

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MASTERCARD INCORPORATED AND MASTERCARD INTERNATIONAL
INCORPORATED
Petitioner

v.

OV LOOP, INC.
Patent Owner

U.S. Patent No. 10,032,171

Inter Partes Review No. IPR2023-1289

**SUPPLEMENTAL DECLARATION OF STUART FISKE
IN SUPPORT OF PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 10,032,171**

EXHIBIT LIST

Exhibit No.¹	Description
1001	U.S. Patent No. 10,032,171 (the “’171 Patent”)
1002	File History of the ’171 Patent
1003	U.S. Provisional App. No. 61/685,863
1004	U.S. Provisional App. No. 61/573,476
1005	U.S. Provisional App. No. 61/575,846
1006	Expert Declaration of Stuart Fiske
1007	Curriculum Vitae of Stuart Fiske
1008	Reserved
1009	U.S. Provisional App. No. 61/619,095 (“the ’095 Provisional”)
1010	U.S. Provisional App. No. 61/635,248 (“the ’248 Provisional”)
1011	U.S. Publication No. 2013/0262317 A1 to Collinge (“Collinge”)
1012	U.S. Patent No. 11,080,693 to Fiske (“Fiske”)
1013	Reserved
1014	U.S. Patent Publication No. 2010/153721 A1 to Mellqvist
1015	U.S. Patent Publication No. 2010/0252624
1016	U.S. Patent Publication No. 2012/0011070 A1 to Ward
1017	Reserved
1018	Declaration of Brian Byrne (“Byrne Declaration”)
1019	<i>EMV Integrated Circuit Card Specifications for Payment Systems, Book 1: Application Independent ICC to Terminal Interface Requirements</i> , Version 4.2 (June 2008) (Attachment A to Byrne Declaration)
1020	<i>EMV Integrated Circuit Card Specifications for Payment Systems, Book 2: Security and Key Management</i> , Version 4.2 (June 2008) (Attachment B to Byrne Declaration)
1021	<i>EMV Integrated Circuit Card Specifications for Payment Systems, Book 3: Application Specification</i> , Version 4.2 (June 2008) (Attachment C to Byrne Declaration)
1022	<i>EMV Integrated Circuit Card Specifications for Payment Systems, Book 4: Cardholder, Attendant, and Acquirer</i>

¹ Citations to patents are made by column and line cite; citations to patent publications are made by paragraph number; citations to other exhibits are made with reference to the Bates-stamped pagination.

Exhibit No.¹	Description
	<i>Interface Requirements</i> , Version 4.2 (June 2008) (Attachment D to Byrne Declaration)
1023	<i>EMV® Contactless Specifications for Payment Systems, Book A: Architecture and General Requirements</i> , Version 2.1 (March 2011) (Attachment E to Byrne Declaration)
1024	<i>EMV® Contactless Specifications for Payment Systems, Book B: Entry Point Specification</i> , Version 2.1 (Attachment F to Byrne Declaration)
1025	<i>EMV® Contactless Specifications for Payment Systems, Book C-1: Kernel 1 Specification</i> , Version 2.1 (March 2011) (Attachment G to Byrne Declaration)
1026	<i>EMV® Contactless Specifications for Payment Systems, Book C-2: Kernel 2 Specification</i> , Version 2.1 (March 2011) (Attachment H to Byrne Declaration)
1027	<i>EMV® Contactless Specifications for Payment Systems, Book C-3: Kernel 3 Specification</i> , Version 2.1 (March 2011) (Attachment I to Byrne Declaration)
1028	<i>EMV® Contactless Specifications for Payment Systems, Book C-4: Kernel 4 Specification</i> , Version 2.1 (March 2011) (Attachment J to Byrne Declaration)
1029	<i>EMV® Contactless Specifications for Payment Systems, Book D: EMV Contactless Communication Protocol Specification</i> , Version 2.1 (March 2011) (Attachment K to Byrne Declaration)
1030	Reserved
1031	Reserved
1032	Expert Declaration of Dr. Sylvia Hall-Ellis dated August 11 (“Hall-Ellis Declaration”) and Attachments
1033	Curriculum Vitae of Dr. Sylvia Hall-Ellis
1034	Carl H. Meyer, Stephen M. Matyas, <i>Cryptography: A New Dimension In Computer Data Security. A Guide for the Design and Implementation of Secure Systems</i> (1982)
1035	Reserved
1036	Reserved
1037	Cristian Radu, <i>Implementing Electronic Card Payment Systems</i> (2003)
1038	Reserved

Exhibit No. ¹	Description
1039	Kefa Rabah, <i>Theory and Implementation of Data Encryption Standard: A Review</i> , Inform. Technol. J., 4: 307-325 (Apr. 2005)
1040	Smart Card Alliance, <i>Card Payments Roadmap in the United States: How Will EMV Impact the Future Payments Infrastructure?</i> A Smart Card Alliance Payments Council White Paper (Feb. 2011)
1041	Dom Morea, <i>EMV in the U.S.: Putting It into Perspective for Merchants and Financial Institutions</i> , A First Data White Paper (2011)
1042	EMVCo FAQ webpage (Aug. 3, 2023, 11:43 AM), https://web.archive.org/web/20110920080137/http://www.emvco.com/faq.aspx?id=37
1043	Newton's Telecom Dictionary (25th ed. 2009)
1044	Ofcom: <i>Mobile not-spots. An update on our research</i> (Nov. 5, 2010)
1045	<i>OV Loop, Inc. v. Mastercard Inc.</i> , No. 7:23-cv-01773-CS, D.I. 31 (S.D.N.Y. June 8, 2023)
1046	Mike Clark, <i>SimplyTapp picks up contactless payments award</i> , NFCW (Aug. 25, 2015), https://www.nfcw.com/2015/08/25/337201/simplytapp-picks-up-contactless-payments-award/
1047	Payal Shukla, Jim Jarvie, <i>SimplyTapp Wins Best Contactless Payments Solution as Future Digital Awards</i> (Aug. 13, 2015), https://www.prweb.com/releases/2015/08/prweb12903489.htm
1048	U.S. Patent Pub. No. 2010/0190437A1
1049	Transcript of Deposition of Stuart Fiske
1050	Supplemental Expert Declaration of Stuart Fiske
1051	Transcript of Deposition of Chuck Easttom
1052	U.S. Patent No. 10,956,899
1053	U.S. Patent No. 8,370,258
1054	U.S. Patent No. 7,357,309
1055	U.S. Patent No. 6,718,314
1056	U.S. Patent Publication No. 20070012763
1057	U.S. Patent Publication No. 20030097344
1058	PayPass – Mag Stripe Technical Specifications, Version 3.3, December 2007

Exhibit No.¹	Description
1059	PayPass – M/Chip 4 Card Technical Specification, Version 1.3.1 - September 2008
1060	U.S. Patent Publication No. 20120160921
1061	U.S. Patent No. 10,902,399
1062	U.S. Patent Publication No. 20120185394
1063	International Standard ISO/IEC 14443-4 (2008)
1064	International Standard ISO/IEC 7816-4 (2005)
1065	International Standard ISO/IEC 7813 (2006)

TABLE OF CONTENTS

I.	Introduction.....	1
II.	Authentication of Evidence	2
III.	Emv and Other Payment Standards Are Evidence of a POSITA’s Knowledge Is Appropriate, as Confirmed by the ‘171 Patent Itself	6
A.	EMV And PayPass Are Critical Background References That Even The ‘171 Patent Relies on To Supply Necessary Disclosures	6
B.	Collinge and Fiske Both Assume the POSITA’s Knowledge of the Background Standards.....	10
IV.	Grounds of Invalidity.....	10
A.	Grounds 1 and 2: Collinge in View of Knowledge of a POSITA	10
1.	Claim 1	10
a.	[1.A-3]: “... configured to request, ... data associated with the account and”	10
b.	[1.C]: “... non-permanent cryptographic key associated with the account;”	12
c.	[1.C]: “the first set of data associated with the account usable by the application to formulate an application protocol data unit response;”	14
d.	[1.E-2]: “... wherein the response cryptogram does not include the local cryptographic key;”	15
e.	[1.E-3]: “device response communication comprising at least one response application data protocol unit containing the response cryptogram and an account identifier for the account.”	16
2.	Claim 10 and 27: “remote computer system”	18
a.	[10.B]/[27.B] “determining a current non-permanent cryptographic key ... using the account identifier”	21
b.	[10.C]/[27.C]: “calculating a calculated cryptogram using the current nonpermanent cryptographic key associated with the account”	21
3.	Claim 8	22

a.	[8.A]: “at the remote computer system ... sending the first non-permanent cryptographic key and the account identifier to the mobile device”	22
b.	[8.B]: “after the interrogation, generating the second non-permanent cryptographic key at the remote computer system, associating ... and sending ...”	23
4.	Claims 4 and 20: “wherein the interrogation conforms to a card specification...”	23
5.	Claim 15 and 32:	25
6.	Claims 6 and 22: “...the response cryptogram comprises encrypting ... according to ...(DES) ...wherein the response cryptogram includes a portion, but not all, of the output”	25
7.	Claims 16 and 33: “wherein the first and second non-permanent cryptographic key are associated with an issuer master key”	26
B.	Grounds 3 and 4: Claims 1-33 Are Anticipated and Rendered Obvious by <i>Fiske</i> in View of Knowledge of a POSITA.....	27
1.	Claim 8, 9.....	27
2.	Claims 10-13, 27-30.....	28
a.	“the authorization request comprising authorization data, wherein the authorization data comprises the response cryptogram and the account identifier”	28
b.	“determining a current non-permanent cryptographic key associated with the account using the account identifier”	29
c.	“calculating a calculated cryptogram”	29
3.	Claims 11 and 28: “the set of inputs comprises unpredictable data; the authorization data comprises the unpredictable data; and the calculated cryptogram is calculated using the unpredictable data.”; Claims 12 and 19: “wherein the POS command communications comprise the unpredictable data”	30
4.	Claims 13 and 30: “wherein the authorization data further comprises values from the transaction”	31
5.	Claims 1 and 17.....	32

a.	[1.E-2]: “wherein the response cryptogram does not include the local cryptographic key associated with the account used to generate the response cryptogram”.32
b.	[1.E-3]: “at least one response application data protocol unit containing the response cryptogram and an account identifier for the account” 33
c.	[1.E-3]: EMV mode Visa transactions 34
6.	Claims 4, 5, 20, and 21 35
7.	Claims 6, 15, 22 and 32 36
V.	Conclusion 36
VI.	DECLARATION 36

I. INTRODUCTION

I, Stuart Fiske, declare as follows:

1. My name is Stuart Fiske. I submit this declaration in support of a petition for Inter Partes Review of U.S. Patent No. 10,032,171 (“the ’171 Patent”). I have prepared this declaration as an expert witness retained by Mastercard Incorporated and Mastercard International Incorporated (“Petitioner”). Herein I give my additional opinions as to how a hypothetical person of ordinary skill in the art (“POSITA”) would understand the scope of the claims and whether certain claims of the ’171 Patent are invalid. I provide technical bases for these opinions as appropriate.

2. This declaration contains statements of my opinions formed to date and the bases and reasons for those opinions. I may offer additional opinions based on further review of materials in this case, including opinions and/or testimony of other expert witnesses. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters contained herein. For my efforts in connection with the preparation of this declaration I have been compensated at my standard rate for consulting. My compensation is in no way contingent on the results of these or any other proceedings relating to the above-captioned patent.

3. In formulating my opinions, I have relied upon my training, knowledge, and experience in the relevant art. Both Collinge (Ex[1011]) and Fiske (Ex[1012]) disclose exactly the same techniques claimed in the '171 Patent.

4. The '171 Patent itself extensively cites existing payment standards and characterizes its invention as provisioning a payment application with a single-use, non-permanent key for generating response cryptograms in accordance with those standards. Ex[1002] 617-618. But Collinge's remote provisioning of session key KS_{UN} and Fiske's remote provisioning of EMV's ICC Session Key both do exactly that. It is my opinion that a POSITA applying her background knowledge would find all of the challenged claims obvious in view of those disclosures, and the claims should be found unpatentable.

5. Further, I understand Patent Owner ("PO") disputes my reliance on Collinge Provisionals (Exs[1009]-[1010]). I understand from counsel that Collinge properly incorporates Collinge Provisionals (Exs[1009]-[1010]) by reference and thus I may properly rely on Collinge Provisionals in rendering my opinion. It is also my opinion that a POSITA reading Collinge would have understood Collinge in reference to Collinge Provisionals, given Collinge's explicit incorporation by reference of Collinge Provisionals. Ex[1011] [0001].

