# Be Prepared: The EMV Pre-play Attack

Mike Bond*, Omar Choudary*, Steven J. Murdoch†, Sergei Skorobogatov*, Ross Anderson*
*Computer Laboratory, University of Cambridge, UK
forename.lastname@cl.cam.ac.uk
†Computer Science Department, University College London, UK
s.murdoch@ucl.ac.uk

***Abstract*—EMV, also known as "Chip and PIN", is the leading system for smartcard-based payments worldwide; it is widely deployed in Europe and is starting to be introduced in the USA too. It replaces the familiar mag-strip cards with chip cards. A cryptographic protocol is executed between a chip card and bank servers based on a message authentication code (MAC) over transaction data, including a nonce called the unpredictable number.**

**We discovered two protocol flaws: first, the lack of a terminal ID to identify involved parties, and second that the nonce is not generated by the relying party. Together, these make EMV vulnerable to the pre-play attack: pre-recorded transaction data from a target card can be replayed at a future location. This powerful attack can be exploited due to weak random number generators, by a man-in-the-middle between the terminal and the acquirer, or by malware in an ATM or POS terminal.**

**Our investigation started when we discovered that EMV implementers often used counters, timestamps or home-grown algorithms to supply the nonce. We describe the survey methodology we developed to chart the scope of this weakness, evidence from ATM and terminal experiments in the field, and our proof-of-concept attack implementation. Finally, we explore why these flaws evaded detection until now.**

## I. The Smoking Gun

EMV is the leading scheme worldwide for card payments and cash withdrawals at ATMs. More than 1.62 billion cards are in use worldwide, and EMV is now being adopted in the USA. EMV cards contain a smartcard chip, and are more difficult to clone than the traditional magnetic-strip cards. Yet in the decade since its introduction, a whole series of significant vulnerabilities have emerged.

The case that kicked off the research we report here was when Mr Gambin, a Maltese customer of HSBC, complained about a series of ATM transactions that were wrongly billed to his card in Palma, Majorca on the 29th June 2011, after he bought a meal at a restaurant there. He was refused a refund and asked us for advice. We observed that one of the fields in the log file, the "unpredictable number", looked rather predictable, as shown in Figure 1. It appears to consist of a 17 bit fixed value followed by a 15-bit counter that cycles every three monutes.

If the "unpredictable number" generated by an ATM is in fact predictable, then a criminal with temporary access to a card (say, in a Mafia-owned shop) can precompute the authentication codes needed to draw cash from that ATM at some time in the future. We call this the "pre-play" attack. We discovered that many ATMs generate poor random numbers; what's worse, a flaw in the protocol can let an attacker substitute a precomputed transaction even where the random number generator is sound.

| Date | Time | UN |
|---|---|---|
| 2011-06-29 | 10:37:24 | F1246E04 |
| 2011-06-29 | 10:37:59 | F1241354 |
| 2011-06-29 | 10:38:34 | F1244328 |
| 2011-06-29 | 10:39:08 | F1247348 |

Figure 1. Consecutive unpredictable numbers from an ATM

We informed the industry in early 2012 so that ATM software could be patched. We are now publishing the details to provide customers the evidence to pursue wrongly-denied claims, and to enable the crypto, security and bank regulation communities to learn the lessons.

## II. Background

In EMV, each bank card contains a smartcard chip, which authenticates transaction data using a message authentication code (MAC) calculated with a symmetric key shared between the card and the card-issuing bank. The chip protects against card counterfeiting. ButEMV did not cut fraud as much as hoped, as can be seen in Figure 2. Criminals adapted in several ways. First, they moved from card cloning to "card-not-present" transactions – Internet, mail-order, and phone-based payments.

Second, they started making magnetic-strip clones of EMV cards. Instead of entering PINs only at ATMs, customers were now entering their PIN in POS terminals, which are much easier to tamper with [1]. Thieves steal card data and PINs, then use mag-strip clones in countries like the USA where ATMs still accept mag-strip cards.

