re NTP, Inc.*, 654 F.3d 1279, 1299 (Fed. Cir. 2011) (reversing IPR obviousness rejections as “based on improper hindsight reasoning” when “the [prior art] already discloses an RF network that connects portable computers to the system. Thus, adding an RF network [in the way necessary to meet the claims] would render the RF network [in the prior art] superfluous.”); *Apple Inc. v. Cellular Comm’n’s Equipment, LLC*, IPR2015-00576, Paper 7, 12 (June 12, 2015) (denying institution where “Petitioner ... fails to provide a specific explanation why [the primary reference] did not already accomplish [the asserted benefits] by itself.”); *F5*

² All emphasis added unless otherwise noted.

Networks, Inc. v. Radware, Ltd., IPR2017-00124, Paper 48, 64-65 (Apr. 23, 2018) (no motivation for modification that “serve[s] no purpose”).

To the extent Petitioner argues that a motivation to modify Orsini is found in the purported knowledge of a POSITA that “when ‘encryption keys are included with the encrypted message’ ‘[s]ecurity’ comes from ‘encrypt[ing] the encryption keys themselves’ with ‘a key encryption key.’” Pet., 40 (citing Exs. 1035, 1036), that concern is addressed in Orsini’s non-cryptographic step. In this step, a POSITA would understand each share key is secured not by encryption, but by hiding it “in different locations from the encrypted data portions or shares” that particular share key was used to encrypt. Ex. 1005 [Orsini] [0299]. Orsini teaches that this is sufficient because an attacker would not have “enough information to decrypt the” data chunk stored in each data depository. *Id.*, [0353]; Ex. 2033 [Rubin-2nd-Decl.] ¶ 67. Thus, in Figure 21’s data-securing method, the session master key, data chunks, and session keys are all either encrypted, parsed, and/or hidden, and thus each potentially vulnerable part of the encryption method is secured in some way. Ex. 2033 [Rubin-2nd-Decl.] ¶ 67.

In addition to its failure to demonstrate that its proposed modifications add any benefit to Orsini, Petitioner also fails to consider their drawbacks. “The Board must weigh the benefits and drawbacks of the modification against each other, to determine whether there would be a motivation to combine.” *Arctic Cat Inc. v.*

Polaris Indus., 795 Fed. App'x. 827, 833 (Fed. Cir. 2019); *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1331-32 (Fed. Cir. 2019) (similar); *Arctic Cat Inc. v. Bombardier Rec. Prods.*, 876 F.3d 1350, 1363 (Fed. Cir. 2017) (“potential problems” with combination coupled with “an alternative approach [that] might reduce those same problems,” supported finding that “a skilled artisan would not have been motivated” to make combination).

The drawbacks of Petitioner’s proposed modification of Orsini are considerable. As Dr. Rubin explains:

Simply concluding that adding an encryption layer would “benefit” Orsini’s method oversimplifies the myriad considerations a POSITA must make in cryptography, even for what may seem to be a “simple” or “small” change. For example, a POSITA would recognize that when fully considering the costs and benefits, an additional layer of encryption may not in reality benefit an encryption scheme or “further secure the plurality of the data chunks” as the claims require. Rather, an additional encryption step may in fact introduce a weakness that an attacker can exploit. *See, e.g.*, Ex. 2007 [Katz] 22 ...; Ex. 2008 [Schneier-Security-Pitfalls-in-Cryptography] ...; Ex. 2009 [Anderson] One such weakness is in the critical area of key management, as it is where systems tend to be most vulnerable. *See* Ex. 2010 [Schneier-Applied-Cryptography] 169

Thus, while adding an additional round of encryption with the same or additional keys may appear to provide improved security, doing so may, on balance, compromise security.

Petitioner's proposal to "re-encrypt data chunks with a first encryption key" (Pet., 36-37) has significant security flaws. Ex. 2010 [Schneier-Applied-Cryptography] 357 ("Encrypting a plaintext block twice with the same key, whether with the same algorithm or a different one, is not smart."). If Orsini were modified such that the session master key (the alleged "first key") were either reapplied or otherwise utilized after the data has been split, that would undermine Orsini's method which protects the alleged first key by splitting it amongst the data shares. And it would entail further complications. By involving the session master key after the splitting process, Petitioner's proposal would necessitate the key management and transport of the session master key, effectively making the division and inclusion of the session master key within the encrypted shares pointless. Under Orsini's existing method, the master session key has already been parsed among the data shares and encrypted. Consequently, the session master key which was previously secured in Figure 21 would be unnecessarily exposed in this proposed modification. ...

Ex. 2033 [Rubin-1nd-Decl.] ¶¶ 69-71.

To the extent Petitioner proposes that a POSITA would be motivated to modify Orsini's Figure 21 to use another key, such as a "group key," "encryption key," or "key-encryption key" (Pet., 37-41), Petitioner too fails to address the

drawbacks associated with managing and securing such keys against discovery by an attacker. Per Dr. Rubin:

Petitioner and Dr. Bhattacharjee argue that “Orsini provides ‘**session** key management’” and discloses a “single point of . . . key management” that could be modified to “centrally generat[e] Orsini’s session master keys (***and any of the conventional encryption keys [used in Petitioner’s obviousness argument]***).” Pet., 23-24 (citing Ex. 1005 [Orsini] [0176]-[0179], [0293], [0381]-[0386], [0412]). But **session** key management is very different than key management for longer-lived keys such as the other “conventional encryption keys” Petitioner relies upon, especially as Figure 21’s session keys (the session master key and share keys) are stored alongside the encrypted data. For example, “session keys should be changed at least daily” if not for each “particular communications session” while “key-encryption keys don’t have to be replaced as frequently,” perhaps “once a year.” Ex. 2010 [Schneier-Applied-Cryptography] 184, 47; *see also id.*, 184

This longer lifespan, however, makes potential compromise of a key-encryption key all the more consequential. *Id.* (“However, if a key-encryption key is compromised, the potential loss is extreme: all communications encrypted with every key encrypted with the key-encryption key.”). Consequently, a POSITA would recognize that a key management solution for session keys would not be appropriate for the “group key,” “encryption key,” or “key-encryption key” used in Petitioner’s alternative proposed modifications. *See* Ex. 1001 [’609]

62:22-23 (parser master, session master, and share encryption “keys may have individual key storage, retrieval, security and recovery options.”).