II. AUTHENTICATION OF EVIDENCE

6. In connection with this declaration, I reviewed or provided certain evidence to Petitioner, specifically, Ex[1058], Ex[1059], Exs[1063]-[1065].

7. Ex[1058] is a document entitled “PayPass – Mag Stripe Technical Specifications, Version 3.3” and was purportedly published in December 2007. I have previously reviewed and worked with “PayPass – Mag Stripe Technical Specifications, Version 3.3” on many occasions prior to August 2012, and am familiar with its contents. Based on my review, I believe that Ex[1058] is a true and correct copy of PayPass – Mag Stripe Technical Specifications, Version 3.3 published in December 2007. I have also reviewed the PDF metadata of the PDF document submitted as Ex[1058], a screenshot of which is produced below. The dates associated with the document further confirm and corroborate my opinion.

Title	PayPass — Mag Stripe Profile
Author	Eddy Van de Velde
Subject	Technical Specifications
Base URL	Add base URL
Key words	3.3
Application	Acrobat PDFMaker 6.0 for Word
Important Dates	
Last Modified	2007-12-21 05:51:19
Created	2007-12-19 07:17:44

8. Ex[1059] is a document entitled “PayPass – M/Chip 4 Card Technical Specification, Version 1.3.1” and was purportedly published in September 2008. I have previously reviewed and worked with “PayPass – M/Chip 4 Card Technical

Specification, Version 1.3.1” on many occasions prior to August 2012, and am familiar with its contents. Based on my review, I believe that Ex[1059] is a true and correct copy of PayPass – M/Chip 4 Card Technical Specification, Version 1.3.1 published in September 2008. I have also reviewed the PDF metadata of the PDF document submitted as Ex[1059], a screenshot of which is produced below. The dates associated with the document further confirm and corroborate my opinion.

Title	PayPass – M/Chip 4
Author	Eddy Van de Velde
Subject	Card Technical Specification
Base URL	Add base URL
Key words	Add keywords
Application	Acrobat PDFMaker 8.0 for Word
Important Dates	
Last Modified	2008-09-16 05:15:41
Created	2008-09-16 05:15:34

9. Ex[1063] is a copy of the standard specification document of International Standard ISO/IEC 14443-4, published on 15 July 2008, that I personally obtained from the webstore of American National Standards Institute (“ANSI”) at <https://webstore.ansi.org/standards/iso/isoiec144432008-1290109>. The ANSI webstore is a trusted repository for standard specification documents as published to the public and relevant industries. Further, I have previously reviewed and worked with this specification document on many occasions prior to August

2012, and am familiar with its contents. I believe that Ex[1063] is a true and correct copy of the standard specification document of International Standard ISO/IEC 14443-4, published on 15 July 2008.

10. Ex[1064] is a copy of the standard specification document of International Standard ISO/IEC 7816-4, published on 15 January 2005, that I personally obtained from the webstore of ANSI at <https://webstore.ansi.org/standards/iso/isoiec78162005>. The ANSI webstore is a trusted repository for standard specification documents as-published to the public and relevant industries. Further, I have previously reviewed and worked with this specification document on many occasions prior to August 2012, and am familiar with its contents. I believe that Ex[1064] is a true and correct copy of the standard specification document of International Standard ISO/IEC 7816-4, published on 15 January 2005.

11. Ex[1065] is a copy of the standard specification document of International Standard ISO/IEC 7813, published on 1 July 2006, that I personally obtained from the website of the International Organization for Standardization (ISO) at <https://www.iso.org/standard/43317.html>. The ISO website is a trusted repository for standard specification documents as-published to the public and relevant industries. Further, I have previously reviewed and worked with this specification document on many occasions prior to August 2012, and am familiar

with its contents. I believe that Ex[1065] is a true and correct copy of the standard specification document of International Standard ISO/IEC 7813, published on 1 July 2006.

III. EMV AND OTHER PAYMENT STANDARDS ARE EVIDENCE OF A POSITA'S KNOWLEDGE IS APPROPRIATE, AS CONFIRMED BY THE '171 PATENT ITSELF

A. EMV And PayPass Are Critical Background References That Even The '171 Patent Relies on To Supply Necessary Disclosures

12. A POSITA reading Collinge (Ex[1011]) and Fiske (Ex[1012]) (and any patent in this field of card payment) would have done so in light of their knowledge of the most common payment specifications, especially EMV, (Exs[1019]-[1029], collectively, "EMV Specifications") and the brand-specific implementations of those EMV standards, such as Mastercard's PayPass Specifications.

13. PO's expert argues that EMV had low adoption in the U.S. by 2012. Ex[2009] ¶¶37-43. This is irrelevant. U.S. adoption rate is irrelevant because a POSITA would have been aware of EMV and motivated to build products that comply with it, due to its wide adoption worldwide and a well-known trend to migrate in the U.S, which PO's expert admits. Ex[2009] ¶37. By 2012, U.S. issuers have started making cards EMV compliant so that they would work worldwide with EMV-compliant terminals—there were more than 1.2 billion EMV-compliant cards in circulation worldwide as of 2011, and with a well-known target migration date of 2015 in the U.S. as well. Ex[1041], 2; Ex[2009] ¶37.

14. I note that the '171 Patent itself expressly relies on the POSITA's knowledge of EMV, PayPass, and other specifications to provide the disclosure necessary to understand and implement the claimed invention. For instance, the '171 Patent notes that the "details of an interrogation between a point-of-sale and a digital credential...are defined in 'card specifications'... . Examples of these card specifications at the time of this filing are MASTERCARD **PAYPASS-Mag Stripe v3.3**, Contactless **EMV...**," Ex[1001] 26:50-27:33, 49:32-39; and that "[m]any point-of-sale interrogation sequences carried out at the time of this filing are based on card specifications such as ... MASTERCARD, PAYPASS, MChip ... Contactless EMV." *Id.* 31:57-64; 25:50-62.

15. Indeed, PayPass-MagStripe explicitly states that "[i]t is **assumed that readers** [of these Specifications] **already have an understanding of**...ISO/IEC 14443, ISO/IEC 7816-4 and **EMV** in particular." Ex[1058], 9. PayPass-M/Chip similarly envisions that the "PayPass – M/Chip 4 application is personalized according to the EMV Card Personalization Specification as defined in [M/CHIP4 CPS]." Ex[1059], 35. PayPass Specifications adopt the commands, messaging formats, and transaction flow that EMV Specifications provide and/or allow. PayPass relies on EMV-compliant terminals and infrastructure and is itself EMV-compliant, and I include here in *infra* Annex A examples sufficient to demonstrate PayPass's compliance with EMV.

16. A POSITA would understand the '171 Patent's reference to "PayPass Mag Stripe v3.3" and "MCHIP" in the context of contactless payment standards, to refer to MasterCard contactless payment standards (compliant with the EMV Specifications) known as the "PayPass Specifications," including, for a POSITA around the time of the priority date, both the PayPass-MagStripe Technical Specification version 3.3 ("PayPass-MagStripe") and PayPass-M/Chip Card Technical Specification version 1.3.1 ("PayPass-M/Chip"). Exs[1058]-[1059]. Note that a POSITA would understand "MagStripe" in "PayPass-MagStripe" here to refer to an EMV-compliant transactions conducted in EMV Contactless "MagStripe *mode*," including computation of application cryptogram (called "CVC3" in the case of PayPass) using a COMPUTE CRYPTOGRAPHIC CHECKSUM command. Ex[1026] §§1.5.7, 3.4.2; Ex[1058], 36.

17. PayPass-MagStripe was published in December 2007 (Ex[1058], 1), and PayPass-M/Chip in September 2008 (Ex[1059], 1), well before Collinge and the '171 Patent, and a POSITA reading either the '171 Patent or Collinge would be familiar with those Specifications. The PayPass Specifications were provided, upon request, to "[Mastercard] members and their authorized agents," as well as "[c]ustomers and vendors" and others in the field. Ex[1058], 9; *see also* Ex[1059], 5.

18. A POSITA's knowledge of the PayPass Specifications is further evidenced by numerous patent publications in card payment systems with similar or earlier priority dates, which cite PayPass as being a known payment specification. *See* Ex[1053] 10:47-52; Ex[1060] [0001]; Ex[1061] 6:60-66, 7:40-60; Ex[1062] [0134]. Indeed, the '171 Patent repeatedly refers to, and in some cases outright copies, excerpts from the PayPass Specifications into its specifications. *See* Ex[1001] 7:2-20; 24:50-54, 26:55, 29:30-33; Fig.2. I provide a comparison chart of these exemplary references at the end of this declaration as Annex B.

19. The POSITA's expected familiarity with EMV and EMV-compliant implementation is also demonstrated by numerous contemporaneous patents. *See e.g.* Ex[1052] 4:56-57; Ex[1053] 5:14-40, 10:47-52; Ex[1054] 3:33-44; Ex[1055] 45:33-35; Ex[1056] [0045]; Ex[1057] [0287].

20. Accordingly, as confirmed by the disclosure of the '171 Patent itself, a POSITA would be expected to already have background knowledge regarding EMV standard as laid out in the EMV Specifications and would find implementing payment techniques in a manner that complies with those specifications to be obvious.

21. Thus, EMV standard practices and implementations of those practices in known commercial standards, such as PayPass, are background knowledge that a POSITA would have had in reading Collinge and Fiske.

B. Collinge and Fiske Both Assume the POSITA's Knowledge of the Background Standards

22. I further note that the Collinge Provisionals explicitly state the disclosures therein are intended to be used in a “standard” PayPass transaction. Ex[1009], 8, 14, 15 22-23. Therefore, a POSITA would consider the PayPass Specifications (and EMV Specifications, with which the PayPass Specifications comply) when reading Collinge.

23. Similarly, Fiske explicitly informs the reader that it provides “improved methods ... *without requiring modification of the EMV infrastructure*, or existing PoS hardware” and repeatedly notes the techniques are to be used in conducting “an EMV transaction”, with “EMV” appearing over fifty times in the specification, including 3 times in the abstract alone. *Id.* 2:54-59; Abstract; 1:20, 57-64; 2:54-59, 67; 3:1-25, 46-62; 6:54-60; 7:20-23, Fig. 3; 8:40-42, Fig. 4; col.9, *passim*; 10:54-56; 12:49-52; 13:32-42; 15:55-58; 17:10-15. Accordingly, a POSITA reading Fiske would have invoked background knowledge about EMV Specifications when reading and implementing Fiske’s disclosures.

IV. GROUNDS OF INVALIDITY

A. Grounds 1 and 2: Collinge in View of Knowledge of a POSITA

1. Claim 1

a. [1.A-3]: “... configured to request, ... data associated with the account and”

24. I understand the PO to argue that Collinge’s request for payment credentials, which includes the CAP token to permit the server to authenticate the user’s mobile device, is somehow not a request. POR, 13-17. This is incorrect. First, Collinge discloses that the credentials are provisioned via a “pull **request** from the mobile device,” Ex[1009], claim 19. Further, a POSITA would deem **requesting** information necessary to conduct a transaction was an obvious way to obtain that data among the finite number of predictable options.

25. Moreover, a POSITA would understand that the CAP token transmission from the mobile device **is** a request for the payment credentials. Indeed, in Collinge, the transmission of the CAP token is not described as triggering any process at the server other than one for generating and returning the encrypted payment credentials to the device. Ex[1011] [0081]-[0085]. The fact that the server performs multiple steps before responding (including authenticating the CAP token to verify that the mobile device is authorized to receive the requested payment credentials), does not make the initiating request message with the CAP token anything less than a “request.”

26. Indeed, it would be obvious to configure the mobile application to *request* the credential data, as opposed to having the server push the data to the mobile device, for many reasons—for example, the “push” mode would result in a waste of bandwidth and resources. Ex[1006] ¶88.

27. None of the PO’s “alternative” methods other than “requesting” the data exist are appropriate. PO’s first purported solution, that a remote system can send data to the mobile device during processing of a transaction, is not a solution at all because a POSITA around the time of the priority date would know that mobile devices at that time often did not have network coverage in indoor or underground settings, where many transactions take place, and thus the server cannot count on reaching the mobile device during processing of a transaction through wireless network. Ex[1044] ¶¶ 4.41-4.43.

28. Then it appears that PO also argues that the remote system can send the data through the POS machine, including through the first communication channel, POR, 16, which is the channel for interrogation between the mobile device and the POS terminal (such as NFC). This also does not work, because it would require the user to hold the mobile device steadily next to the POS terminal not only for the interrogation but also for the *entire duration of authorization* while the POS terminal transmits the authorization request and waits for the authorization response from the server—this can take seconds, given the network latency and duration of the remote server’s processing. Thus, such a method would not be reliable and would undermine the whole purpose of contactless transactions, which is to allow the user to complete a transaction with a quick tap on the POS terminal.

b. [1.C]: “... non-permanent cryptographic key associated with the account;”

29. I understand that PO argues that Collinge’s non-permanent cryptographic key (KS_{UN}) is not “associated with the account” because a POSITA could not recover the account number from the key. This is not what the claim requires. The claims do not require that the account number be **known** or **recoverable** from the key, only that the key be “associated with the account.”