Third, a number of technical vulnerabilities emerged. For example, a stolen EMV card can be used in a POS device without knowing the PIN; a crook can use a man-in-the-middle device to trick the terminal into believing that the right PIN was entered, while the card thinks it is authorising a chip-and-signature transaction [2]. Criminals

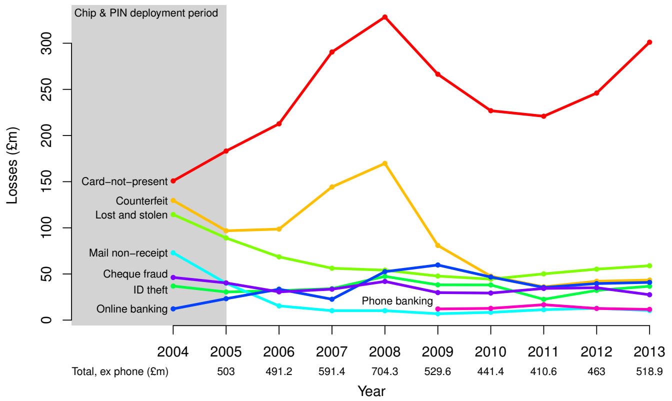


Figure 2. Fraud on UK payments cards (Fincial Fraud Auction UK 2014)

have now gone on trial in France for exploiting this "no-PIN" vulnerability [3].

But a lot of fraud is still unexplained, and is often blamed on the cardholder. So there is a public interest in discovering new vulnerabilities, and the preplay attack discovered here explains several fraud cases reported to us by cardholders in Spain, Poland and the Baltic states.

## III. The pre-play attack

An EMV transaction consists of three phases, as illustrated in Figure 3:

1) **card authentication** in which card details are read and authenticated by the ATM or POS terminal;
2) **cardholder verification** in which the person who presents the card is verified whether by PIN or signature; and
3) **transaction authorization** in which the issuing bank decides whether the transaction should proceed.

The principals are the card, the ATM/POS device and the issuer. It is the third phase that is of interest for the pre-play attack. In transaction authentication, the ATM or PIN entry device (PED) sends the card the amount, the currency, the date, the terminal verification results (TVR – the results of various checks performed by the ATM), and a nonce (in EMV terminology, the "unpredictable number" or UN). The card responds with a MAC known as the authorization request cryptogram (ARQC), calculated over these records, over the application transaction counter (ATC – a 16 bit number stored by the card and incremented on each transaction) and the issuer application data (IAD – a proprietary data field to carry information from the card to its issuer).

The ARQC is sent to the issuer, which verifies it, checks whether funds are available, that the card has not been reported stolen, and that the transaction does not look suspicious. It then returns to the ATM an authorization response code (ARC) and an authorization response cryptogram (ARPC). The card verifies the ARPC (which is typically a MAC over the ARQC exclusive-or'ed with the ARC), and returns an authenticated settlement record known as a transaction certificate (TC), which may be sent to the issuer immediately, or some time later as part of a settlement process. All these MACs are computed using a key shared between the card and the issuing bank, so the ATM or POS terminal cannot verify them.

In a normal EMV transaction the card sends an ARQC to prove that it is alive, present, and engaged in the transaction. The ATM or POS device relies on the issuer to verify this and authorise the transaction. The unpredictable number ensures that transactions are unique, and tied to a specific terminal. But If an attacker can predict it, then he can mount a "pre-play" attack which is indistinguishable from card cloning: authentication data are collected at one terminal at moment in time, and played to one or more possible verifying parties later. For example, a tampered terminal in a store can collect card details and ARQCs from a victim for use later, at an ATM or POS whose UN can be predicted or manipulated. We now describe the two protocol flaws in detail.

### A. *EMV protocol flaws*

The first flaw is the specification, which does not require the identity of the terminal – a classic mistake, reminiscent of [4]. While the EMV framework can support a terminal ID through a list of fields to be MACed in the ARQC (the CDOL1), the standard format developed by Visa (the version 10 cryptogram format) requires only the terminal country code.

The second flaw is in the protocol architecture: while the terminal generates the random number, it is the issuing bank that relies on it. Therefore, the issuer depends on the merchant for transaction freshness, but the merchant may not have the incentive to provide it, may not be able to deliver it correctly due to lack of end-to-end authentication. In fact, the terminal might even be collusive.

Recently there has been some formal analysis of EMV, but this flaw was not discovered [5]. The model made two errors. First, the UN was modelled as a fresh nonce, even though this is not required by EMV. Second, the issuer and terminal are modelled as the same principal, whereas they are not; the terminal communicates with an acquirer (the merchant's bank) that in turn sends the transactions to a switch that finally relays the transactions to the issuer.