Moreover, key management for a “group key,” “encryption key,” or “key-encryption key” would necessarily be different than for key management for the “session” keys in Figure 21. In Figure 21, the session master key used to encrypt the dataset is parsed within the data shares (Ex. 1005 [Orsini] [0297]) and the session share keys are stored along with the data shares (*id.*, [0299]). Therefore, so long as the party reconstructing the dataset possesses the shares, he or she also possesses the session keys necessary to decrypt the data. By contrast, the “group key,” “encryption key,” or “key-encryption key” would not be stored alongside the data but would rather need to be distributed out-of-band to the decrypting party. Consequently, those keys would need to be maintained and stored on both ends and in transit such that no adversary gets access to them. Thus, the threat model for the “group key,” “encryption key,” or “key-encryption key” is very different than for a “session key.” Petitioner and Dr. Bhattacharjee have not mentioned, let alone adequately addressed, these different threat models. Their superficial consideration of key management, an incredibly important aspect of cryptography, is thus insufficient and flawed.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 73-75.

Petitioner attempts to justify its failure to properly consider the lack of benefits and substantial drawbacks of its proposed modification of Orsini by

claiming that it is merely applying a “known” method disclosed in a number of other references not included in Ground 1. Pet., 36-42. The Board properly rejected this argument in its ID, finding that Petitioner failed to “explain how the cited prior art would have been used to modify Orsini” and “[t]hus, it is not apparent how Petitioner intends to rely on these references in its challenge.” ID, 48. Although references beyond a Petitioner’s grounds may “document the knowledge that skilled artisans would bring to bear in reading the prior art identified as producing obviousness,” *Ariosa Diagnostics v. Verinata Health, Inc.*, 805 F.3d 1359, 1365 (Fed. Cir. 2015), Petitioner cannot use them to supply a missing limitation, effectively subverting the requirement to properly identify the combinations it relies upon in its Grounds. Nor can it use such references to establish a motivation to modify Orsini. *Virtek Vision Int’l ULC v. Assembly Guidance Sys., Inc.*, 97 F.4th 882, 886-7 (Fed. Cir. 2024) (“We conclude that the Board erred as a matter of law with regard to the motivation to combine. It does not suffice to meet the motivation to combine requirement to recognize that two alternative arrangements ... were both known in the art.”); *RPX Corp. v. Parity Networks, LLC*, IPR2018-00097, Paper 7, 14 (Apr. 24, 2018) (“Petitioner also cites to several other references . . . but does not assert these references are part of the combination. Pet. 34–35. Relying on knowledge in the art does not . . . show a

motivation to combine existing components in the manner claimed free of hindsight motivation.”).³

In the end, as the Board found, Petitioner’s proposed modification of Orsini and its purported motivation to make that modification are entirely hindsight-driven, based on its knowledge today of what the inventors disclosed in the ’609 and claimed in claim 1. ID, 47 (“we also agree with Patent Owner that Petitioner’s reason to modify Orsini is based on impermissible hindsight.”), 48-49 (similar), 49-50 (similar), 50-51 (similar). It is well-settled that “a patent composed of several elements is not proved obvious merely by demonstrating,” as Petitioner tries to do here, “that each of its elements was, independently, known in the prior art.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418–19 (2007). Rather,

³ Petitioner cites *Intel Corp. v. PACT XPP Schweiz AG*, 61 F.4th 1373, 1380 (Fed. Cir. 2023) for the proposition that “if there’s a known technique to address a known problem using prior art elements according to their established functions, then there is a **motivation to combine**,” (Pet., 37) but, it is inapposite because Petitioner does not even attempt to show a motivation to combine Orsini with any of these references. *Edwards Lifesciences Corp. v. Aortic Innovations LLC*, IPR2023-01232, Paper 10, 36 (Feb. 7, 2024) (“Petitioner’s rationale to modify Spenser II based on inchoate and **unasserted combinations** is also unpersuasive.”)

Petitioner must “identify **a reason** that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does.” *Id.* And this it has failed to do. As Dr. Rubin explains:

Petitioner’s proffered modifications do not seem to create any advantage for Orsini’s system and, instead, would seem to undermine Orsini’s approach. At the very least, Petitioner’s proffered modifications are inadequately developed. Instead, Petitioner’s modifications ultimately seem to be driven by hindsight. To reiterate, in each instance, Petitioner begins with Orsini’s Figure 21 which sets out a fully functional system (Ex. 1005 [Orsini] [0380]) that Petitioner has not improved upon. Adopting the solution of “performing a cryptographic operation based on the first key to further secure the plurality of data chunks” seems driven by hindsight, not any technological need or benefit that would improve upon Orsini. It seems the claim limitation itself is being used as a roadmap driving the modification.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 76.

Petitioner’s proffered modifications thus provide no benefit, entail significant drawbacks, and are precisely the sort of hindsight-driven analysis the Federal Circuit has rejected. *Ruiz v. A.B. Chance Co.*, 357 F.3d 1270, 1275 (Fed. Cir. 2004) (“This form of hindsight reasoning, using the invention as a roadmap to find its prior art components, would discount the value of combining various

existing features or principles in a new way to achieve a new result - often the very definition of invention.”); ID, 47, 48-49, 49-50, 50-51.

Petitioner’s invocation of using a “group key” or encrypting the “respective second keys” with a “key-encryption key” are likewise hindsight-driven, merely following the ’609’s newly-disclosed inventive concept of encrypting the share encryption keys with a workgroup key. *Compare* Pet., 37-41 with Ex. 1001 [’609] 79:46-65. As the Federal Circuit has held, however, “[t]he inventor’s own path itself never leads to a conclusion of obviousness; that is hindsight.” *Univ. of Strathclyde v. Clear-Vu Lighting LLC*, 17 F.4th 155, 165 (Fed. Cir. 2021) (quoting *Otsuka Pharm. Co., v. Sandoz, Inc.*, 678 F.3d 1280, 1296 (Fed. Cir. 2012)); *InTouch Techs., Inc. v. VGo Commc’ns., Inc.*, 751 F.3d 1327, 1351 (Fed. Cir. 2014) (rejecting expert’s obviousness theory that “relied on the [] patent itself as her roadmap for putting what she referred to as pieces of a ‘jigsaw puzzle’ together.”); *In re Conrad*, 759 Fed. App’x. 982, 985 –86 (Fed. Cir. 2019) (obviousness requires “a specific” rationale to combine limitations to make the claimed invention that is “**unrelated to** the rationale behind the **inventor’s** reasons for making the **[claimed]** invention.”) (collecting cases).

Petitioner, thus, contends a POSITA would be motivated to modify Orsini in a fashion that would yield no benefit identified in the prior art (in view of Orsini storing share keys with shares they were not used to encrypt, which Petitioner

ignores), entails significant drawbacks, is based upon inchoate and nonasserted combinations, and is ultimately driven by using the inventor's own path against him. As the Board found, Petitioner's proffered modification is rank hindsight and should be rejected.