30. As the Collinge Provisionals disclose, KS_{UN} is uniquely, mathematically, and deterministically associated with the account, because it is generated from a card-level key KD_{UN} , which is in turn generated from the account number (PAN), and the DES3 encryption-algorithm operation at each step results an outcome unique to the input. Ex[1009] [0069]-[0071]. In other words, a POSITA would know, mathematically, that a different PAN would result in a different KD_{UN} , which would in turn result in a different KS_{UN} , even if other variables (e.g. the transaction counter ATC) are the same.

31. PO’s expert appears to assert that KS_{UN} is not associated with an account because Collinge’s value “Y”, an input for generation of KS_{UN} (Ex[1009] 26-27), does not contain the full PAN. Ex[1051] 164:6-18. That is not correct. With a typical 16-digit credit or debit card PAN, Collinge’s value “X” would be 18-digits after adding a two-digit PSN (Ex[1021], 45, 150 (showing length of PSN being 2 digits)). Taking the 16 rightmost digits of X to obtain Y, as taught in Collinge, would truncate the first two digits of the PAN. Ex[1009] 26-27. Because the first six digits

of a PAN identify an issuer and payment network (Ex[1065] §7.4.1), even without the first two digits, ***Y still uniquely identifies the account*** of a given issuer. Y is then encrypted using the Issuer Master Key (IMK_{UN}) ***unique to an issuer***, to derive the KD_{UN} , resulting in no loss of information. That is, a given KS_{UN} would be associated with only one account globally.

32. Accordingly, KS_{UN} is associated with the account. *Id.*

c. [1.C]: “the first set of data associated with the account usable by the application to formulate an application protocol data unit response;”

33. The encrypted payload in Collinge contains the data referred to in this limitation. Further, Collinge explicitly provides that the invention was to enable a PayPass transaction (*supra* §III.B), and a POSITA would know that PayPass is an EMV-compliant standard, which uses APDU messages (as most smart card payment systems at that time did and still do globally). *Supra* §III.A. This is evidenced by the PayPass Specifications themselves, Ex[1058] §§III.3.1, III.3.3; Ex[1059] §2.4.1, the EMV Specifications, *e.g.* Ex[1021] §§6.1-6.2, as well as textbook materials, Ex[1037], 46-47. APDU messages are standard communication format for both contactless and contact readers and payment, as defined in ISO/IEC 7816-4 standard, a fact acknowledged by the '171 Patent itself. Ex[1001] col. 6, Table 1.

34. Thus, a POSITA would be well aware that APDUs are simply standardized messages exchanged between smart cards and smart card POS

terminals, that any response message, including the cryptogram, would be conveyed via an APDU. Accordingly, a POSITA would know to implement Collinge with APDUs to ensure interoperability with the payment network.

d. [1.E-2]: “... wherein the response cryptogram does not include the local cryptographic key;”

35. Collinge, via its incorporation of the Collinge Previsions, recites an explicit formula for CVC3 generation by using KS_{UN} as an encryption key to perform DES3 cryptographic operation,

2. Calculate T as follows:

$T := DES3(KS_{UN})[M]$

The two least significant bytes of T are the $CVC3_{TRACK1/2}$

Ex[1009], 29-30. As demonstrated by the step-by-step analysis of DES generation of a cryptogram in Ex[1039], which encrypts a message $M=0123456789ABCDEF$, with key $K=133457799BBCDFF1$, resulting in encrypted text $C=85E813540F0AB405$. Ex[1039], 9, 14. Thus, it is obvious that Collinge’s CVC3 would not contain KS_{UN} . Ex[1006] ¶103.² Although PayPass-MagStripe originally provided static KD_{CVC3} keys for generation of cryptograms, Ex[1058] §III.3.8.1,

² I understand the PO as challenging the relevance of Ex[1034] and Ex[1039]. Ex[1034] demonstrates that it was common knowledge that MACs like CVC3 are calculated using DES algorithms, while Ex[1039] shows the details of the DES encryption calculation and proves that DES results in an output that does not include the key, and thus DES3, which involves three consecutive operations of DES and further scrambles the output, would not result in an output that includes the key.

Collinge teaches use of a one-time cryptographic key KS_{UN} , which provides enhanced security and remains fully compatible with PayPass-MagStripe because KS_{UN} , like KD_{CVC3} , is also a 16-byte key, Ex[1009], 28, and thus generates a CVC3 cryptogram in the same way as KD_{CVC3} that can be processed by the issuer. *Compare* Ex[1009] 29-30 (CVC3 generation) *with* Ex[1058] §III.3.8.2.

36. Indeed, it is also only common sense that the result of cryptographic calculations like DES3 would not include the key—otherwise anyone with the cryptogram could use the included key to create a valid yet fraudulent payment request—undermining the entire purpose of using encryption for this in the first place. Therefore, a POSITA would understand or find it obvious that Collinge’s CVC3 does not include the cryptographic key.

e. [1.E-3]: “device response communication comprising at least one response application data protocol unit containing the response cryptogram and an account identifier for the account.”

37. I understand that PO argues that a POSITA would not know that PayPass, which Collinge enables, is EMV-compliant or that Ex[1026] describes a “MasterCard” card interrogation. POR, 25-28. Neither is correct.

38. CVC3 and Application Cryptogram (AC) are well known in the art to be names of response cryptograms generated by the payment application for authorization of a transaction in an EMV-compliant transaction. Ex[1037], 54, 57; Ex[1052], 14:63-65. Further, a POSITA would know that Ex[1026], which relates

to “Kernel 2,” is describing POS terminal interactions with a Mastercard-branded smart card. See Ex[1026], 23 (“this document defines the behavior of the Kernel used in combination with card supporting a MasterCard brand...”).

39. A POSITA reviewing Collinge would be familiar with the PayPass Specifications, and know that PayPass is Mastercard’s implementation of contactless EMV-compliant transactions, and/or find it obvious to implement Collinge to be EMV-compliant (Ex[1026], §§3.4.2, 3.4.3, 3.8.1–3.8.3, 5.2, 5.3). This is corroborated by the PayPass Specifications, which provide the same response APDUs including the ones here under the first construction.³ Ex[1058] §II.3.1 (steps 6 & 8). Further, a POSITA would also know that in Visa transactions under EMV, in response to the GPO command, the “card” may return an AC in the same message with the Track 2 Equivalent Data with the PAN, under the second construction. Ex[1027] §§2.4.1, 5.2.2.3, 5.4.3.2, Table E-6.

40. Finally, a POSITA could easily implement Collinge based on knowledge of PayPass and EMV, since Collinge itself specifically informs the reader to conduct PayPass (which is EMV-compliant) transactions. Specifically, once the mobile device (card) receives KS_{UN} in the encrypted payload, it can generate,

³ The claim language reads “application data protocol unit” (ADPU), for purposes of my declaration in this proceeding, I believe that this term should be read as an “application protocol data unit” or APDU because there is no ambiguity what the term refers to here.

following standard practices, a response cryptogram ([1.E-2]), whether it is CVC3 as explicitly disclosed by Collinge and EMV MagStripe-mode, Ex[1009] 29-30; Ex[1026] §3.4.2, or an EMV-mode Application Cryptogram. In particular, the KS_{UN} is a 16-byte value, Ex[1009], 28; Ex[1020] § A1.3.1, and thus a POSITA would be motivated to and will be able to successfully use KS_{UN} directly as the 16-byte AC Session Key in the EMV AC formula, without modification. Ex[1020] § 8.1.2.

41. Once the card generates the response cryptogram, it is only a matter of following practices well known in the art to decide whether to send it back to the POS terminal in a separate message (Mastercard PayPass) or include it in the same response as Track1/2 data (Visa). Ex[1058] §II.3.1; Ex[1027] §§2.4.1, 5.2.2.3, 5.4.3.2, Table E-6. Accordingly, a POSITA would reasonably expect success when implementing Collinge's techniques for PayPass or EMV-mode transactions.

2. Claim 10 and 27: “remote computer system”

42. PO appears to incorrectly assert that the “remote computer system” in Collinge is limited to the red box in the PO-supplied annotation below:

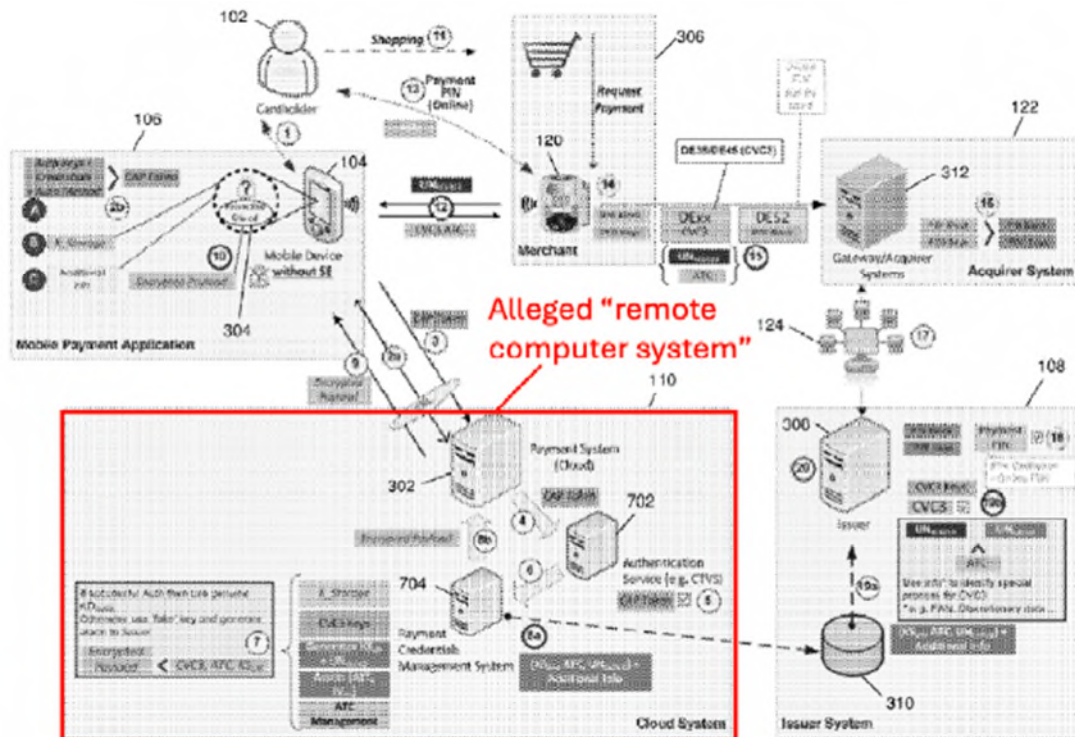


FIG. 7

43. However, my opinion was not limited by the mapping of the claimed “remote computer system” to the portion identified in Petitioner’s red box. To the contrary, the functionality performed in at least the structures identified below in the blue box in the corresponding drawing in Figure 7 of the Collinge Provisionals (which are expressly stated and shown to share a “connection 178” therebetween) correspond to the functions of the remote computer system:

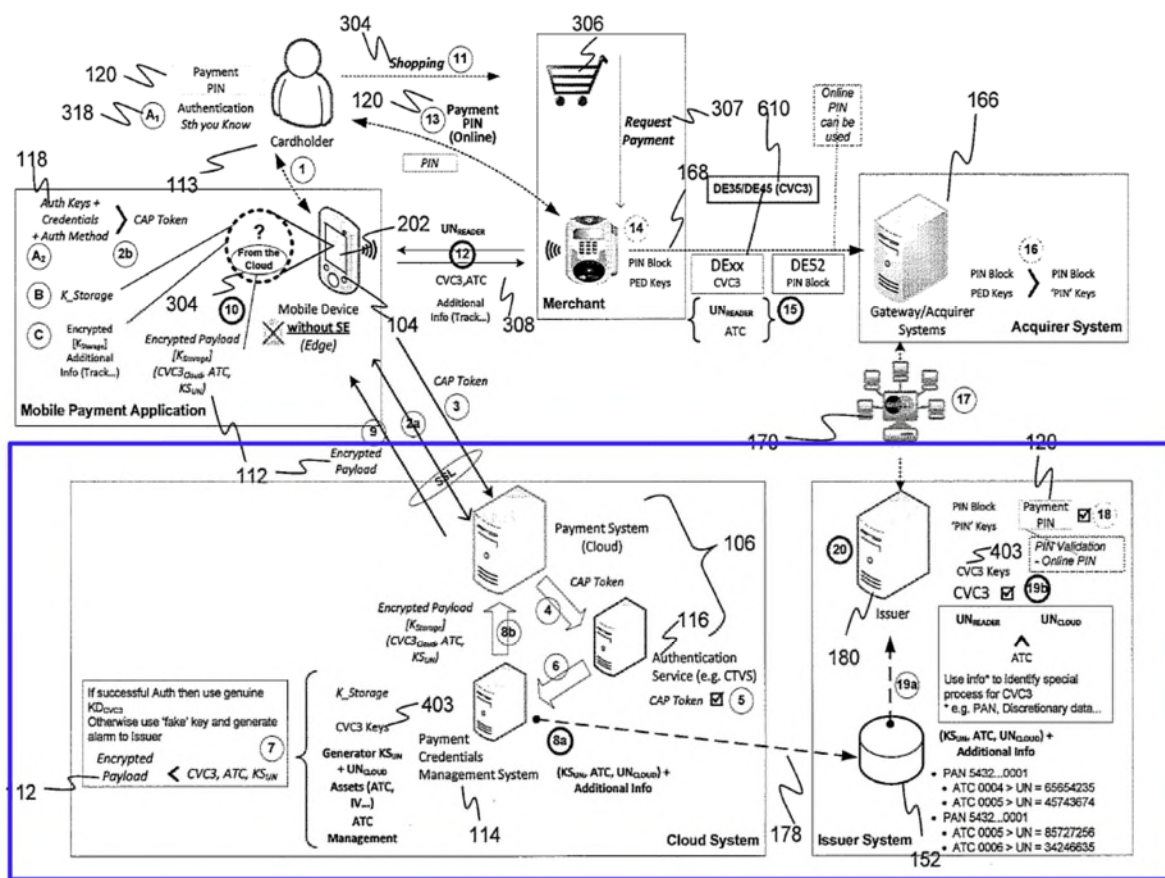


FIG. 7

Ex[1009] [0046], Fig. 7.

44. As I explained, “computer system” is a very loose term and there is no fixed number of computers such that one computer system can or should include. Ex[1049], 25:25-26:8. Further, PO mischaracterized my testimony when it states that I confirmed during deposition that “remote-SE system 110,” as shown in the red box, is what I identifies as the claimed remote computer system. POR, 30 (citing Ex[2035] at 74:15-75:9). That is wrong as the questions were directed at claim 1, and I merely acknowledged that the functionality expressly recited in claim 1 was

performed in Collinge’s remote-SE, which, as shown above, is within the blue-boxed portion of the remote computer system. Ex[1049] 73:3-75:9.

a. [10.B]/[27.B] “determining a current non-permanent cryptographic key ... using the account identifier”

45. Collinge Provisionals do not teach that retrieval of KS_{UN} is done using *only* the ATC, Ex[1009] 23, and a POSITA would understand that the issuer would necessarily need to use an account identifier such as a PAN to retrieve or re-calculate (“determine”) the correct key because, as explained above for [1.C], KS_{UN} is generated uniquely with and for a PAN and is specific to a PAN. Collinge thus merely omits what was already well known in the art at that time.

b. [10.C]/[27.C]: “calculating a calculated cryptogram using the current nonpermanent cryptographic key associated with the account”

46. I understand PO to argue that Collinge’s statement that “[m]ethods for validating a CVC3 will be apparent to persons having skill in the relevant art,” Ex[1011] [0087], serves to “expressly omit[.]” details of validation. POR 36. That is wrong. A POSITA would read that as a confirmation that validation of CVC3 was well known in the art—which it was.

47. Collinge’s statement that the technique for validating a message authentication code, such as CVC3 was “apparent” to a POSITA, is evidenced by payment-related textbook materials, which make clear that this technique for verifying a cryptogram is basic common sense in the industry. Ex[2037] 57-58.

Regardless, the most obvious way to implement Collinge would be to recalculate the CVC3 and verify that the calculated value matches the received one, because a genuine or legitimate CVC3 would be calculated by a genuine card that has the genuine credential data including the nonpermanent cryptographic key; thus, for an issuer to validate a CVC3 received from a card through a POS terminal, i.e. to determine whether that CVC3 is legitimate, it would be obvious to calculate that CVC3 using the same genuine credential data including the same cryptographic key, which is known to the issuer, and see if they match, exactly as the textbook explains and as a POSITA would know. *Id.*

3. Claim 8

a. [8.A]: “at the remote computer system ... sending the first non-permanent cryptographic key and the account identifier to the mobile device”

48. Collinge discloses receiving from a remote-SE all necessary “payment credentials” to create a transaction request message including account information (e.g. PAN).” Ex[1011] [0009], [0065]. Because the non-permanent cryptographic key is specific to an account and only usable for the associated account, *see also supra* §IV.A.1.b, it would be obvious to send an account identifier along with the key, so that the mobile device knows which account’s key it is receiving, which a POSITA would understand would be useful when designing a card capable of being

used with multiple accounts or a credential wallet that could be used with multiple issuers and accounts.

b. [8.B]: “after the interrogation, generating the second non-permanent cryptographic key at the remote computer system, associating ... and sending ...”

49. A POSITA understands that a new KS_{UN} is required for each transaction (interrogation) because it is associated with a transaction counter in addition to associated with an account. Ex[1009], 28. Further, KS_{UN} in Collinge is not generated by the mobile device (and indeed could not be, because the mobile device does not know UN_{CLOUD} —an input to the KS_{UN} generation formula) (Ex[1009] at 27-28 (“ UN_{CLOUD} is **not** sent to the Mobile device”), and is instead generated at the remote computer system. Ex[1011] [0083]. Therefore, a POSITA would readily understand that, logically and obviously, when the mobile device uses up the one-time key KS_{UN} in a transaction *after an interrogation*, it should request a new KS_{UN} from the remote computer for the next transaction, and in response, the remote computer system would generate a KS_{UN} associated with that account and send it to the mobile device in an encrypted payload in response to such a request.

4. Claims 4 and 20: “wherein the interrogation conforms to a card specification...”⁴

⁴ I assume that that, together with claim 4, PO intended to address claim 20, which overlaps with claim 4, not claim 21.

50. As explained above, Collinge Provisionals explicitly state that the invention was designed to enable PayPass transactions, where interrogations are defined by specification documents including the PayPass Specifications, *supra* §III.B, which a POSITA would understand as also evidenced by the '171 Patent itself. Ex[1058] §II.3.

51. Similarly, in my opinion, PO's argument that the Collinge provisionals fail to mention "m/chip" is irrelevant. A POSITA at that time understood that PayPass Specifications, which are card specifications, include PayPass-M/Chip, published in September 2008. *Supra* §III.A. Also, the Mastercard MCHIP (PayPass-MChip) specification, which a POSITA understands to include the PayPass-MChip Specifications in the context of contactless payment, Ex[1059], is admitted prior art for contactless payment in the '171 Patent. Ex[1001] at 25:50-63, 31:55-60.

52. I disagree with Patent Owner and Dr. Easttom's assertion that card specifications "often" provide only a "loose framework" and thus require modification as being both vague and irrelevant—they fail to point to any point in the PayPass or EMV Specifications that actually requires modification, or how such modification would result in non-compliance with the specification. POR, 42-43, Ex[2009] ¶ 128. On the contrary, because interrogation is a highly standardized process, once the card obtains the key from the encrypted payload, it is only a matter of following standard practices well known in the art to generate the responses

including the cryptogram and respond to POS commands. Ex[1058] §II.3.1; Ex[1027] §§ 2.4.1, 5.2.2.3, 5.4.3.2, Table E-6. Indeed, a POSITA would understand stages 11 to 14 of Collinge’s transaction flow, Ex[1011] [0085]-[0086], Ex[1009] 22-23, as carrying out a “standard” interrogation between the mobile device and the POS terminal—because a “standard shopping experience” will likely encounter a standard card payment infrastructure, whose behavior is defined by specification documents such as EMV Specifications, and failure to conform to a card specifications would result in failure of transaction, and/or inability to use the card, which would be a bad cardholder experience, as is understood by a POSITA.

53. Thus, a POSITA would not require mapping of each set of inputs or responses onto a specification to understand or find it obvious that Collinge discloses interrogations conforming to PayPass (hence EMV) specifications—Collinge and Collinge Provisionals’ reference to PayPass and m/chip are more than sufficient to convey that to a POSITA.

5. Claim 15 and 32:

Supra §IV.A.4.

6. Claims 6 and 22: “...the response cryptogram comprises encrypting ... according to ...(DES) ...wherein the response cryptogram includes a portion, but not all, of the output”

54. The Collinge Provisionals disclose that the “DES3” algorithm is used to calculate the cryptogram. Ex[1009], [0071].

2. Calculate T as follows:

$$T := \text{DES3}(KS_{UN})[M]$$

55. PO's argument that other algorithms might also be used, even if true, does not make the use of DES3 for purpose of calculating the MAC/cryptogram, as taught in both Collinge and the EMV specification, any less obvious.

56. Notably, Dr. Easttom has not named a single card payment specification that uses the "alternative" algorithms he proposes to generate a response cryptogram. Ex[2009] ¶¶132-135. A POSITA does not need to, as PO argues, "cho[os]e" DES, POR 46, when the standard specifications and Collinge already did.

7. Claims 16 and 33: "wherein the first and second non-permanent cryptographic key are associated with an issuer master key"

57. The use of Issuer Master Keys to derive keys used to generate MAC cryptograms for payments is a fundamental principle known to POSITAs, as evidenced by the prior art, standard specifications, and textbook materials. Ex[1009], 25-26; Ex[1020] §§8.1.2, A1.3.1, A1.4; Ex[1037], 58. Whether or not EMV "mandates" a specific algorithm does not detract from a POSITA's knowledge.

58. PO again mischaracterized my testimony here—I did not say that Collinge departed from EMV's key derivation approach of using issuer master keys in session key derivation. *Cf.* POR, 48-49. I merely noted that Collinge uses an

additional piece of data (UN_{CLOUD}) in that process, which is not prohibited by the EMV standard. Ex[1049] 77:12-78:17; Ex[1020] §§8.1.2, A1.3-A1.4. Fundamentally, a POSITA would know or find this element obvious because the issuer must control the ultimate key (IMK) to derive all other keys to validate cryptograms and approve transactions, for security and efficiency reasons.

B. Grounds 3 and 4: Claims 1-33 Are Anticipated and Rendered Obvious by *Fiske* in View of Knowledge of a POSITA

1. Claim 8, 9

59. Fiske, just like Collinge in ground 1, involves receiving “‘digital credential data’ required for authorization” including the primary account number (PAN) from the remote server. This PAN would be included with the cryptogram by the card when requesting payment authorization. See Ex[1012], 11:34-44 (discussing provisioning the user device with data sufficient to make a purchase); 3:44-46 (discussing PAN sent to terminal from the mobile application); 4:49-41 (discussing the mobile application sending the account number to the terminal during a transaction).

60. Like Collinge, the non-permanent cryptographic key in Fiske (ICC Session Key) is specific to an account and only usable for that account. Ex[1012], 12:45-52, 12:50-57, 13:17-24. Therefore, a POSITA would find it obvious to send an account identifier to the mobile device along with the key, so that the mobile device knows for certain which account’s key it is receiving. See *supra* §IV.A.3.a.

2. Claims 10-13, 27-30

a. “the authorization request comprising authorization data, wherein the authorization data comprises the response cryptogram and the account identifier”

61. PO’s argument here appears to ignore the analysis of this element incorporated from the Petition’s analysis of claim 1. As noted above, the Petition explains that the device is provisioned with all data necessary to perform a transaction, which includes both the PAN (needed by the issuer to associate the transaction with a particular account for correct financial tracking) and the cryptogram/MAC (which the issuer uses to verify the authenticity of the request). *See supra* §IV.B.1; Ex[1037] 57.