### B. *Pre-play attacks based on a weak RNG*

The EMV protocol designers did not think carefully about what is required for the UN to be "unpredictable". The specifications and conformance testing procedures simply required that four consecutive transactions performed by the terminal should have unique unpredictable numbers [6, test

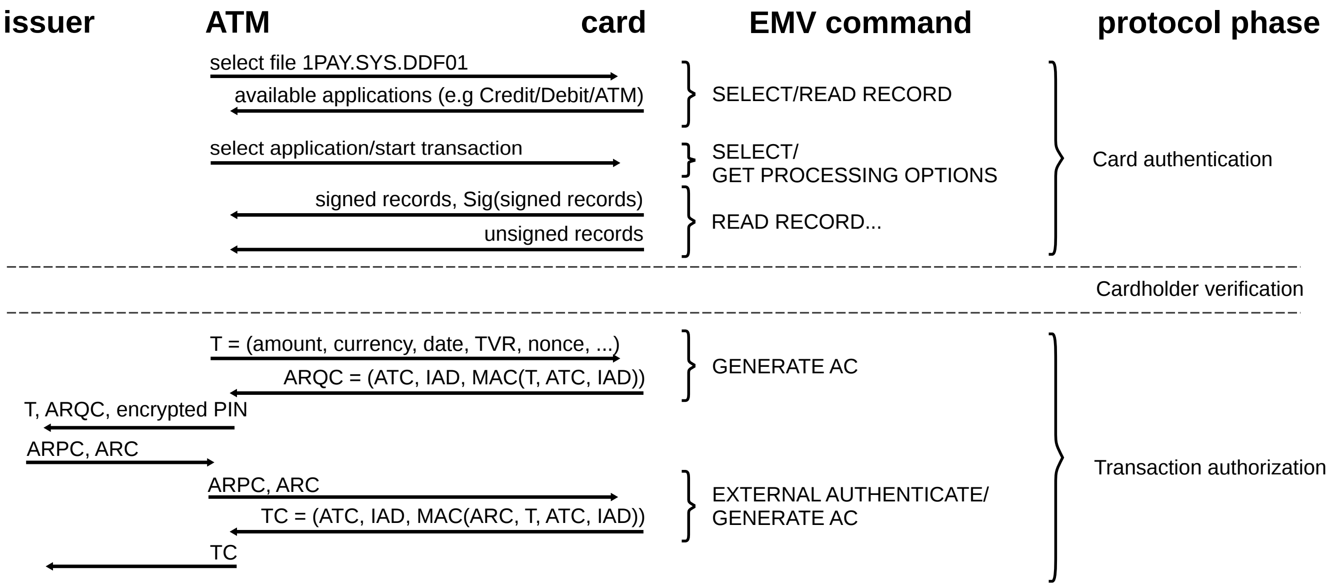

issuer
ATM
card
EMV command
protocol phase
select file 1PAY.SYS.DDF01
available applications (e.g Credit/Debit/ATM)
SELECT/READ RECORD
select application/start transaction
SELECT/
GET PROCESSING OPTIONS
signed records, Sig(signed records)
unsigned records
READ RECORD...
Card authentication
Cardholder verification
T = (amount, currency, date, TVR, nonce, ...)
ARQC = (ATC, IAD, MAC(T, ATC, IAD))
GENERATE AC
T, ARQC, encrypted PIN
ARPC, ARC
ARPC, ARC
TC = (ATC, IAD, MAC(ARC, T, ATC, IAD))
EXTERNAL AUTHENTICATE/
GENERATE AC
TC
Transaction authorization

Figure 3. Outline of an EMV transaction at ATM. Note that while the messages between card and ATM have been verified, messages between issuer and ATM may vary depending on card scheme rules

2CM.085.00]. Thus a rational implementer in a hurry would simply use a counter.

Since we disclosed this flaw, the EMV 4.2 specification now offers guidance as to how to generate the unpredictable number but previous versions left the algorithm entirely up to implementers. Even the suggested construction (hash or exclusive-or of previous ARQCs, transaction counter and time) would not be adequate if the ATM is rebooted and both the time and transaction counter are predictable.