III. THE FOSTER GROUND FAILS (ALL CLAIMS, GROUND 2).

Petitioner's second ground argues the claims are anticipated and/or rendered obvious by Foster. Pet., 3. Foster, however, is a fundamentally different invention from the '609 and concerns protecting content streamed or transmitted from a content owner to a consumer rather than stored data as the '609 does (Section III.A) and, consequently, Petitioner's second ground also fails for multiple reasons (Sections III.B-D).

A. The '609 And Foster Fundamentally Differ.

The '609 and Foster arise in entirely different contexts, are directed towards addressing fundamentally different problems, and result in very different inventions. As Dr. Rubin explains:

The '609's claimed inventions, and related disclosure, concern securing data within a system. Ex. 1001 ['609] 1:34-39 ("computer systems... are often storing, archiving, and transmitting all types of sensitive information. Thus, an ever-increasing need exists for ensuring data stored and transmitted over these systems cannot be read or otherwise compromised."). This is accomplished by the claimed

method of parsing the dataset into chunks, encrypting those chunks with respective second keys, further securing the chunks with a cryptographic operation based on a first key, and then storing at least one chunk on a storage device. *Id.*, cl. 1. Each of these steps is intended to secure the dataset. “[C]ryptographic parsing and splitting” of the dataset into chunks and “physical[] remov[al]” of at least one of the encrypted and “further secure[d]” chunks from the system onto a storage device removes the possibility of “attack” or “compromise” of the whole dataset. *Id.*, 56:44-58.

Foster, by contrast, concerns a system wherein a content owner provides content to Foster’s system for encryption so that the content can be streamed (*e.g.*, via the internet) to a recipient.

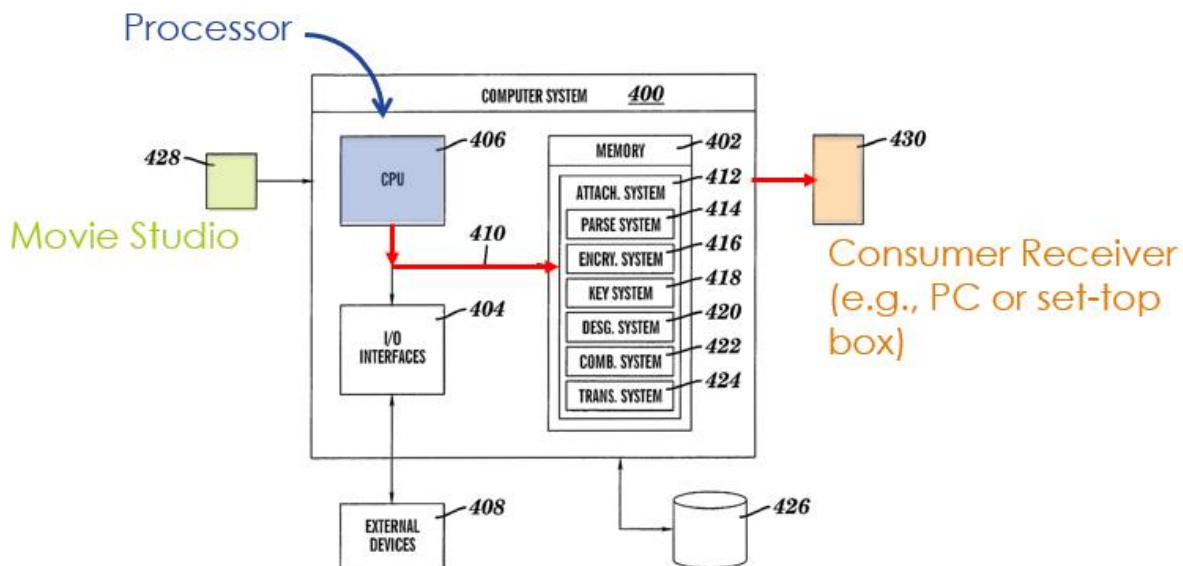


FIG. 16

Ex. 1006 [Foster] Fig. 16 (annotated); [0002] (“a method, system and program product for attaching a title key to encrypted content for synchronized transmission to a recipient.”); [0078] (“computer system 400 is intended to be representative of any system capable of processing content units of an elementary content stream received from source 428, and transmitting the processed content units to a recipient 430.”).

Foster, thus, essentially acts as an encryption middleman for streamed content. Foster, in contrast to the claimed inventions, does not “generate” a plurality of data chunks from a dataset. Rather, it just streams encrypted content in network packets (in accord with the applicable standards). Nor does Foster’s system even store at least one chunk on a storage device.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 92-94.

Thus, unsurprisingly, Petitioner’s Foster grounds fail for multiple reasons.

B. Petitioner Fails To Show Foster Discloses “Generating A Plurality Of Data Chunks Based On A Data Set” (Limitation 1[c][1]).

Limitation 1[c][1] requires “generating a plurality of data chunks based on a data set.” Petitioner maps the claimed “plurality of data chunks” to Foster’s content units (more specifically, MPEG packet payloads) and the claimed “data

set” to Foster’s “content.”⁴ Pet., 65-66. Petitioner, however, fails to show that Foster’s packet payloads are “data chunks.”

Petitioner asserts with no further explanation that Foster’s “payloads” are “data chunks”:

As illustrated in Foster’s Figure 8 . . . , Foster generates a plurality of “**payloads**” (“***data chunks***”) by “parcel[ing]” the elementary digital content. *See also supra* §VII.B.1 (1[PRE]); Foster, [0064]; Bhattacharjee, ¶¶183-84.

Pet., 65 (emphasis in original). As Dr. Rubin explains:

Petitioner and Dr. Bhattacharjee rely on Foster’s packet “payloads” as the claimed data chunks. Pet., 65. Foster teaches that these “payloads” refer to the payloads of MPEG Packetized Elementary Stream (PES) packets (also referred to in Foster as “units”), as illustrated below in Figure 13:

⁴ Petitioner does not contend it would be obvious to modify Foster to satisfy this limitation.