62. Indeed, a POSITA would immediately realize that an authorization request would of course include an account identifier—otherwise the issuing bank, having only information like transaction amount or currency, would not know which account this transaction needs to be charged against. Additionally, as explained in the next section, *infra* Part IV.B.2.b, the issuing bank also needs the account information to determine the key in order to calculate (and thus validate) the Application Cryptogram ARQC in authorizing a transaction. Therefore, a POSITA reading the disclosures of Fiske, which discusses issuing bank processing including verifying ARQC and authorization of a transaction, Ex[1012] 7:58-60, 14:45-63, 15:59-16:30, Fig. 7, would know as a matter of course or find it obvious that the

authorization request would need to include a PAN. This is further corroborated by textbook materials teaching standard practice in card payment systems, which explicitly lists the PAN as part of the authorization request. Ex[1037] 57.

b. “determining a current non-permanent cryptographic key associated with the account using the account identifier”

63. Fiske expressly discloses that the issuer identifies “which Session Key was used” when determining whether to authorize the transaction. Ex[1012], 16:13-14 and Fig. 7. Figure 7, in turn, shows that the issuer processing of authorization request requires examination of the received ARQC, an Application Cryptogram. A POSITA would understand that the issuer would necessarily need to use an account identifier such as a PAN to retrieve or re-calculate (“determine”) the correct key because the Session Key is generated uniquely for a PAN and is specific to a PAN. *See also* Ex[1012] 8:30-67 & Fig. 4 (illustrating “process of deriving an ICC session key from a unique ICC Master Key **according to the EMV protocols**”); Ex[1020], §§A1.3-1.4 (describing EMV protocol for key generation using PAN as input in generation method).

c. “calculating a calculated cryptogram”

64. For the same reason as discussed for Collinge, the issuer’s verification of the received MAC/cryptogram by recalculating the expected cryptogram and comparing it to the received one is ancient and obvious. Ex[1037] 58; *supra* §IV.A.2.b.

3. Claims 11 and 28: “the set of inputs comprises unpredictable data; the authorization data comprises the unpredictable data; and the calculated cryptogram is calculated using the unpredictable data.”; Claims 12 and 19: “wherein the POS command communications comprise the unpredictable data”

65. Regardless of whether the EMV specifications are incorporated by reference in Fiske, they are part of the POSITA’s background knowledge and must be considered when interpreting Fiske’s disclosure. *Supra* §§III.A-III.B; Ex[1012] 2:54-59 (Fiske stating his invention works “without requiring modification of the EMV infrastructure.”).

66. Also, regardless of whether the portions of the EMV specification are mandatory or merely suggested, a POSITA would understand the EMV specifications to show that this claim element merely recites the most-well known, standard way to generate a payment cryptogram, as the EMV Specifications explicitly list “Unpredictable Number” among input data from the terminal used to generate the cryptogram (Ex[1020] §8.1.1), which is repeated by the issuer when verifying the cryptogram. Ex[1037] 57-58. A POSITA thus understands that the “response cryptogram” is the cryptogram generated by the mobile device (card) in response to a COMPUTE CRYPTOGRAPHIC CHECKSUM, a GENERATE AC, or a GPO command. Ex[1026] §§3.4.2, 3.4.3, 5.2, 5.3; Ex[1027] §2.4.1. This knowledge is further evidenced by EMV Specifications cited above and renders this element known or obvious to a POSITA.

4. Claims 13 and 30: “wherein the authorization data further comprises values from the transaction”

67. The point of the authorization in any “payment system” (the title of Fiske’s invention) is for the account issuer to authorize a transaction, which means the issuer must receive the transaction data, such as the amount user is attempting to spend, and in which currency, to determine whether to permit the payment. Ex[1006] ¶214. Indeed, this is corroborated by textbook evidence. Ex[1037] 58 (Authorization request includes “[a]ll the data objects necessary to *reproduce the transaction claim.*”).

68. Dr. Easttom’s opinion here, which states that “[a] POSITA would have recognized that validation of a cryptogram (e.g., using the response cryptogram) would not require any ‘values from the transaction,’” Ex[2009] ¶234, is both technically wrong and irrelevant. It is wrong because an Application Cryptogram (AC) is indeed generated with values from the transaction, Ex[1020] §§8.1.1-8.1.2, and thus these values must be included in an authorization request for the issuer to re-generate the AC and verify it. Dr. Easttom’s opinion is irrelevant because the claim language speaks of what is included in the authorization data, not what is required for validation of cryptogram, on which he purports to opine. His further opinion that “if ‘values from the transaction’ were needed for a different purpose, e.g., to determine if a payment amount is within acceptable limits, that information need not be included as part of an ‘authorization request’ with the ‘response

cryptogram and the account identifier,” *id.*, is wrong, for the reasons explained—the whole point of an “authorization request” is to ask for authorization of a transaction, and the issuer will not be able to authorize a transaction without knowing what the transaction is, including how much money should be charged to which account.

5. Claims 1 and 17

a. [1.E-2]: “wherein the response cryptogram does not include the local cryptographic key associated with the account used to generate the response cryptogram”

69. Because the Application Cryptogram (AC) is generated using the Session Key as encryption key, it is common-sense knowledge for any person, and certainly a POSITA, even without knowing *how* the AC is generated, that the encryption key will not be included in the resulting cryptogram; otherwise anyone with the cryptogram could use the included key to create a fraudulent payment request, which undermines the entire purpose of using cryptographic techniques in the first place. *Supra* §IV.A.1.d.

70. In any event, a POSITA knows that EMV uses DES3 to generate an AC, which does not contain the key. Ex[1020] §§A1.2, B1.1. As demonstrated by the step-by-step analysis of DES generation of a cryptogram in Ex[1039], which encrypts a message M=0123456789ABCDEF, with key K=133457799BBCDFF1, resulting in encrypted text C=85E813540F0AB405. Ex[1039], 9, 14. DES3, which

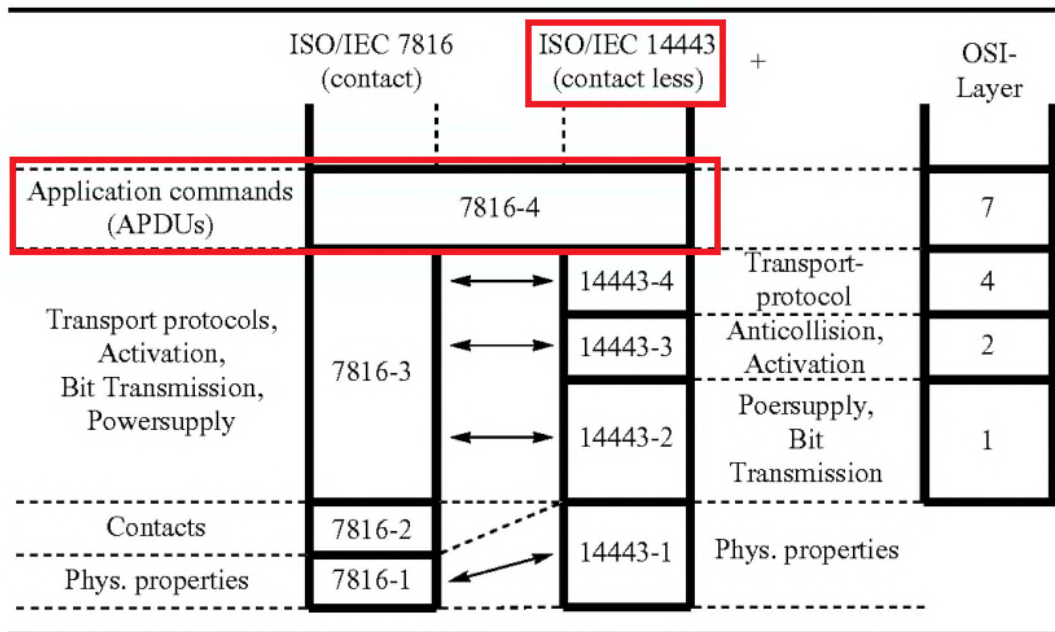
involves three operations of DES, further scrambles the output and certainly does not contain the key. Again, despite naming numerous encryption algorithms, PO and its expert could not identify a single algorithm where the resulting encrypted text (cryptogram) includes the key. POR 46, 63-64.

b. [1.E-3]: “at least one response application data protocol unit containing the response cryptogram and an account identifier for the account”

71. It is well known to a POSITA in the field that it is essentially a universal practice to communicate with payment terminals using APDUs, a fact supported by specifications and textbook evidence. Ex[1058] §II.3.1; Ex[1021] §6.5; Ex[1026] Table 2.2; Ex[1037] 55-56.

72. EMV contactless transactions use these same long-established communication protocols, including the APDU features claimed here. *Compare e.g.* Ex[1026] §5 and Ex[1058] §§III.2-III.3 with Ex[1021] §6.5 (using contact APDU commands and responses for contactless). Specifically, contactless cards still depend on APDUs from ISO/IEC7816-4 for messaging, as explicitly stated in ISO/IEC14443 for contactless card and reflected in contactless EMV Specifications as well. Ex[1063], 6; Ex[1024], 9. Even the '171 Patent itself acknowledges this relationship between standards. Ex[1001] Table 1 (annotated).

TABLE 1



c. [1.E-3]: EMV mode Visa transactions

73. EMV mode Visa transactions satisfy this element under the second construction of the R-APDU Term. My previous declaration made clear that “[f]or similar reasons, *as Grounds 1-2, [1.E-3]*, a POSA would have found this element obvious.” Ex[1006] ¶184. In Grounds 1-2, [1.E-3] of my previous declaration, I explained how an EMV mode Visa transaction worked—specifically, the mobile device (card) would return both Track 2 Equivalent Data (including account number) and the cryptogram in the same R-APDU message in response to a GPO command. Ex[1006] ¶109. This explanation applies to Collinge as well as to Fiske, because it explains how *EMV* works, and Fiske enables EMV transactions. *Supra* §III.B.

74. I understand that PO raised a question of “why” a POSITA would choose to implement Fiske (a patent assigned to Visa) to conduct an EMV mode Visa transaction. POR, 69-70. Fiske explicitly says that—the patent discloses an invention to provide “improved methods ... ***without requiring modification of the EMV infrastructure***, or existing PoS hardware” and repeatedly notes the techniques are to be used in conducting “an EMV transaction.” Ex[1012], 2:54-59. In addition, a POSITA would recognize the benefit of offering a system that would be interoperable with one of the most prevalent payment brands in the world, such that Visa card issuers and card holders could take advantage of the offering easily.

75. Finally, PO appears to argue that EMV mode Visa transaction is not consistent with Fiske, by pointing to a section of the standard that is neither cited by my previous declaration nor relevant in any way to the issues in this case. The Section PO points to (Section 6.1 of Ex[1027]) relates to a processing flow where the payment card is required to be presented twice to the terminal in order to perform “update processing.” Ex[1027], 83-84. PO fails to explain why the fact that “update processing” does not use a GENERATE AC command has any relevance to the arguments here. Regardless, I have explained above that Visa-mode contactless transactions rely on the cryptogram generation occurring in response to the Get Processing Options (GPO) command.

6. Claims 4, 5, 20, and 21

76. PO's only argument here appears to be based on EMV not being incorporated by reference. As explained repeatedly above, a POSITA reading Fiske would understand that it would implement Fiske with standard EMV practices and thus knows or finds it obvious to conform with EMV Specifications. *Supra* §III.B.

7. Claims 6, 15, 22 and 32

77. PO's argues here that a POSITA's knowledge of portions of the EMV Specification that are "not mandatory" but are the only implementations discussed therein, somehow does not render those implementations obvious. That does not appear correct to me, because knowledge or obviousness to a POSITA is not dependent on whether any portion is "mandatory," and, in any event, those portions are widely implemented. Because Fiske explicitly calls for standard implementation of EMV and thus a POSITA would know or find it obvious to generate the cryptogram under standard EMV methods

V. CONCLUSION

78. Claims 1-33 of the '171 Patent are invalid, for reasons stated above.

VI. DECLARATION

I hereby declare that all statements made herein of my own knowledge are true and all statements made on information and belief are believed to be true. I further understand that willful false statements and the like are punishable by fine or imprisonment, or both under Section 1001 of Title 18 of the United States Code. I declare under penalty of perjury that the foregoing is true and correct.

Executed on 20 September, 2024.

SS Fiske

Stuart Fiske

ANNEX A

Examples of Excerpts Showing PayPass Compliance with EMV

Aspect / Notes	PayPass-MagStripe Ex[1058]	EMV (MagStripe Mode)
Transaction flow Shows both specifications perform a select AID and a GPO to determine whether the transaction will continue in Mag stripe mode or in EMV mode.	From § II.3.1 at 23-24 1. The <i>PayPass</i> reader issues the SELECT AID command to select the <i>PayPass</i> – Mag Stripe application. 2. The card responds with the <i>FCI</i> . The <i>FCI</i> may contain the <i>PDOL</i> . The <i>PDOL</i> is a list of tags and lengths of reader-resident data objects needed by the <i>PayPass</i> card in the GET PROCESSING OPTIONS command. 3. The <i>PayPass</i> reader issues the GET PROCESSING OPTIONS command. If there is no <i>PDOL</i> in the <i>FCI</i> , then the <i>PayPass</i> reader uses the command data field ‘8300’. Otherwise the command data field contains a <i>Command Template</i> (tag ‘83’) comprising the concatenated list of data objects resulting from processing the <i>PDOL</i> . 4. The card returns the <i>AIP</i> and the <i>AFL</i> . The <i>AFL</i> has a fixed value indicating that all data to be read by the <i>PayPass</i> reader are included in record 1 of the file with SFI 1	From EMV Book B (Ex[1024]): §3.3.3.4 Entry Point shall send the SELECT (AID) command with the ADF Name of the selected Combination (with Extended Selection if appended). From EMV Book C-2 (Ex[1026]): §3.4 Upon receipt of an ACT Signal, the Kernel initiates the transaction on the Card through a GET PROCESSING OPTIONS command. Based on the response from the Card, in particular the <i>Application Interchange Profile</i> , the Kernel continues with either a mag-stripe mode or an EMV mode transaction.