*1) UN data collection:* Markettos and Moore [7] first showed that a pre-play attack was possible against EMV if the attacker could sabotage the RNG. However, before our work, there was no empiral work on the quality of the RNGs used by actual ATMs or POS terminals. So we set out to collect UNs generated by ATMs and POS terminals in our area, which together with log files from legal cases give us an initial view of the EMV system in practice.

In order to obtain UN data and high-resolution timestamps from ATMs, we made a set of passive monitoring cards by adding our own ATM protocol analyser circuitry, consisting of an additional microcontroller and memory, to a standard debit card. This process required careful placement and connection of very small components (down to 0.4 mm pin pitch) and custom hardware to retrieve the transaction logs. The modified card is shown in Figure 4. It remains a valid payment card – the transaction flow proceeds as normal – so it should always be accepted. It can also be inserted into a variety of ATMs and POS devices without arousing suspicion[1].

For each ATM investigated, we harvested between five and fifty unpredictable numbers by performing repeated balance enquiries[2] and then finally a small cash withdrawal. We used balance enquiries to minimise the number of withdrawals and avoid triggering any fraud monitoring systems.

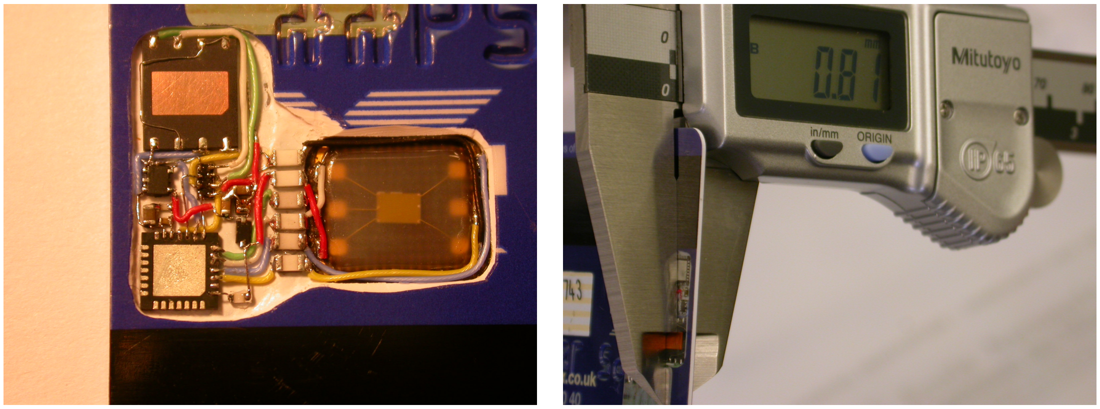
At POS terminals, sales assistants are often briefed to avoid handling or even looking at customer cards. So we could use existing monitoring tools such as the Smart Card Detective [8], which relies on a hidden wire running up the experimenter's sleeve.

During our UN collection campaign, we performed more than 1,000 transactions across 22 different ATMs and five POS terminals. Table I(a) shows a selection of data collected from various ATMs exhibiting some ineffective algorithms. ATM1 and ATM2 contain a typical pattern, which we denote *characteristic C*, where the high bit and the third nibble of each UN are always set to zero. This alone reduces the entropy of the unpredictable numbers from 32 to 27 bits. 11 of 22 ATMs we looked at exhibited this. These included ATMs of wildly different ages and running different operating systems, so we suspect it to be an artifact of a particular EMV kernel post-processing rather than of the RNG source itself.

In Table I(b) we show a list of stronger consecutive unpredictable numbers retrieved from a local POS terminal. Even in this case the first bit appears to remain 0, which

[1] For ethical and prudential reasons we informed the police that such experiments were underway; we also went through our local ethics process.

[2] It seems all transactions at ATM are authenticated by EMV protocol runs, but some with a zero withdrawal amount.



