



OWASP

Open Web Application
Security Project

Hack in, Cash out

Hacking and Securing Payment Technologies

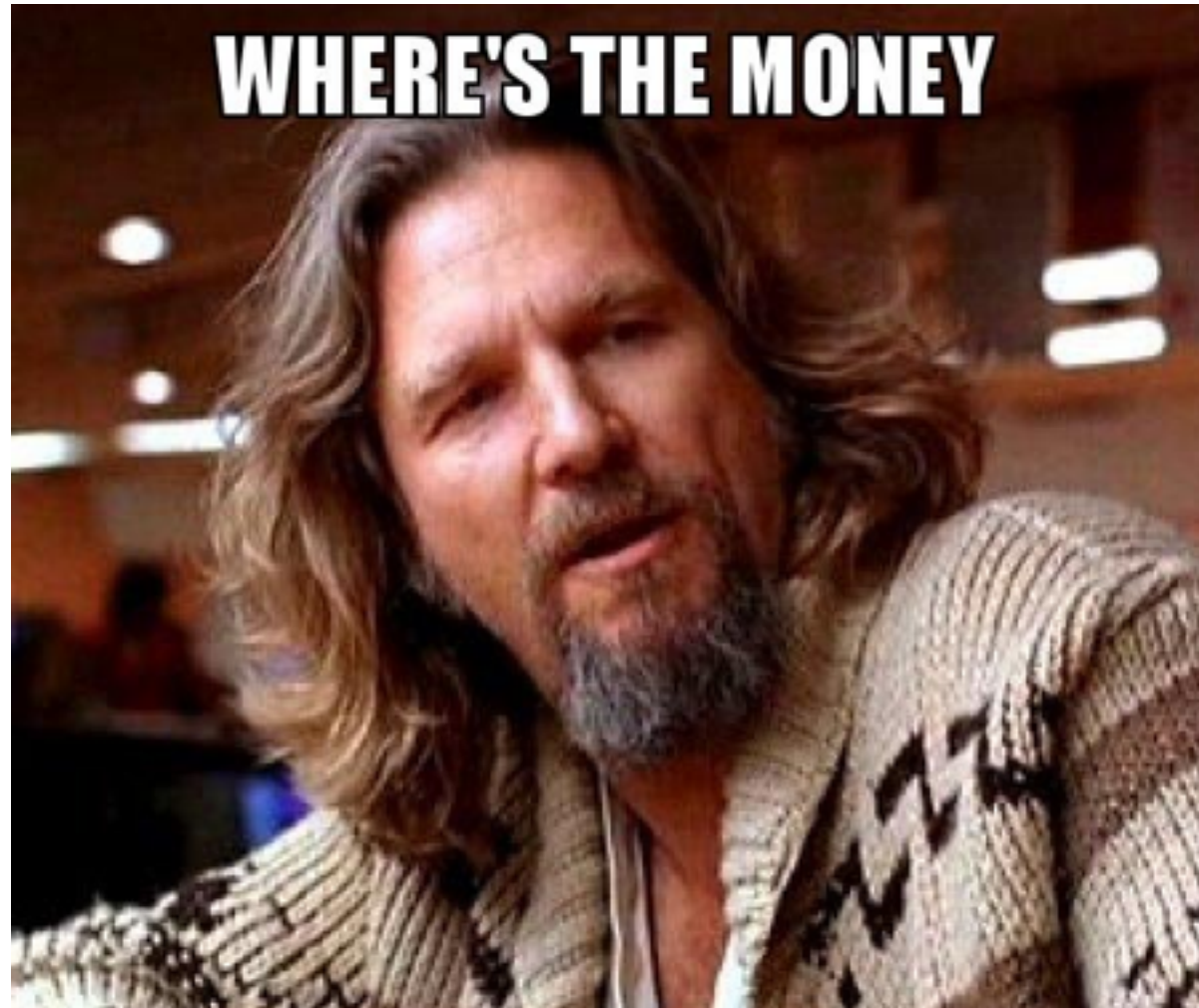
Tim Yunusov

POSITIVE TECHNOLOGIES

Transaction stream fraud



Main question of the payment pentest



Good pentest



Bad pentest



Get money from the bank



From our own accounts



Decisions, decisions...



4 accounts in 2018
4 accounts in 2019



Card payment processing



Card



Endpoint



Acquirer



Card brands



Issuer's
Authorisation host



Endpoints

	ATM	POS	Online
Money	+	+	+
Card's data	+	+	+
Card's testing	Limited	+	+
Card's attacks	Limited	+	Limited



Most ATMs can be hacked in under 20 minutes

Experts tested ATMs from NCR, Diebold Nixdorf, and GRGBanking.