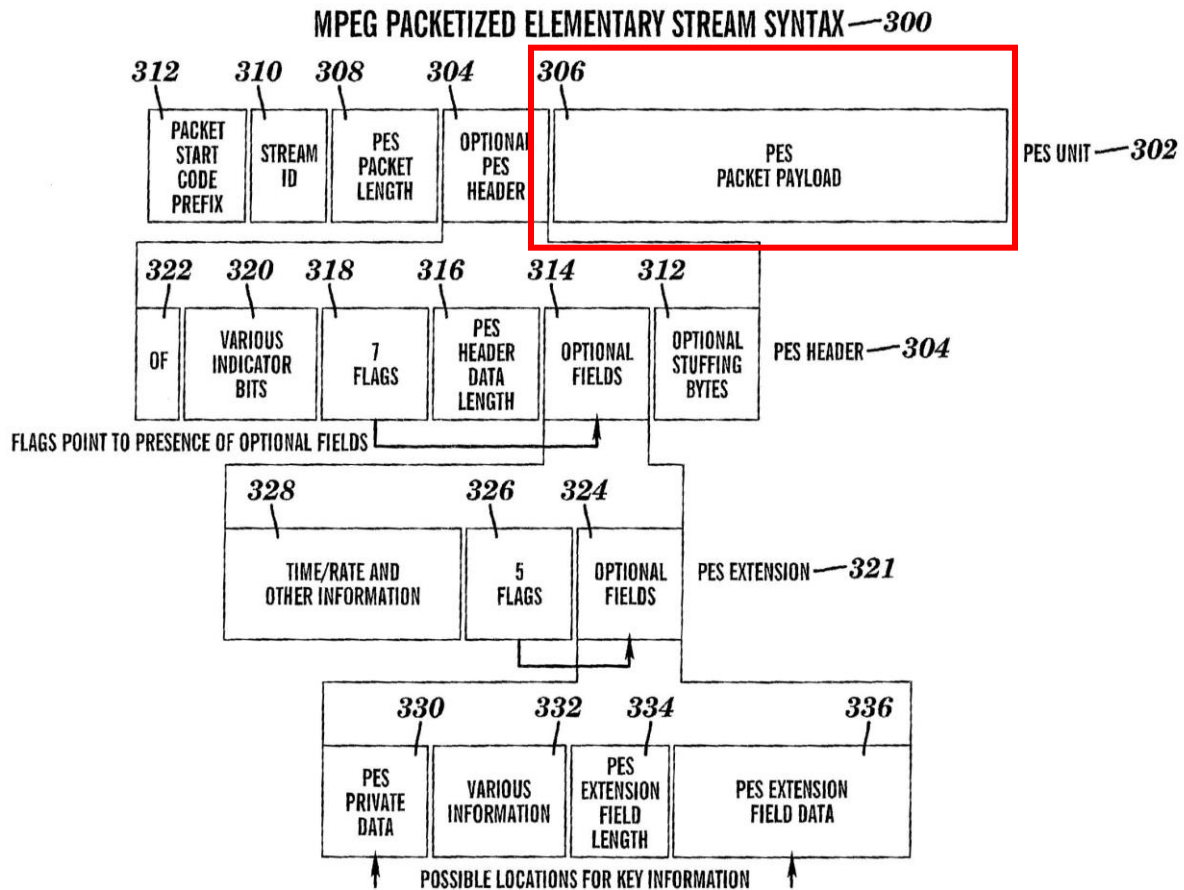


FIG. 13

Ex. 1006 [Foster] Fig. 13, [0072] (“a MPEG packetized elementary stream 300 is parceled into PES (packet) units 302 PES unit 302 includes packetized elementary stream (PES) packet payload 306” which “will be encrypted with a title key”). Foster’s PES packets are also referred to as “data packets.” See Ex. 2011 [Tsai] [0001], cl. 1. Consequently, Petitioner and Dr. Bhattacharjee map the claimed “data chunks” to Foster’s packets payloads which are then individually encrypted. There is no teaching in Foster that these packet payloads are themselves sub-divided into shares or chunks.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 97.

But neither Petitioner nor Dr. Bhattacharjee tells us what “data chunks” are or why packet “payloads” are “data chunks.” Certainly, Foster never refers to its packet payloads as “data chunks” or shares. At best, Petitioner’s assertion is conclusory and does not satisfy its burden of proof. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (citing *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)) (“[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.”); *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.”). That is sufficient reason to reject Petitioner’s Foster ground.

Nor is there any reason to just assume that a packet payload would be deemed a chunk. As Dr. Rubin explains, the literature frequently refers to a packet or packet payload as something that is divided into “chunks” or “shares”:

Petitioner and Dr. Bhattacharjee, again, rely upon Foster’s packet payloads as data chunks. Numerous technical articles, however, refer to a packet or packet payload being divided into chunks or shares. See,

e.g., Ex. 2036 [Hur] 1202 (“The new scheme uses a content-based chunking algorithm to consistently divide both files and packet payloads into multiple byte sequences, called chunks.”); Ex. 2037 [Li-I] 1 (“These services allow breaking down packet payloads into smaller units (called chunks) ...”); *see also id.*, 2 (“It provides a packetization method that breaks down the payload into multiple chunks, each with a certain significance.”); Ex. 2012 [Pek] 3 (“[The d]ata packet is delivered as a sequence of chunks.”); Ex. 2013 [Groote] 3 (“the input is a large data packet ... which is delivered in small chunks.”); Ex. 2038 [McKeown] 4-5 (“A packet consists of multiple chunks...); Ex. 2041 [Li-II] 7 (“the payload could be divided into multiple equal-sized chunks ...”); Ex. 2042 [Kumar] 9 (“each node sends out 3 shares in a single packet”); Ex. 2043 [Mao] 9532 (“We break [data packet] D into n shares according to the (t, n) -threshold secret sharing algorithm.”).

Ex. 2033 [Rubin-2nd-Decl.] ¶ 98.

This usage, wherein a packet is divided into data chunks is, moreover, reflected in the ’609 patent itself. Per Dr. Rubin:

Consistent with this usage in the literature, the ’609 identifies a data packet (or packet payload) as something that is divided into shares (*i.e.*, chunks). Ex. 1001 [’609] 82:15-18 (“At step 4106, ***the data packet or data block may then be partitioned into two or more data shares . . .***”), 81:30-31 (“***Packets of data (e.g., network packets) may also be efficiently split*** and restored”), 81:48-52 (“...simplified ***data partitioning process 4010 ... may be performed on network packets, . . .***”). Therefore, the ’609 identifies data packets for streaming, such as

those disclosed in Foster, as the unit to be parsed, not as “data chunks [generated] based on a data set.” In other words, by disclosing a method for secure transmission of MPEG data packets, Foster does not disclose “generating a plurality of data chunks” as a POSITA would understand the limitation based on the ’609’s specification. Notably, neither Petitioner nor Dr. Bhattacharjee identifies any teaching in the ’609 or in the technical literature suggesting that a data packet would qualify as a “data chunk.”

Ex. 2033 [Rubin-2nd-Decl.] ¶ 99.