(a) Rear of card showing real EMV chip (right), monitoring microcontroller (bottom left), and flash storage (top left)

(b) Card is 0.8mm at thickest point so within tolerance for use within EMV terminals

Figure 4. Passive monitoring card used to collect UN data

Table I
CATEGORISED UNPREDICTABLE NUMBERS

(a) From Various ATMs

| Counters | | Weak RNGs | |
|---|---|---|---|
| ATM4 | eb661db4 | ATM1 | 690d4df2 |
| ATM4 | 2cb6339b | ATM1 | 69053549 |
| ATM4 | 36a2963b | ATM1 | 660341c7 |
| ATM4 | 3d19ca14 | ATM1 | 5e0fc8f2 |
| ATM5 | F1246E04 | ATM2 | 6f0c2d04 |
| ATM5 | F1241354 | ATM2 | 580fc7d6 |
| ATM5 | F1244328 | ATM2 | 4906e840 |
| ATM5 | F1247348 | ATM2 | 46099187 |
| | | ATM3 | 650155D7 |
| | | ATM3 | 7C0AF071 |
| | | ATM3 | 7B021D0E |
| | | ATM3 | 1107CF7D |

(b) From local POS terminal

| Stronger RNGs | |
|---|---|
| POS1 | 013A8CE2 |
| POS1 | 01FB2C16 |
| POS1 | 2A26982F |
| POS1 | 39EB1E19 |
| POS1 | 293FBA89 |
| POS1 | 49868033 |

might suggest the use of a signed integer.

Based on our analysis of RNGs from logs, ATMs and POS terminals, we can distinguish three broad classes of ineffective RNGs: (*a*) an obviously weak RNG algorithm (e.g. counters or clocks directly used as the UN, homegrown algorithms, casting down to the wrong integer size); (*b*) a simple RNG with little or no seeding (e.g. linear congruential generator, combinations of fixed bits and bits that cycle, using standard C library calls such as `time()` and `rand()`); (*c*) an RNG that can be put into a predictable state (e.g. a strong RNG fed by a weak source of randomness that's restarted on power-up, or an RNG that relies only on data from previous transactions).

*2) Harvesting the data:* Given temporary access to an EMV card, whose holder enters the PIN, and a range of possible unpredictable numbers to be harvested, the crook programs his evil terminal to read the static data from the card and call `GENERATE AC` to obtain an ARQC and TC for each possible UN. For each card several dozen ARQCs can be harvested. The only limitation is the time that the card can be left in a sabotaged POS while the customer believes that the machine is waiting for authorisation.

*3) Cashing out:* In the case of the ATM in Majorca that started this line of research, the counter rolls over every three minutes, so an attacker might ask a card in his store for twenty ARQCs around a point in the 15-bit counter's cycle. On visiting the ATM his attack card would first calibrate to the ATM's counter, and then initiate transactions when the counter is expected to be in the range for which he has captured ARQCs. We show an illustration of the pre-play attack based on a weak RNG in Figure 5 (left).

*4) Implementation and evaluation:* We used test cards with known ARQC-generation keys (UDK) to prove the attack's viability using an indistinguishability experiment. First, we took two test cards A and B loaded with the same ARQC-generation keys, initialised with the same ATC and handled identically. Then, we harvested data from card A and programmed it on to a "pre-play card", implemented using the ZeitControl BasicCard platform. Finally, we compared traces between the pre-play card version of card A and the real card B, and observed that they are identical. This means that, at a protocol level, it is impossible for an ATM to distinguish between the real and pre-play cards.