Aspect / Notes	PayPass-MagStripe Ex[1058]	EMV (MagStripe Mode)
	From §II.3.2: 3.2.1.1 The <i>PayPass</i> reader must perform the following logical functions in the order shown: 1. Get Processing Options Processing as specified in Section 3.4.1 of PART II. 2. Read Application Data as specified in Section 3.4.2 of PART II. 3. Application Version Number Checking as specified in Section 3.4.3 of PART II. 4. Compute Cryptographic Checksum Processing as specified in Section 3.4.4 of PART II. 5. Completion as specified in Section 3.4.5 of PART II.	From <i>id.</i> §3.4.2: For a mag-stripe mode transaction, after the GET PROCESSING OPTIONS command, the Kernel continues with the following steps: 1. It reads the data records from the Card, containing <i>Track 1 Data</i> and <i>Track 2 Data</i>, together with instruction on how to populate the discretionary data. 2. It issues the COMPUTE CRYPTOGRAPHIC CHECKSUM command, including <i>Unpredictable Number (Numeric)</i> to the Card, requesting the Card to return a CVC3 cryptogram, calculated over <i>Unpredictable Number (Numeric)</i>. 3. It populates the <i>Track 2 Data</i> with the <i>Unpredictable Number (Numeric)</i>, the <i>Application Transaction Counter</i>, and <i>CVC3 (Track2)</i>. 4. If <i>Track 1 Data</i> is present, it populates the <i>Track 1 Data</i> with the <i>Unpredictable Number (Numeric)</i>, the <i>Application Transaction Counter</i>, and <i>CVC3 (Track1)</i>. 5. It sets <i>nUN</i> equal to the (meaningful) length of the <i>Unpredictable Number (Numeric)</i> and populates <i>Track 2 Data</i> and (if present) <i>Track 1 Data</i> with this value. 6. It requests the transaction to be sent online.

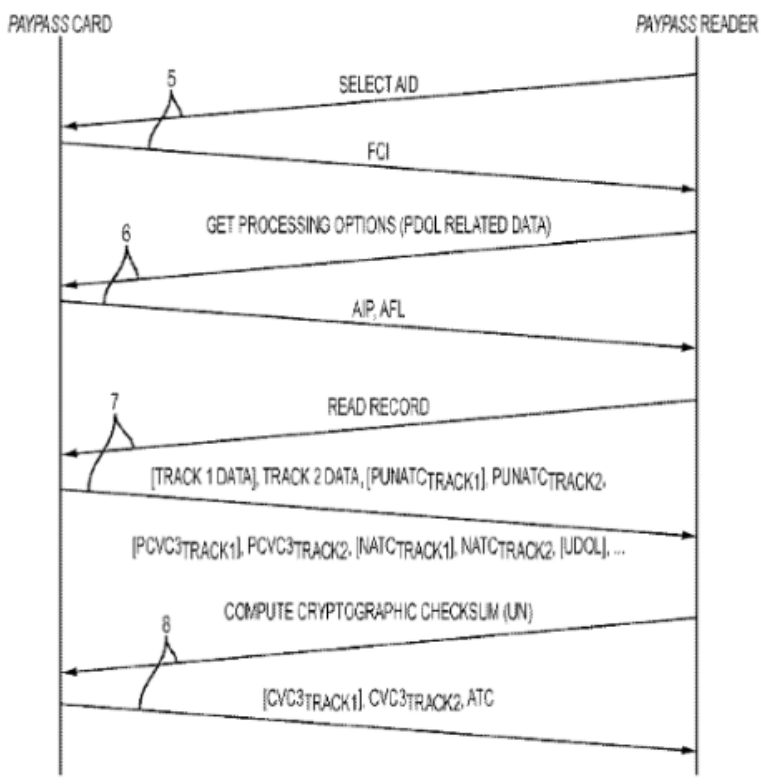
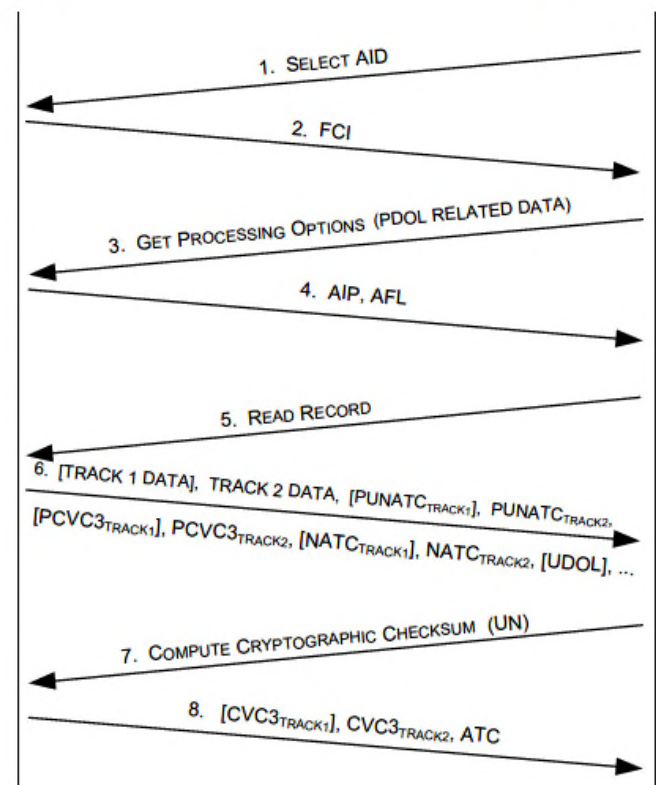
Aspect / Notes	PayPass-MagStripe Ex[1058]	EMV (MagStripe Mode)
Card Key Derivation	From §III.3.8.1: This section specifies the key derivation method used to generate the <i>ICC Derived Key for CVC3 Generation (KDCVC3)</i>. <i>KDCVC3</i> is a 16-byte DES3 key derived from the <i>Issuer Master Key for CVC3 Generation (IMKVC3)</i> as follows: 1. Concatenate from left to right the PAN (without any 'F' padding) with the PAN sequence number⁷ (if the PAN sequence number is not available, then it is replaced by a '00' byte). If the result X is less than 16 digits long, pad it to the left with hexadecimal zeros in order to obtain an eight-byte number Y in numeric (n) format. If X is at least 16 digits long, then Y consists of the 16 rightmost digits of X in numeric (n) format. 2. Compute the two eight-byte numbers: $ZL := \text{DES3}(\text{IMKVC3})[Y]$ $ZR := \text{DES3}(\text{IMKVC3})[Y \oplus ('FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF')]$ and define : $KDCVC3 := (ZL ZR)$.	From EMV Book 2 (Ex[1020]), §A1.4.1: 1. Concatenate from left to right the decimal digits of the Application PAN with the PAN Sequence Number (if the PAN Sequence Number is not present, then it is replaced by a '00' byte). If the result X is less than 16 digits long, pad it to the left with hexadecimal zeros in order to obtain an 8-byte number Y in numeric format. If X is at least 16 digits long, then Y consists of the 16 rightmost digits of X in numeric format. 2. Compute the two 8-byte numbers $ZL := \text{DES3}(\text{IMK})[Y]$ and $ZR := \text{DES3}(\text{IMK})[Y \oplus ('FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF')]$ and define $Z := (ZL ZR)$

Aspect / Notes	PayPass-MagStripe Ex[1058]	EMV (MagStripe Mode)
Cryptogram generation	From §III.3.8.2: 1. Concatenate the data listed in Table 44 in the order specified to obtain an 8 byte data block (D): Table 44—Track 1 CVC3 Data Objects Data Object Length <i>IVCVC3TRACK1</i> 2 bytes <i>UN</i> 4 bytes <i>ATC</i> 2 bytes 2. Calculate O as follows: $O := \text{DES3}(KDCVC3)[D]$ 3. The two least significant bytes of O are the <i>CVC3TRACK1</i>.	From <i>id.</i> §8.1.2 at 170: The 8-byte Application Cryptogram shall be generated using the MAC algorithm specified in Annex A1.2 and ISO/IEC 9797-1 Algorithm 3 with DES, and $s=8$. <i>id.</i> §A1.2: Cryptogram Computation Process the 8-byte blocks $X_1, X_2, \dots, X_k$ with the block cipher in CBC mode using the leftmost MAC Session Key block KSL: $H_i := \text{ALG}(KSL)[X_i \text{ xor } H_{i-1}]$, for $i = 1, 2, \dots, k$ with initial value $H_0 := ('00' \parallel '00' \parallel '00' \parallel '00' \parallel '00' \parallel '00' \parallel '00' \parallel '00')$. Compute the 8 byte block H_{k+1} in one of the following two ways. ☐ According to ISO/IEC 9797-1 Algorithm 3: $H_{k+1} := \text{ALG}(KSL)[\text{ALG-1}(KSR)[H_k]]$

Aspect / Notes	PayPass-MChip Ex[1059]	EMV (EMV Mode)												
Transaction Flow The various PayPass M/Chip transaction flows may be determined by the application states in Figure 2.1 and the initiation commands as shown in §2.2.2 A typical flow may involve: • Select(AID) to move to the SELECTED state • Get Processing Options to move to the INITIATED state • Multiple READ RECORD commands within the INITIATED state • A GENERATE AC command to move to either the ONLINE state if an ARQC is returned, or to the SCRIPT state otherwise If no SCRIPT command is received, the card application	From §§2.2.1-2.2.2 Figure 2.1—State Machine of the PayPass – M/Chip 4 Application Table 2.2—Initiation Commands for the PayPass – M/Chip Select 4 Application <table><tr><th>Initiation Commands</th><th>SW1-SW2</th></tr><tr><td>GET CHALLENGE</td><td>'9000'</td></tr><tr><td>GET DATA</td><td>'9000'</td></tr><tr><td>INTERNAL AUTHENTICATE</td><td>'9000'</td></tr><tr><td>VERIFY</td><td>'9000', '6983', '63Cx'</td></tr><tr><td>READ RECORD</td><td>'9000'</td></tr></table>	Initiation Commands	SW1-SW2	GET CHALLENGE	'9000'	GET DATA	'9000'	INTERNAL AUTHENTICATE	'9000'	VERIFY	'9000', '6983', '63Cx'	READ RECORD	'9000'	From EMV Book C-2 (Ex[1026]) §3.4.3 For an EMV mode transaction, after the GET PROCESSING OPTIONS command, the Kernel continues with the following steps: 1. It determines which form of Offline Data Authentication to perform. 2. It reads the data records of the Card (using READ RECORD commands). If the same transaction involving the same Card is recognized in the Kernel’s internal log of torn transactions, then an attempt is made to recover the transaction – see section 3.7. 3. It performs Terminal Risk Management and Terminal Action Analysis, and selects a cardholder verification method for the transaction. 4. It requests an <i>Application Cryptogram</i> from the Card by issuing a GENERATE AC command. If a response is not received from the Card, the Kernel considers the transaction as “torn”, and stores the transaction details in its internal log of torn transactions, before terminating – see section 3.7. 5. It performs Offline Data Authentication as appropriate.
Initiation Commands	SW1-SW2													
GET CHALLENGE	'9000'													
GET DATA	'9000'													
INTERNAL AUTHENTICATE	'9000'													
VERIFY	'9000', '6983', '63Cx'													
READ RECORD	'9000'													

ANNEX B

Examples of '171 Patent's Reliance on PayPass-MagStripe

'171 Patent (Ex[1001])	PayPass-MagStripe (Ex[1058])
7:8-14 (“FIG. 2 describes the detailed commands and responses expected when a reader interrogates a MASTERCARD PAYPASS Mag Stripe Ver. 3.3 application running on a SE. ...”), Fig. 2:  FIG. 2	§II.3.1, Fig. 2 Figure 2—Command – Response APDU Flow 

'171 Patent (Ex[1001])