Moreover, the difference between the claimed “data chunks” and Foster’s packet payloads is significant. As Dr. Rubin explains whereas the ’609’s data chunks or shares are generated to further secure the data, Foster’s packet payloads merely exist as a by-product of the pertinent MPEG streaming standard:

Nor does the ’609’s discussion of shares suggest that a packet payload would be deemed a “data chunk.” As the ’609 explains, partitioning the data set into chunks increases security because when “data ... remains whole and intact,” it is “subject to attack.” The ’609 addresses this concern, in part, by “performing a cryptographic parsing and splitting of the encrypted file.” Ex. 1001 [’609] 56:44-58. Thus, the ’609 teaches that splitting the data set into shares (*i.e.*, chunks) help secure the data.

A packet payload (as in Foster), however, does not exist to protect the data. Rather it exists by virtue of the MPEG packetized elementary stream standard. Ex. 1006 [Foster] [0072]-[0073]; Ex. 2014

[Overview-of-MPEG-2-Systems] (“An elementary stream is broken up into packets, forming a *packetized elementary stream* (PES).”). These packets allow for transmission of the content. See Ex. 2015 [Newton’s-Telecom-Dictionary] 626 (“**Packet** 1. Generic term for a bundle of data, usually in binary form, organized in a specific way *for transmission*.”); Ex. 2016 [Oxford-Dictionary-of-Computing] 257 (“a group of binary digits forming a message. The group is *transmitted* together”).

Thus, the ’609 splits a data set into shares or chunks as a means of protecting a dataset. This is why a packet in the ’609 is an example of a data set that is split into shares. See Ex. 1001 [’609] 82:15-18. By contrast, Foster’s packet payloads exist not to protect data but, rather, because it is just a by-product of a pre-existing standard. Consequently, a POSITA reading the ’609 patent would not consider a packet payload to be a data chunk in view of the ’609’s disclosure.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 100-102.

Thus, per the ’609 patent, a packet payload is not a share/chunk, it is something that is divided into shares/chunks. While the specification discusses dividing a data set into “shares” rather than “data chunks,” a POSITA would understand the “data chunks” in the claims correspond to the shares described in the ’609. As Dr. Rubin explains:

I note that while the ’609 does not use the term “chunk(s)” throughout the specification, these “shares” described are the

equivalent of the claimed “chunks.” The claims recite “generating a plurality of data chunks based on a data set, wherein each data chunk ... comprises ***less than an entirety of data*** of the data set.” This corresponds to the ’609’s teachings wherein a data set is divided into shares. *E.g.*, Ex. 1001 [’609] 21:21-37 (“The data may be split into any ... size unit[s]” which “may be distributed into one or more shares.”); 54:17-18 (“A cryptographic split (cryptosplit) partitions the data into N number of shares.”); 56:47-54 (“performing a cryptographic parsing and splitting of the encrypted file into two or more portions or shares”); 71:35-37 (“secure data parser 3000 may include built-in capabilities for parsing and splitting data into portions (also referred to herein as shares) of data”).

Ex. 2033 [Rubin-2nd-Decl.] ¶ 103.

Nor is there anything unusual about “data chunks” in the claim corresponding to the “shares” described in the specification. A patent may use different terms interchangeably to mean the same thing. *Baran v. Med. Device Techs., Inc.*, 616 F.3d 1309, 1316 (Fed. Cir. 2010) (internal citations omitted) (“detachable” and “releasably” have the same meaning); *see also Haddad v. United States*, 164 Fed. Cl. 28, 66 (2023) (“authenticity risk rating,” “authentication rating,” “ID forgery risks rating” used interchangeably).

Thus, in accord with the relevant technical literature, a POSTIA reviewing the ’609’s specification would understand that a packet payload is not a “data

chunk,” rather, it is something that may be divided into “data chunks.” Rather, a packet payload is identified as a paradigmatic example of a data set that is divided into shares. In claim construction, “the plain meaning is not the meaning of the term in the abstract.” *Sci. Applications Int’l Corp. v. U.S.*, 154 Fed. Cl. 594, 623 (2021). Instead, it is the meaning in the context of the patent’s specification. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005) (en banc). Here, the ’609’s specification, consistent with the literature, indicate to a POSITA that Foster’s “payloads” are not the claimed “data chunks.” *Phillips*, 415 F.3d at 1315 (the specification “‘is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.’”) (quoting *Vitronics Corp. v. Conceptronic*, 90 F.3d 1576, 1582 (Fed. Cir. 1996)).

Thus, Petitioner fails to demonstrate that Foster discloses or renders obvious “generating a plurality of data chunks based on a data set.”

C. Petitioner Fails To Show That Foster Discloses Or Renders Obvious “Executing Code By A Processor To Perform ... Storing ... At Least One Data Chunk ... On At Least One Storage Device” (Limitations 1[a] & 1[f]).

The claims center around a “processor” that “execut[es] code” to perform a number of steps (limitation 1[a]). This includes “**storing**, in a memory coupled to the processor, **at least one data chunk** of the plurality of data chunks with data

indicative of at least one of the distinct encryption keys ***on at least one storage device***” (limitation 1[f]). Ex. 1001 [’609] cl. 1.

Petitioner identifies Foster’s “CPU 406” as the claimed processor. See Pet., 62. Petitioner further contends that the “storage device” is recipient 430.

Petitioner contends that Foster’s “***receiver (e.g., consumer set-top box) can store the content units***” (the alleged chunks). *Id.*, 73, 75 (“Each content unit 116 ... is ***stored on receiver*** at ‘receipt 430’”).

Petitioner, however, does not attempt to show that Foster’s CPU 406, executes any code relating to storing content units on recipient 430 or instructing recipient 430 to store content units. Petitioner does not mention—let alone explain—how Foster’s CPU 406, which is part of Foster’s computer system 400, executes code to store the content units on the separate recipient 430 (*e.g.*, a set-top box). See Pet., 72-75. Nor is there any reason to believe that Foster’s CPU 406 executes code to perform storing the content units on Foster’s recipient 430 because, as Dr. Rubin explains, Foster’s CPU 406 simply transmits the content units to the recipient 430 and is agnostic as to what the recipient 430 does with the content units after transmission:

As can be seen in Foster’s Figure 16 below, Foster’s **recipient 430** is not part of the **computer system 400** which contains **CPU 406**:

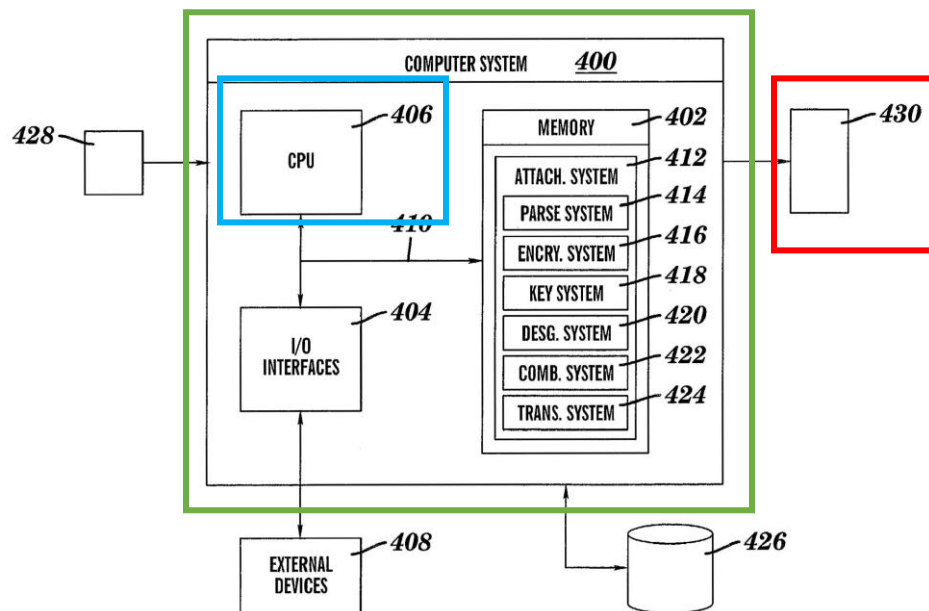


FIG. 16

Ex. 1006 [Foster] Fig. 16, [0075] (listing what “computer system 400 generally comprises,” not including recipient 430). Consequently, for Foster’s CPU 406 to “execute code ... to perform” “storing ... at least one data chunk,” Foster’s CPU 406 must at least instruct recipient 430 to store or perform some action related to storing a content unit.

Foster, however, teaches that its computer system 400 merely “transmit[s] the processed content units to [the] recipient 430.” *Id.*, [0078]. Once the recipient 430 receives the content unit, the “recipient 430 *can*,” if it so chooses “then synchronously store the encrypted content and attached title key.” *Id.*, [0079]. Significantly, Foster teaches that recipient 430 can also instantly display the content units rather than storing them, which would be the typical usage in the

television context Foster contemplates. *See id.*, [0008] (content unit processed “for synchronized transmission to, **or** storage by a receiver.”); [0052] (Foster seeks to improve upon prior art “Conditional Access System (CAS) such as a cable or satellite television system.”); [0073] (content units can be transported via “[a] transport stream [which] is typically used in error-prone broadcast environments such as digital television.”).

Thus, Foster’s computer system 400 transmits content units to recipient 430. But neither computer system 400 nor CPU 406 executes code instructing recipient 430 to store the content. Fundamentally, in Foster’s system, computer system 400 is agnostic as to what the recipients do with the content units, and instead, the recipient determines whether the content units are stored or not. Indeed, Foster only teaches sending a **header** and a header extension with **key information** along with the encrypted payload:



46

optional identifier information 120 and optional verifier 122.” *Id.*, [0066]. Notably, Foster does not teach that the header or the header extension contains instructions on what the receiver should do with the content units upon receipt. Consequently, the decision on how the content is handled and whether it is stored is made by the receiver itself (by its own processor(s) and code). Thus, Foster’s CPU 406 (*i.e.*, the alleged processor) does not execute code for “storing” the alleged data chunks.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 110-112. Thus, there is no indication that Foster’s CPU 406 provides any instructions to recipient 430 concerning whether to store any content.

Petitioner offers an alternative, obviousness theory attempting to account for the fact that recipient 430 is not “coupled” to CPU 406. Pet., 77 (“POSAs had reason to use one of the conventional storage devices coupled to Foster’s processor to store data sets encrypted by Foster’s processor.”). This argument is discussed in Section III.D, *infra*. But even if Petitioner’s modification were accepted, Petitioner presents no argument as to how (or why) a POSITA would be motivated to modify CPU 406 to execute code to store encrypted content units on these “conventional storage devices.” *Id.*, 76-78. At best, Petitioner’s argument is vague and conclusory. But obviousness may not be shown through conclusory statements. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (citing *In re Kahn*, 441

F.3d 977, 988 (Fed. Cir. 2006)) (“[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.”); *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.”). Thus, Petitioner also fails to carry its burden as to its obviousness theory.

Therefore, Petitioner fails to demonstrate that Foster discloses or renders obvious “executing code by a processor to perform” “**storing**, in a memory coupled to the processor, **at least one data chunk** of the plurality of data chunks with data indicative of at least one of the distinct encryption keys **on at least one storage device**.”

D. Petitioner Fails To Show That Foster Discloses Or Renders Obvious “Storing, In A Memory Coupled To The Processor, At Least One Data Chunk . . . On At Least One Storage Device” (Limitation 1[f]).

Limitation 1[f] requires “storing, **in a memory coupled to the processor**, at least one data chunk of the plurality of data chunks with data indicative of at least one of the distinct encryption keys on at least one storage device.” Ex. 1001 [’609]

cl. 1. Petitioner, however, fails to show that the alleged “memory” (an alleged hard-drive on Foster’s recipient 430) is “coupled” to the alleged processor. *See* Section III.D.1. Petitioner, moreover, fails to demonstrate a POSITA would be motivated to modify Foster to address this shortcoming. *See* Section III.D.2.

1. Petitioner Fails To Demonstrate That Foster Discloses Limitation 1[f].

Petitioner fails to show that the memory on Foster’s receiver is “coupled to” the processor. Petitioner identifies Foster’s CPU as the alleged processor and the alleged hard drive on Foster’s receiver as the alleged “memory.” *Pet.*, 75. As Dr. Rubin explains, however, there is no showing that Foster’s processor is “coupled to” the “memory” on the receiver:

Petitioner and Dr. Bhattacharjee allege that the alleged processor is “coupled to” the receiver via “transmission system 424” and “bus 410.” *Pet.*, 75; *Ex. 1003 [Bhattacharjee-Decl.] ¶ 205.*