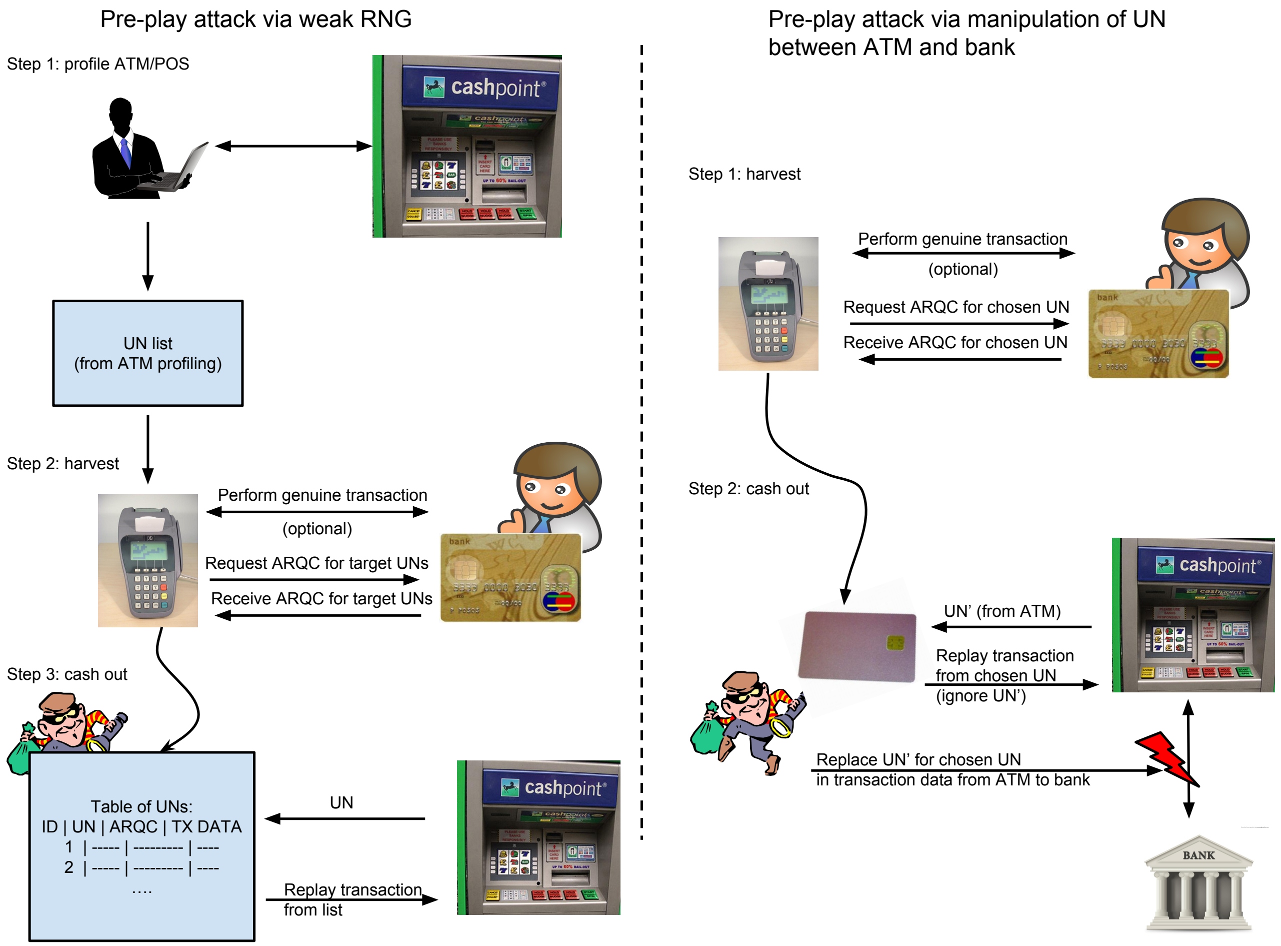


Figure 5. Overview of the pre-play attack using a weak RNG (left) or tampering with the UN at the ATM/POS side (right)

### C. Attacks based on UN modification

In real life, we cannot rely on communications between the merchant and the card issuing bank to be protected by encryption or even authentication. This is a well-known problem from ATM networking (see [9, p336]). In such cases, a man-in-the-middle device between the terminal and the bank can be used to attack even systems whose random number generation is sound. In this case, it is no longer necessary to profile an ATM or POS terminal. The attacker can simply choose an arbitrary UN and obtain the related transaction data, including the ARQC, from the victim's card. He then replays the transaction data at a terminal and replaces the terminal's real UN with his chosen one, as shown in Figure 5 (right). This could be an attractive way to attack merchants with high-value transactions, such as jewelers or investment firms. Even if they guard their premises and their POS equipment, an attacker can go after their network link at a utility cabinet.

### D. Other attack variants

Even if the UN generation algorithms are patched, or the communication link between the merchant and the issuer bank is secured, there are several other protocol attack variants.

*1) Malware infection:* There have been numerous cases of malware-infected ATMs operating in Eastern Europe, and of POS devices being infected in the USA. Depending on the internal architecture, it may be easy for such malware to sabotage the choice of UN. In fact, one bank suggested to us that the ATM that kicked off this whole research project may have been infected with malware [10]. Alternatively, the malware might collude to fix up the UN to match the presented ARQC.

*2) Supply chain attacks:* Such attacks have already been seen against POS terminals in the wild, and used to harvest magnetic strip data. So it is feasible that a criminal (or even a state-level adversary) might sabotage the RNG deliberately, either to act predictably all the time, or to enter a predictable mode when triggered via a covert channel. A suitably sabotaged RNG would probably only be detected via reverse engineering or observation of real-world attacks.