7:15-46 (“FIG. 3 describes example transactions at the ISO/IEC 7816-4 Application Protocol Data Unit (APDU) protocol level between a reader and a card. The transaction processing depicted in FIG. 3 is representative of the MASTERCARD PAYPASS Mag Stripe application card specification v. 3.3. ...), Fig. 3:

```

9 {
  nUN: 3
  UN: 00 00 08 99
  ATC: 00 5E
  CVC3TRACK1: 2 LEAST SIGNIFICANT BYTES OF
               DES3(KDCVC3) [77 80 00 00 08 99 00 5E]
  CVC3TRACK1: FB C7
  CVC3TRACK2: 2 LEAST SIGNIFICANT BYTES OF
               DES3(KDCVC3) [99 D3 00 00 08 99 00 5E]
  CVC3TRACK2: B8 92

  • SELECT PPSE COMMAND
  10 {
    >CARD: 00 A4 04 00 0E 32 50 41 59 2E 53 59 53 2E 44 44
          46 30 31 00
    CARD>: 6F 23 84 0E 32 50 41 59 2E 53 59 53 2E 44 44 46
          30 31 A5 11 BF 0C 0E 61 0C 4F 07 A0 00 00 00 04
          10 10 87 01 01 90 00

  • SELECT AID COMMAND
  11 {
    >CARD: 00 A4 04 00 07 A0 00 00 00 04 10 10 00
    CARD>: 6F 17 84 07 A0 00 00 00 04 10 10 A5 0C 50 0A 4D
          61 73 74 65 72 43 61 72 64 90 00

  • GET PROCESSING OPTIONS COMMAND
  12 {
    >CARD: 80 A8 00 00 02 83 00 00
    CARD>: 77 0A 82 02 00 00 94 04 08 01 01 00 90 00

  • READ RECORD COMMAND
  13 {
    >CARD: 00 B2 01 0C 00
    CARD>: 70 7F 9F 6C 02 00 01 56 3E 42 35 34 31 33 31 32
          33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50 4C
          49 45 44 2F 4E 4F 54 5E 30 39 30 36 31 30 31 33
          33 30 30 30 33 33 33 30 30 30 32 32 32 32 30
          30 30 31 31 31 31 30 9F 64 01 03 9F 62 06 00 00
          00 38 00 00 9F 63 06 00 00 00 00 E0 E0 9F 65 02
          00 0E 9F 66 02 0E 70 9F 6B 13 54 13 12 34 56 78
          48 00 D0 90 61 01 90 00 99 00 00 00 0F 9F 67 01
          03 90 00

  • COMPUTE CRYPTOGRAPHIC CHECKSUM COMMAND
  14 {
    >CARD: 80 2A 8E 80 04 00 00 08 99 00
    CARD>: 77 0F 9F 61 02 B8 92 9F 60 02 FB C7 9F 36 02 00
          5E 90 00
  }
}

```

FIG. 3**PayPass-MagStripe (Ex[1058])**

§A.4

A.4 Transaction Processing

```

nUN: 3
UN: 00 00 08 99
ATC: 00 5E
CVC3TRACK1: 2 least significant bytes of
             DES3(KDCVC3) [77 80 00 00 08 99 00 5E]
CVC3TRACK1: FB C7
CVC3TRACK2: 2 least significant bytes of
             DES3(KDCVC3) [99 D3 00 00 08 99 00 5E]
CVC3TRACK2: B8 92

• SELECT PPSE Command
>CARD: 00 A4 04 00 0E 32 50 41 59 2E 53 59 53 2E 44 44
      46 30 31 00
CARD>: 6F 23 84 0E 32 50 41 59 2E 53 59 53 2E 44 44 46
      30 31 A5 11 BF 0C 0E 61 0C 4F 07 A0 00 00 00 04
      10 10 87 01 01 90 00

• SELECT AID Command
>CARD: 00 A4 04 00 07 A0 00 00 00 04 10 10 00
CARD>: 6F 17 84 07 A0 00 00 00 04 10 10 A5 0C 50 0A 4D
      61 73 74 65 72 43 61 72 64 90 00

• GET PROCESSING OPTIONS Command
>CARD: 80 A8 00 00 02 83 00 00
CARD>: 77 0A 82 02 00 00 94 04 08 01 01 00 90 00

• READ RECORD Command
>CARD: 00 B2 01 0C 00
CARD>: 70 7F 9F 6C 02 00 01 56 3E 42 35 34 31 33 31 32
      33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50 4C
      49 45 44 2F 4E 4F 54 5E 30 39 30 36 31 30 31 33
      33 30 30 30 33 33 33 30 30 30 32 32 32 32 30
      30 30 31 31 31 31 30 9F 64 01 03 9F 62 06 00 00
      00 38 00 00 9F 63 06 00 00 00 00 E0 E0 9F 65 02
      00 0E 9F 66 02 0E 70 9F 6B 13 54 13 12 34 56 78
      48 00 D0 90 61 01 90 00 99 00 00 00 0F 9F 67 01
      03 90 00

• COMPUTE CRYPTOGRAPHIC CHECKSUM Command
>CARD: 80 2A 8E 80 04 00 00 08 99 00
CARD>: 77 0F 9F 61 02 B8 92 9F 60 02 FB C7 9F 36 02 00
      5E 90 00

```

'171 Patent (Ex[1001])	PayPass-MagStripe (Ex[1058])																																																																																																																																																																																																																																
7:49-63 (“FIG. 4 describes how the reader 21 in FIG. 5 formats the response data received from the SE into accepted Track1 and Track2 data strings that are used in existing processing systems. ...”), Fig. 4: THE PAYPASS READER GENERATES THE TRACK 2 DATA TO BE INCLUDED IN THE AUTHORIZATION MESSAGE AS FOLLOWS: CVC3_{TRACK2} IN BASE 10: 47250 ATC IN BASE 10: 94 => TRACK 2 DATA: <table><tr><td>54</td><td>13</td><td>12</td><td>34</td><td>56</td><td>78</td><td>48</td><td>00</td><td>D0</td><td>90</td><td>61</td><td>01</td></tr><tr><td>9</td><td>094</td><td>99</td><td>899</td><td>250</td><td></td><td></td><td>3</td><td>F</td><td></td><td></td><td></td></tr><tr><td>↑↑↑</td><td>↑↑↑</td><td>↑↑↑</td><td>↑↑↑</td><td>↑↑↑</td><td></td><td></td><td>↑</td><td></td><td></td><td></td><td></td></tr><tr><td>ATC</td><td>UN</td><td>CVC3_{TRACK2}</td><td>nUN</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> THE PAYPASS READER GENERATES THE TRACK 1 DATA TO BE INCLUDED IN THE AUTHORIZATION MESSAGE AS FOLLOWS: CVC3_{TRACK1} IN BASE 10: 64455 ATC IN BASE 10: 94 => TRACK 1 DATA: <table><tr><td>42</td><td>35</td><td>34</td><td>31</td><td>33</td><td>31</td><td>32</td><td>33</td><td>34</td><td>35</td><td>36</td><td>37</td><td>38</td><td>34</td><td>38</td><td>30</td></tr><tr><td>30</td><td>5E</td><td>53</td><td>55</td><td>50</td><td>50</td><td>4C</td><td>49</td><td>45</td><td>44</td><td>2F</td><td>4E</td><td>4F</td><td>54</td><td>5E</td><td>30</td></tr><tr><td>39</td><td>30</td><td>36</td><td>31</td><td>30</td><td>31</td><td>33</td><td>33</td><td>34</td><td>35</td><td>35</td><td>33</td><td>33</td><td>33</td><td>30</td><td>39</td></tr><tr><td>34</td><td>32</td><td>32</td><td>32</td><td>32</td><td>32</td><td>38</td><td>39</td><td>39</td><td>31</td><td>31</td><td>31</td><td>31</td><td>31</td><td>33</td><td></td></tr></table> ASCII DUMP: B5413123456784800^SUPPLIED/NOT^0906101 334553330942222289911113 ↑↑↑ ↑↑↑ ↑↑↑ ↑ CVC3_{TRACK1} ATC UN nUN FIG. 4	54	13	12	34	56	78	48	00	D0	90	61	01	9	094	99	899	250			3	F				↑↑↑	↑↑↑	↑↑↑	↑↑↑	↑↑↑			↑					ATC	UN	CVC3 _{TRACK2}	nUN									42	35	34	31	33	31	32	33	34	35	36	37	38	34	38	30	30	5E	53	55	50	50	4C	49	45	44	2F	4E	4F	54	5E	30	39	30	36	31	30	31	33	33	34	35	35	33	33	33	30	39	34	32	32	32	32	32	38	39	39	31	31	31	31	31	33		§A.4 The PayPass reader generates the track 2 data to be included in the authorization message as follows: CVC3_{TRACK2} in base 10: 47250 ATC in base 10: 94 => track 2 data: <table><tr><td>54</td><td>13</td><td>12</td><td>34</td><td>56</td><td>78</td><td>48</td><td>00</td><td>D0</td><td>90</td><td>61</td><td>01</td></tr><tr><td>9</td><td>094</td><td>99</td><td>899</td><td>250</td><td></td><td></td><td>3</td><td>F</td><td></td><td></td><td></td></tr><tr><td>↑↑↑</td><td>↑↑↑</td><td>↑↑↑</td><td>↑↑↑</td><td>↑↑↑</td><td></td><td></td><td>↑</td><td></td><td></td><td></td><td></td></tr><tr><td>ATC</td><td>UN</td><td>CVC3_{TRACK2}</td><td>nUN</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> The PayPass reader generates the track 1 data to be included in the authorization message as follows: CVC3_{TRACK1} in base 10: 64455 ATC in base 10: 94 => track 1 data: <table><tr><td>42</td><td>35</td><td>34</td><td>31</td><td>33</td><td>31</td><td>32</td><td>33</td><td>34</td><td>35</td><td>36</td><td>37</td><td>38</td><td>34</td><td>38</td><td>30</td></tr><tr><td>30</td><td>5E</td><td>53</td><td>55</td><td>50</td><td>50</td><td>4C</td><td>49</td><td>45</td><td>44</td><td>2F</td><td>4E</td><td>4F</td><td>54</td><td>5E</td><td>30</td></tr><tr><td>39</td><td>30</td><td>36</td><td>31</td><td>30</td><td>31</td><td>33</td><td>33</td><td>34</td><td>35</td><td>35</td><td>33</td><td>33</td><td>33</td><td>30</td><td>39</td></tr><tr><td>34</td><td>32</td><td>32</td><td>32</td><td>32</td><td>32</td><td>38</td><td>39</td><td>39</td><td>31</td><td>31</td><td>31</td><td>31</td><td>31</td><td>33</td><td></td></tr></table> ASCII dump: B5413123456784800^SUPPLIED/NOT^0906101 334553330942222289911113 ↑↑↑ ↑↑↑ ↑↑↑ ↑ CVC3_{TRACK1} ATC UN nUN	54	13	12	34	56	78	48	00	D0	90	61	01	9	094	99	899	250			3	F				↑↑↑	↑↑↑	↑↑↑	↑↑↑	↑↑↑			↑					ATC	UN	CVC3 _{TRACK2}	nUN									42	35	34	31	33	31	32	33	34	35	36	37	38	34	38	30	30	5E	53	55	50	50	4C	49	45	44	2F	4E	4F	54	5E	30	39	30	36	31	30	31	33	33	34	35	35	33	33	33	30	39	34	32	32	32	32	32	38	39	39	31	31	31	31	31	33	
54	13	12	34	56	78	48	00	D0	90	61	01																																																																																																																																																																																																																						
9	094	99	899	250			3	F																																																																																																																																																																																																																									
↑↑↑	↑↑↑	↑↑↑	↑↑↑	↑↑↑			↑																																																																																																																																																																																																																										
ATC	UN	CVC3 _{TRACK2}	nUN																																																																																																																																																																																																																														
42	35	34	31	33	31	32	33	34	35	36	37	38	34	38	30																																																																																																																																																																																																																		
30	5E	53	55	50	50	4C	49	45	44	2F	4E	4F	54	5E	30																																																																																																																																																																																																																		
39	30	36	31	30	31	33	33	34	35	35	33	33	33	30	39																																																																																																																																																																																																																		
34	32	32	32	32	32	38	39	39	31	31	31	31	31	33																																																																																																																																																																																																																			
54	13	12	34	56	78	48	00	D0	90	61	01																																																																																																																																																																																																																						
9	094	99	899	250			3	F																																																																																																																																																																																																																									
↑↑↑	↑↑↑	↑↑↑	↑↑↑	↑↑↑			↑																																																																																																																																																																																																																										
ATC	UN	CVC3 _{TRACK2}	nUN																																																																																																																																																																																																																														
42	35	34	31	33	31	32	33	34	35	36	37	38	34	38	30																																																																																																																																																																																																																		
30	5E	53	55	50	50	4C	49	45	44	2F	4E	4F	54	5E	30																																																																																																																																																																																																																		
39	30	36	31	30	31	33	33	34	35	35	33	33	33	30	39																																																																																																																																																																																																																		
34	32	32	32	32	32	38	39	39	31	31	31	31	31	33																																																																																																																																																																																																																			
19:3-14: “The MASTERCARD PAYPASS card specification indicates that the response to the Select PPSE APDU 83 will always be the same: a. APDU to the card for Select PPSE APDU: 00 A4 04 00 0E 32 50 41 59 2E 53 59 53 2E 44 44 46 30 31 00 b. APDU response from the card for Select PPSE APDU is ALWAYS: 6F 23 84 0E 32 50 41 59 2E 53 59 53 2E 44 44 46 30 31 A5 11 BF 0C 0E 61 0C 4F 07 A0 00 00 00 04 10 10 87 01 01 90 00”	§A.4 SELECT PPSE Command >CARD: 00 A4 04 00 0E 32 50 41 59 2E 53 59 53 2E 44 44 46 30 31 00 CARD>: 6F 23 84 0E 32 50 41 59 2E 53 59 53 2E 44 44 46 30 31 A5 11 BF 0C 0E 61 0C 4F 07 A0 00 00 00 04 10 10 87 01 01 90 00																																																																																																																																																																																																																																