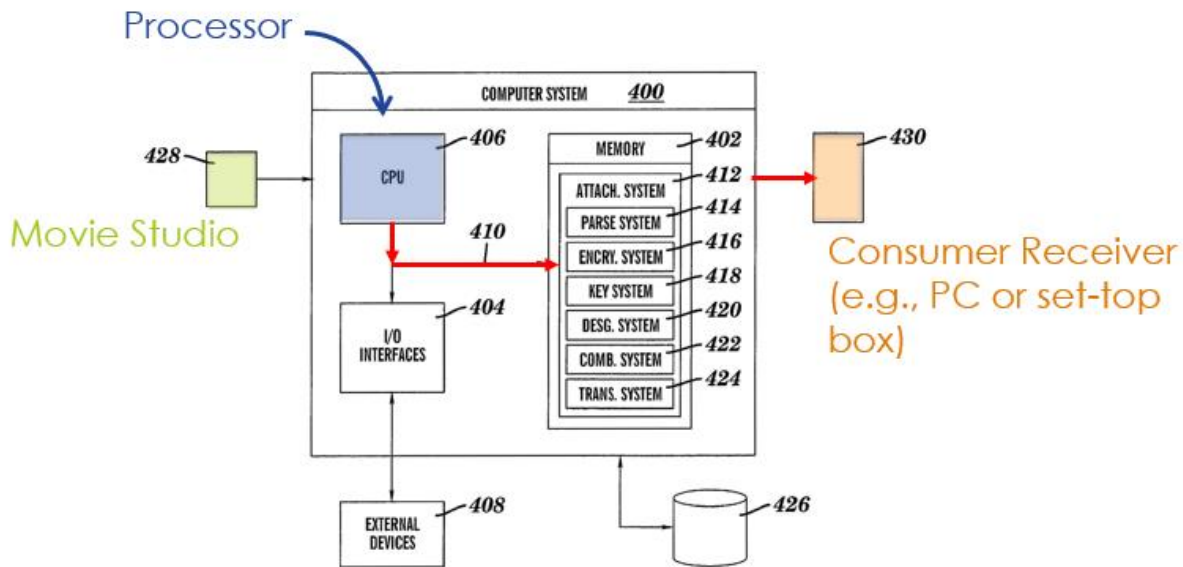


FIG. 16

Ex. 1006 [Foster] Fig. 16 (annotated). Petitioner further contends that the “memory” is an alleged hard drive on Foster’s receiver. See Pet., 75 (“that receiver is a ‘storage device’ (e.g., consumer set top box) with ‘memory’ (e.g., a hard drive).”).

Foster’s CPU, as discussed previously, adds encryption information to packets for distribution to receivers. As discussed previously, however, there is no indication that the CPU tells the receiver what to do with those content units. It certainly does not tell the receiver to store the content units. Thus, there is no reason to believe that Foster’s CPU is “coupled to” the alleged hard drive on the receiver (e.g., set top box).

Certainly, nothing in Foster’s disclosure suggests that the CPU is “coupled to” the alleged “hard drive” on the recipient. Notably,

Foster does not mention any “hard drive” on the recipient’s system. Certainly, there is no teaching that Foster’s CPU is coupled to any memory on the recipient. Petitioner points to Foster’s “bus 410” but bus 410 is a “communication link” for the components in Foster’s computer system 400; it does not connote a communication link to the recipient, much less the recipient’s alleged hard drive. Ex. 1006 [Foster] [0076] (“Bus 410 provides a communication link ***between each of the components in the computer system 400.***”). Similarly, Foster’s transmission system 424 is described as transmitting information to recipient 430. See Ex. 1006 [Foster] [0079] (“In any event, the encrypted content packet, the header and the header extension will be combined into a single processed content unit by combination system 422 and synchronously transmitted to recipient 430 via transmission system 424.”). In Foster, it is the recipient’s system alone that decides whether to store the information. *Id.* (“Once received, recipient 430 can then synchronously store the encrypted content and attached title key.”). Thus, at most, Foster discloses sending content units from Foster’s system to the recipient. Foster discloses that it is solely the recipient that decides whether to store that information. The independent decision at the recipient to store the transmitted information (or not store it) does not disclose or suggest that any alleged hard drive on the recipient is “coupled to” Foster’s CPU 406.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 118-121. Thus, Petitioner has not shown that Foster’s processor is “coupled to” the alleged hard drive on Foster’s recipient.

2. Petitioner Fails To Demonstrate A POSITA Would Modify Foster As Proposed.

Tacitly admitting that Foster does not disclose “storing, in a memory coupled to the processor, at least one data chunk,” Petitioner “alternatively” argues that a POSITA “had reason” to modify Foster such that “Foster’s encrypted content [is stored] in the memory of storage devices coupled to the processor (*e.g.*, through a bus, direct wired/wireless connection, and/or network-based connection.” Pet., 76. Petitioner argues that a POSITA would have modified Foster to do so in order to avoid “repeat[ing] the encryption processing every time digital content (*e.g.*, a movie) needed to be encrypted and delivered to recipients” and thereby allowing Foster’s system to facilitate “(1) backup, (2) video-on-demand distribution, and/or (3) distribution to later-identified recipients.” *Id.*, 77. Petitioner further contends that so storing the content would have been applying a “known technique” to solve a known problem of “how to store encrypted content for delivery to multiple recipients at varying times . . .” *Id.*

But Petitioner’s proposed modification does not actually solve any problem. As Dr. Rubin explains, a POSITA would recognize that the purported benefits Petitioner identifies vis-à-vis “(1) backup, (2) video-on-demand distribution, and/or (3) distribution to later-identified recipients,” are already solved by Foster’s system because those benefits are provided by the content owner:

Foster's system is meant merely to act as a middleman for encryption and synchronized transmission of content (*e.g.*, a movie) from **content sources** to **consumer recipients**, as can be seen in Figure 16 below.