*3) Collusive merchant:* We have recently seen a transaction dispute in which a customer claims to have made one small purchase at a merchant yet his bank claims he made ten large ones too. These were filed via three different acquirers, and report different terminal characteristics despite coming from the same terminal – so the evidence of fraud is clear. yet the bank maintains the transactions are the customer's fault, and the case is proceeding. Banks' unwillingness to charge back such transactions to merchants is an open incitement to merchant fraud.

A merchant might maliciously modify their EMV stack to be vulnerable, or inject replayed card data into the authorisation/settlement system. He could take a cut from crooks who come to use cloned cards at their store, or just pre-play transactions directly. We also have evidence of merchants tampering with transaction data to represent transactions as PIN-verified when they were not, so as to shift liability and cut transaction fees. In the UK, there was a string of card cloning attacks on petrol stations where a gang bribed store managers to look the other way when PIN pads were tampered with and monitoring devices inserted into network connections – exactly what's needed for a pre-play attack.

## IV. Limitations and Defences

The limits on the effectiveness of the pre-play attack relate to the data fields included in the MAC, and the quality of cryptographic checks done by the issuer. When the issuer follows card scheme standards, the transaction *country*, *date* and *amount* must be chosen in advance. The PIN of the card must harvested at the same time as the pre-play data (or already known). Also, subsequent use of the real card might advance the application transaction counter or ATC (a counter kept by the card) and invalidate recorded data.

### A. Defences against random-number attacks

The simplest fix for random-number attacks is a cryptographically secure random number generator (RNG), but this is not necessarily practical. RNG design is a matter for acquiring banks, ATM vendors, merchants and POS terminal suppliers, while the cost of fraud falls on card issuing banks and customers. Issuers might unilaterally try to detect evidence of harvesting, such as large gaps in ATC sequences. They should reject online transactions with out-of-order ATCs, but this is easier said that done, as transaction re-ordering can occur in offline payments. We've seen banks processing duplicated transactions without checking the ATC.

### B. Defences against protocol attacks

In the short-to-medium term, issuers would do better to meticulously verify the transaction certificate (TC). This is sent by the card to the terminal as the transaction completes, and should be submitted to the issuer when the transaction is presented for settlement. It states whether the card verified the ARPC, which in turn was computed by the card-issuing bank after it verified the ARQC. A pre-play attack can still yield a TC, but its IAD will show that issuer authentication did not complete successfully.

But TC checking is rare. Visa's 'Transaction Acceptance Device Guide' section 5.12 states:

> "Devices operating in a single-message or host-capture environment should ensure a TC is generated for approved transactions. Although not needed for clearing, generating a TC ensures that cards do not request unnecessary online approvals on subsequent transactions and also provides liability protection for acquirers."

Mitigating acquirer liability in the event of stand-in processing is all very well, but our concern here is the liability faced by the cardholder.

In the event of a court having to decide whether a series of disputed transactions from a single terminal was made with the cardholder's collusion or via a pre-play attack, the first forensic test should be to examine the TC. If a valid TC is generated by a card following a correct ARPC that in turn followed a correct ARQC, then the card was present and active when the ARPC was generated. This does not totally exclude fraud, as there may have been a relay attack [11]; but pre-play attacks at least appear unlikely.

## V. Discussion

The potential vulnerability of EMV to a poor random number generator was discussed in the abstract by Murdoch [12]. Markettos and Moore [7] explored how otherwise secure true random number generators could be manipulated to produce more deterministic output, and how to exploit a weak RNG in an EMV transaction. But this paper is the first work to show that poor random number generators exist in the wild, that they have been implicated in fraud, how they can be exploited, and the protocol flaws in the EMV specification that make this so hard to counter.

It's interesting to compare the pre-play attack to full cloning (where the ARQC generation keys are extracted). One might imagine full cloning is much more powerful, but since each card has its own ATC, these will diverge in due course and become detectable.

In fact, a pre-play attack could be more powerful than a full cloning attack, for reasons of scale. If keys could be extracted from cards at no cost, say using a power analysis attack conducted by a rogue terminal, then a cloning attack might be done on an industrial scale, but this is unlikely, as the industry has spent 15 years and millions of dollars on countermeasures. It is more likely that a cloning attack would involve card capture followed by destrucive reverse engineering, and perhaps a semi-invasive attack costing tens to hundreds of dollars per card.