'171 Patent (Ex[1001])	PayPass-MagStripe (Ex[1058])																																								
	2.3.3 SELECT PPSE Command Message The SELECT command message for the PPSE is coded according to Table 23. Table 23—SELECT PPSE Command Message <table><tr><th>Code</th><th>Value</th></tr><tr><td>CLA</td><td>'00'</td></tr><tr><td>INS</td><td>'A4'</td></tr><tr><td>P1</td><td>'04'</td></tr><tr><td>P2</td><td>'00'</td></tr><tr><td>Lc</td><td>'0E'</td></tr><tr><td>Data</td><td>'325041592E5359532E4444463031'</td></tr><tr><td>Le</td><td>'00'</td></tr></table> The data field of the command message contains the PPSE directory name ("2PAY.SYS.DDF01"). Response Message The data field of the response message contains the <i>FCI</i> and is coded according to Table 24. Table 24—FCI Returned in Response Message of SELECT PPSE <table><tr><th>Data Object</th><th>Tag</th><th>Length</th></tr><tr><td><i>FCI Template</i></td><td>'6F'</td><td>33–44</td></tr><tr><td><i>DF Name</i> ('325041592E5359532E4444463031')</td><td>'84'</td><td>14</td></tr><tr><td><i>FCI Proprietary Template</i></td><td>'A5'</td><td>15–26</td></tr><tr><td><i>FCI Issuer Discretionary Data</i></td><td>'BF0C'</td><td>12–23</td></tr><tr><td><i>Directory Entry</i></td><td>'61'</td><td>10–21</td></tr><tr><td><i>ADF Name</i></td><td>'4F'</td><td>5–16</td></tr><tr><td><i>Application Priority Indicator</i> ('01')</td><td>'87'</td><td>1</td></tr></table> The <i>FCI</i> of the PPSE is stored in the card during the pre-personalization phase.	Code	Value	CLA	'00'	INS	'A4'	P1	'04'	P2	'00'	Lc	'0E'	Data	'325041592E5359532E4444463031'	Le	'00'	Data Object	Tag	Length	<i>FCI Template</i>	'6F'	33–44	<i>DF Name</i> ('325041592E5359532E4444463031')	'84'	14	<i>FCI Proprietary Template</i>	'A5'	15–26	<i>FCI Issuer Discretionary Data</i>	'BF0C'	12–23	<i>Directory Entry</i>	'61'	10–21	<i>ADF Name</i>	'4F'	5–16	<i>Application Priority Indicator</i> ('01')	'87'	1
Code	Value																																								
CLA	'00'																																								
INS	'A4'																																								
P1	'04'																																								
P2	'00'																																								
Lc	'0E'																																								
Data	'325041592E5359532E4444463031'																																								
Le	'00'																																								
Data Object	Tag	Length																																							
<i>FCI Template</i>	'6F'	33–44																																							
<i>DF Name</i> ('325041592E5359532E4444463031')	'84'	14																																							
<i>FCI Proprietary Template</i>	'A5'	15–26																																							
<i>FCI Issuer Discretionary Data</i>	'BF0C'	12–23																																							
<i>Directory Entry</i>	'61'	10–21																																							
<i>ADF Name</i>	'4F'	5–16																																							
<i>Application Priority Indicator</i> ('01')	'87'	1																																							

'171 Patent (Ex[1001])	PayPass-MagStripe (Ex[1058])
19:21-29: “The MASTERCARD PAYPASS card specification indicates that the response to the Select PAYPASS AID APDU 84 always be the same: a. APDU to the card for Select PAYPASS AID APDU: 00 A4 04 00 07 A0 00 00 00 04 10 10 00 b. APDU response from the card for Select AID APDU is ALWAYS: 6F 17 84 07 A0 00 00 00 04 10 10 A5 0C 50 0A 4D 61 73 74 65 72 43 61 72 64 90 00 ... Furthermore, the MASTERCARD PAYPASS card specification indicates that the response to the GPO (Get Processing Options) APDU 785 always be the same: a. APDU to the card for GPO APDU: 80 A8 00 00 02 83 00 00 b. APDU response from the card for GPO APDU is ALWAYS: 77 0A 82 02 00 00 94 04 08 01 01 00 90 00 ... The MASTERCARD PAYPASS card specification indicates that the response to the Read Record APDU 786 is to always be the same for a particular card that has been personalized, but will be different from personalized card to a different personalized card: a. APDU to the card for Read Record APDU: 00 B2 01 0C 00 b. APDU response from the card for Read Record APDU is ALWAYS for a specific card: 70 7F 9F 6C 02 00 01 56 3E 42 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50 4C 49 45 44 2F 4E 4F 54 5E 30 39 30 36 31 30 31 33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 32 30 30 30 31 31 31 31 30 9F 64 01 03 9F 62 06 00 00 00 38 00 00 9F 63 06 00 00 00 00 E0 E0 9F 65 02 00 0E 9F 66 02 0E 70 9F 6B 13 54 13 12 34 56 78 48 00 D0 90 61 01 90 00 99 00 00 00 0F 9F 67 01 03 90 00”	§A.4 SELECT AID Command >CARD: 00 A4 04 00 07 A0 00 00 00 04 10 10 00 CARD>: 6F 17 84 07 A0 00 00 00 04 10 10 A5 0C 50 0A 4D 61 73 74 65 72 43 61 72 64 90 00 GET PROCESSING OPTIONS Command >CARD: 80 A8 00 00 02 83 00 00 CARD>: 77 0A 82 02 00 00 94 04 08 01 01 00 90 00 READ RECORD Command >CARD: 00 B2 01 0C 00 CARD>: 70 7F 9F 6C 02 00 01 56 3E 42 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50 4C 49 45 44 2F 4E 4F 54 5E 30 39 30 36 31 30 31 33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 32 30 30 30 31 31 31 31 30 9F 64 01 03 9F 62 06 00 00 00 38 00 00 9F 63 06 00 00 00 00 E0 E0 9F 65 02 00 0E 9F 66 02 0E 70 9F 6B 13 54 13 12 34 56 78 48 00 D0 90 61 01 90 00 99 00 00 00 0F 9F 67 01 03 90 00

'171 Patent (Ex[1001])	PayPass-MagStripe (Ex[1058])																		
29:12-25: “For a MASTERCARD PAYPASS MSD v. 3.3 credential, table 3 shows a technical breakdown of how the data in the read compute cryptographic checksum response is formulated from data in the application system in proper APDU response BER-TLV format.” TABLE 3 <table><tr><td>77 0F</td><td></td><td></td></tr><tr><td>T</td><td>L</td><td>V</td></tr><tr><td>9F 61</td><td>02</td><td>B8 92 T2_CVC3</td></tr><tr><td>9F 60</td><td>02</td><td>FB C7 T1_CVC3</td></tr><tr><td>9F 36</td><td>02</td><td>00 5E ATC</td></tr><tr><td>90 00</td><td></td><td></td></tr></table>	77 0F			T	L	V	9F 61	02	B8 92 T2_CVC3	9F 60	02	FB C7 T1_CVC3	9F 36	02	00 5E ATC	90 00			§II.2.2.3 2.2.3 Data Field Returned in the Response Message The data field of the response message is a constructed data object with tag ‘77’ (<i>Response Message Template</i>). The value field may include several BER-TLV coded data objects, but must always include the <i>CVC3TRACK2</i> (tag ‘9F61’) and the <i>ATC</i> (tag ‘9F36’). The value field may also include the <i>CVC3TRACK1</i> (tag ‘9F60’). Additional data objects returned in the data field that are not described in this specification must be ignored by the <i>PayPass</i> reader. §A.4 • COMPUTE CRYPTOGRAPHIC CHECKSUM Command >CARD: 80 2A 8E 80 04 00 00 08 99 00 CARD>: 77 0F 9F 61 02 B8 92 9F 60 02 FB C7 9F 36 02 00 5E 90 00
77 0F																			
T	L	V																	
9F 61	02	B8 92 T2_CVC3																	
9F 60	02	FB C7 T1_CVC3																	
9F 36	02	00 5E ATC																	
90 00																			
29:30-39 with each interrogation by a mobile device. The CVC3 is cryptogram that is expected to be calculated with this algorithm from the MASTERCARD PAYPASS MSD v3.3 card specification: First KDcvc3 is created: $Z_L := \text{DES3}(\text{IMK}_{\text{cvc3}})[Y]$$Z_R := \text{DES3}(\text{IMK}_{\text{cvc3}})[Y \oplus ('FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF')]$ and define: $\text{KD}_{\text{cvc3}} := (Z_L Z_R).$	§III.3.8.1 3.8.1 KDcvc3 This section specifies the key derivation method used to generate the <i>ICC Derived Key for CVC3 Generation (KD_{CVC3})</i>. KD_{CVC3} is a 16-byte DES3 key derived from the <i>Issuer Master Key for CVC3 Generation (IMK_{CVC3})</i> as follows: 1. Concatenate from left to right the PAN (without any ‘F’ padding) with the PAN sequence number⁷ (if the PAN sequence number is not available, then it is replaced by a ‘00’ byte). If the result X is less than 16 digits long, pad it to the left with hexadecimal zeros in order to obtain an eight-byte number Y in numeric (n) format. If X is at least 16 digits long, then Y consists of the 16 rightmost digits of X in numeric (n) format. 2. Compute the two eight-byte numbers: $Z_L := \text{DES3}(\text{IMK}_{\text{CVC3}})[Y]$$Z_R := \text{DES3}(\text{IMK}_{\text{CVC3}})[Y \oplus ('FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF' 'FF')]$ and define : $\text{KD}_{\text{CVC3}} := (Z_L Z_R).$																		

'171 Patent (Ex[1001])	PayPass-MagStripe (Ex[1058])								
29:50-58 Next T2_IVCVC3 is created: $O := \text{PayPass-MagStripe}(3.3) \text{ Annex B alg}(KD_{cvc3})[\text{Track 2 Data}]$ The least two significant bytes of the output data are defined to be T2_IVCVC3. Finally, the CVC3 cryptogram for track 2 is calculated: $O := \text{DES3}(KD_{cvc3})[T2_JVcvc3 \parallel UN \parallel ATC]$	§III.3.8.2 3.8.2 Dynamic CVC3 Generation The $CVC3_{TRACK1}$ is generated using DES3 encipherment as follows: 1. Concatenate the data listed in Table 44 in the order specified to obtain an 8 byte data block (D): Table 44—Track 1 CVC3 Data Objects <table> <tr> <th>Data Object</th><th>Length</th></tr> <tr> <td>$IVCVC3_{TRACK1}$</td><td>2 bytes</td></tr> <tr> <td>UN</td><td>4 bytes</td></tr> <tr> <td>ATC</td><td>2 bytes</td></tr> </table> 2. Calculate O as follows: $O := \text{DES3}(KD_{CVC3})[D]$ 3. The two least significant bytes of O are the $CVC3_{TRACK1}$. The $CVC3_{TRACK2}$ is generated in the same way by replacing $IVCVC3_{TRACK1}$ with $IVCVC3_{TRACK2}$.	Data Object	Length	$IVCVC3_{TRACK1}$	2 bytes	UN	4 bytes	ATC	2 bytes
Data Object	Length								
$IVCVC3_{TRACK1}$	2 bytes								
UN	4 bytes								
ATC	2 bytes								