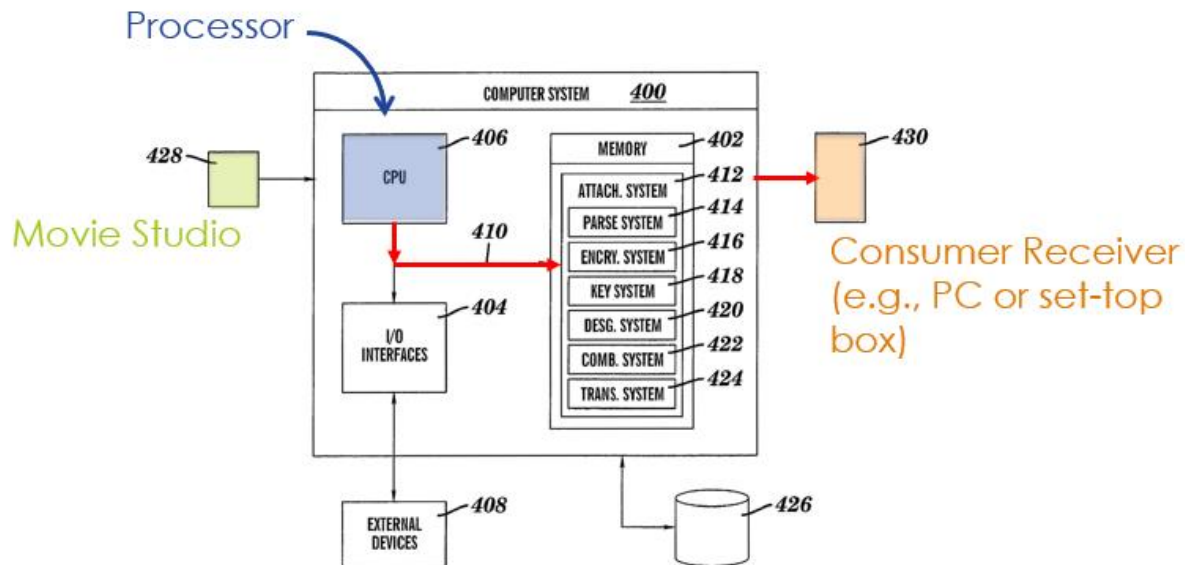


FIG. 16

Ex. 1006 [Foster] Fig. 16 (annotated); [0002] (“a method, system and program product for attaching a title key to encrypted content for synchronized transmission to a recipient.”); [0078] (“computer system 400 is intended to be representative of any system capable of processing content units of an elementary content stream received from source 428, and transmitting the processed content units to a recipient 430.”).

Foster further teaches that the content “source” or “owner” is an entity that owns and provides the content. *Id.*, [0038], [0042]. A POSITA would thus recognize that storage for “(1) backup, (2) video-

on-demand distribution, and/or (3) distribution to later-identified recipients” is already provided by the content source/owner. Indeed, none of the references Petitioner cites as alleged background knowledge of a POSITA teaches storage of content by an intermediary. *See* Exs. 1045, 1050, 1060. Moreover, Foster does not indicate that its system has any problems that storing encrypted content for delivery to multiple recipients would solve or benefit. Foster teaches synchronously transmitting content to recipients, thereby eliminating delays found in prior art systems. *Id.*, [0006]. Foster further contemplates this synchronized transmission in the streaming and cable television contexts, which already involve multiple recipients. *Id.*, [0002], [0064]. There simply is no problem that would be solved by pre-encrypting the content.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 123-124.

Moreover, Petitioner’s proposed modifications would either compromise the streamed content’s security or require significant resources to prevent such a thing, as Dr. Rubin explains:

The entire premise of Foster’s system is that it can receive content ***without*** storing it, encrypt the content, and synchronously transmit it to consumers. *See* Ex. 1006 [Foster] [0002], [0006]. Such a system entails numerous benefits for it ensures the content is secure. By contrast, if Foster’s system were altered so that the content were stored in Foster’s system, that would create a significant security risk as Foster’s system would be hacked and an adversary would only have

to target the key management block on a single consumer receiver to compromise all of the stored pre-encrypted content. There is simply no reason why a POSITA would be motivated to do this as it would surely frustrate Foster’s purpose of securing the content (vis-à-vis encryption) and transmitting it to the receiver without exposing the content to further risk. As discussed, a POSITA would recognize that content is already securely stored in Foster’s content sources/owners. A POSITA—and, more importantly, the content owners—would not be motivated to expose the content to potential theft unnecessarily. Moreover, if Foster’s system were modified to store pre-encrypted content, significantly more storage would have to be added and that additional storage would have to be as secure as the storage at the content provider/owner. Otherwise, content provider/owners would never trust a third-party system to store its content. Indeed, Foster teaches security of the content is of such a concern that the title keys used to encrypt individual content packets need to be changed relatively “frequently.” Ex. 1006 [Foster] [0055].

Ex. 2033 [Rubin-2nd-Decl.] ¶ 126.

Thus, Petitioner’s proposed modification is not needed by Foster and would, in fact, frustrate Foster’s purpose by comprising security. A POSITA would not be motivated to make Foster’s system worse. *Polaris Indus., Inc. v. Arctic Cat, Inc.*, 882 F.3d 1056, 1069-70 (Fed. Cir. 2018) (reversible error to find motivation to combine where resulting combination “would run contrary to [the primary reference’s] stated purposes.”); *Plas-Pak Indus., Inc. v. Sulzer Mixpac AG*, 600 F.

App’x. 755, 759 (Fed. Cir. 2015) (“A change in a reference’s ‘principle of operation’ is unlikely to motivate a [POSITA] to pursue a combination with that reference.”); *In re Ratti*, 270 F.2d 810, 813 (CCPA 1959) (no motivation to combine where “suggested combination of references would require . . . a change in the basic principles under which the [reference] was designed to operate.”); *Arctic Cat Inc. v. Polaris Indus.*, 795 F. App’x. at 833 (“The Board must weigh the benefits and drawbacks of the modification against each other, to determine whether there would be a motivation to combine.”); *Henny Penny*, 938 F.3d at 1331-32 (rejecting argument “Board placed undue weight on the disadvantages of introducing Iwaguchi’s TPM sensor into Kauffman’s system” where tradeoffs of combination may “yield an unappetizing combination”).

Thus, Petitioner fails to demonstrate that Foster discloses or renders obvious “storing, ***in a memory coupled to*** the processor, at least one data chunk of the plurality of data chunks with data indicative of at least one of the distinct encryption keys on at least one storage device.”

IV. CONCLUSION

Petitioner fails to demonstrate by a preponderance of the evidence that any of the claims are invalid.

Respectfully submitted,

/ Stephen J. Elliott /

Stephen J. Elliott (Reg. No. 52,858)

Andrei Iancu (Reg. No. 41,862)

SULLIVAN & CROMWELL LLP

Kenneth J. Weatherwax (Reg. No. 54,528)

Nathan Lowenstein, *pro hac vice*

Colette Woo, *pro hac vice*

LOWENSTEIN & WEATHERWAX LLP

Date: September 12, 2024

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITS

This Patent Owner Response (the “POR”) consists of 10,969 words, excluding table of contents, table of authorities, certificate of service, this certificate, or table of exhibits. The POR complies with the type-volume limitation of 14,000 words as mandated in 37 C.F.R. § 42.24. In preparing this certificate, counsel has relied on the word count of the word-processing system used to prepare the paper (Microsoft Word).

Respectfully submitted,

/ Colette Woo /

Date: September 12, 2024

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the following documents were served by electronic service, by agreement between the parties, on the date below:

PATENT OWNER RESPONSE

EXHIBITS 2033, 2034, 2036-2038, 2041-2044

The names and addresses of the parties being served are as follows:

Elisabeth H. Hunt	EHunt-PTAB@wolfgreenfield.com
Gregory S. Nieberg	GNieberg-PTAB@wolfgreenfield.com
Nathan R. Speed	nspeed@wolfgreenfield.com

Respectfully submitted,

/ Colette Woo /

Date: September 12, 2024