By contrast, a preplay attack could scale massively. If a gang succeeds in compromising a number of terminals (which was done in the UK physically by three separate gangs in the mid-2000 [13]) or in compromising the communications to a number of high-value stores (which was done to jewelry stores in Hatton Garden in the 1980s) the cards can have ARQCs harvested in one location and presented in another. The same holds if a number of terminals are compromised by malware.

## VI. Conclusions

EMV has been around for more than ten years, yet criminals keep finding serious new attacks. It is shocking that many ATMs and point-of-sale terminals have seriously defective random number generators which leave the system open to fraud. It is even worse that an associated protocol failures leave it open to fraud at scale using malware in point-of-sale terminals – a growing real-world problem.

This flaw challenges current thinking about authentication. Existing models of verification do not easily apply to a complex multi-stakeholder environment; indeed, EMV was verified to be secure. We explained why that verification didn't work. In addition, mechanisms for rolling out fixes across networks with huge installed bases of cards and terminals, and strong externalities, are nowhere near serviceable.

We have exposed a structural governance failure that gives rise to systemic risk. In a multi-party world where not even the largest card-issuing bank or acquirer or scheme operator has the power to fix a problem unilaterally, we cannot continue to rely on a slow and complex negotiation process between merchants, banks and vendors. Regulators have been credulous in accepting industry assurances about operational risk management, and it is time for them to take an interest. It is welcome that the US Federal Reserve is now paying attention, and time for European and other regulators to follow suit.

### *Acknowledgments*

Steven J. Murdoch is funded by the Royal Society. Omar Choudary is a recipient of the Google Europe Fellowship in mobile security and this research is supported in part by that fellowship. The opinions expressed in this paper do not represent the views of Google.

## References

[1] S. Drimer, S. J. Murdoch, and R. Anderson, "Thinking inside the box: system-level failures of tamper proofing," in *IEEE Symposium on Security and Privacy (Oakland)*, May 2008, pp. 281–295.

[2] S. J. Murdoch, S. Drimer, R. Anderson, and M. Bond, "Chip and PIN is broken," in *IEEE Symposium on Security and Privacy (Oakland)*, May 2010.

[3] S. Sellami, "L'imparable escroquerie à la carte bancaire," Le Parisien, 24 January 2012, http://www.leparisien.fr/faits-divers/l-imparable-escroquerie-a-la-carte-bancaire-24-01-2012-1826971.php.

[4] R. Anderson and R. Needham, "Programming Satan's computer," in *Springer Lecture Notes in Computer Science vol 1000*, 1995, pp. 426–441.

[5] J. de Ruiter and E. Poll, "Formal analysis of the EMV protocol suite," in *Theory of Security and Applications (TOSCA 2011)*, ser. LNCS, S. Moedersheim and C. Palamidessi, Eds., vol. 6693. Springer, March 2011, pp. 113–129.

[6] EMVCo, LLC, "Terminal level 2, test cases," Type Approval, November 2011, version 4.3a.

[7] A. T. Markettos and S. W. Moore, "Frequency injection attack on ring-oscillator-based true random number generators," in *Workshop on Cryptographic Hardware and Embedded Systems*, 2009, pp. 317–331.

[8] O. Choudary, "The smart card detective: a hand-held EMV interceptor," University of Cambridge, Computer Laboratory, Tech. Rep. UCAM-CL-TR-827, December 2012.

[9] R. Anderson, *Security Engineering – A Guide to Building Dependable Distributed Systems*. Wiley, 2003.

[10] *ATM Malware*, SC Magazine, October 2013, http://www.pcworld.com/article/2058360/atm-malware-may-spread-from-mexico-to-englishspeaking-world.html.

[11] S. Drimer and S. J. Murdoch, "Keep your enemies close: Distance bounding against smartcard relay attacks," in *USENIX Security Symposium*, August 2007.

[12] S. J. Murdoch, "Reliability of chip & PIN evidence in banking disputes," in *Digital Evidence and Electronic Signature Law Review*, vol. 6. Pario Communications, November 2009, pp. 98–115, ISBN 0-9543245-9-5.

[13] "Petrol firm suspends chip-and-pin," BBC News, 6 May 2006.