By [Catalin Cimpanu](#) for [Zero Day](#) | November 16, 2018 -- 05:30 GMT (05:30 GMT) | Topic: [Security](#)

blackhat
USA 2018
AUGUST 4-9, 2018
MANDALAY BAY / LAS VEGAS

Blackbox is dead –
Long live Blackbox!

Vladimir Kononovich
Aleksy Stennikov

pssecurity.com #BHUSA #BLACKHATEVENTS POSITIVE TECHNOLOGIES



@A1ex S

@groke1105

@ivachyou

@L_AGalloway





Terminal Simulator

Testing Tomorrow's Transactions Today

<https://www.terminalsimulator.com/>

require a PC/SC card reader. If you don't have such a reader, you can order a simulator kit, that will include a dual interface (contact and contactless) reader, and the associated software on CD. Please go ahead and register as a user of this site, and then you will be able to download or order the products that are listed below.

Welcome

This site has been created to allow you to obtain the latest versions of our Terminal Simulator software. This software requires a PC/SC card reader. If you don't have such a reader, you can order a simulator kit, that will include a dual interface (contact and contactless) reader, and the associated software on CD. Please go ahead and register as a user of this site, and then you will be able to download or order the products that are listed below.

Featured Products



Activation Extension
2020

Extension of
activation to January
2020

View more details



```
Terminal Simulator
-----
Tag : 91 [Issuer Authentication Data]
Length: 04
Data : 7457472F641878770000
-----
ARPC: 7457472F64187877
ARPC Response Code: 0000
Online Response is DECLINED
-----
Transaction time = 5170ms (Terminal: 3484ms APDU: 1686ms)
Generating AC command
-----
COOL2 = 810A8A0295059F37049F4C08
COOL2 data = 7457472F64187877000001000000000003080333990e3e8f66626cfe2
+ finished generating AC command
-----
+ sending 2nd GENERATE AC command
-----
---> Command :
CLA INS P1 P2 LC LE
80 A0 00 00 1B 00
-----
74 57 47 2F 64 18 78 77 00 00 01 00 00 00 00 00 two/d.[w.....
00 30 80 33 39 90 03 88 F6 06 26 EF 02 .0.3S...F6..
-----
<--- Response: (SW1-SW2) 9000 (Length) 28
-----
77 29 9F 27 01 00 9F 36 02 05 ED 9F 26 08 F9 38 w).....0....0...8
BF 34 1E 57 60 56 9F 10 12 01 10 20 90 03 24 04 .4.W.V.....$.
00 90 83 00 00 00 03 40 00 00 FF .....0...
-----
Transaction time = 176 ms
-----
Tag : 77 [response message Template Format 2]
Length: 29
+ Tag : 8e37 [cryptogram information data]
Length: 01
Data : 80
+ Tag : 9F36 [Application Transaction Counter (ATC)]
Length: 02
Data : 05ED
+ Tag : 9F26 [Application cryptogram]
Length: 08
Data : #038F341E576056
+ Tag : 9F10 [Issuer Application Data]
Length: 12
Data : #11020900324040090A30000000340000FF
-----
+ GENERATE AC processed ok
CID = 80 - AAC
ATC = 05ED
Application counter = #038F341E576056
```

HELP OPTIONS



POS+RCE – is the instrument

- EMV/NFC core real implementation
 - May contain a lot of bugs
- Real payment process workflow
 - Payment packet
 - Configurations (limits, etc)
 - Offline authentication and risk management



Example of the payment packet

```
{*invoiceId*:"INV2-██████████",*dateTime*:"2019-03-18T11:05:58+0000",*card*:{*reader*:  
{*vendor*:"██████████",*readerSerialNumber*:"10552290",*dev*:"██████████"},*inputType*:"contactless_chip",*emvData*:"500a4d6173746572436172645f24032109305f280208265f2a0208265f340100  
820219808407a0000000041010950500000080009a031903189c01209f0206000000001009f0306000000000009f0607a00000000410109f090200029f101201108000032208000000000000000000ff9f  
120a4d6173746572436172649f1a0208269f1c0831323334353637389f1e0831303535323239309f2608cad7779c1aa2a3c99f2701009f33030008089f34031f03029f3501229f360204429f3704aaaaaaaaadf28  
0100df30020201dfae022097c2508543007dfe18bbdd076c76d61dd813094c12e6ac83855232ae43caa05edfae030affff9802840f88200168dfae5a08537590ffffff5611"}}}
```

BER encoding





- TLV – Tag Length Value



Example

- AA0105 [hex]
- Tag – AA
- Length – 1 byte
- Value - 05

```
00 69 00 43 00 EA 01  
00 01 0D 03 00 12 06  
00 A7  
00 20  
01  
03  
00 33  
00 53  
00 01 \\Number of TLVs - top level  
29 FF 00 4D \\Parent TLV  
Nested TLVs [ 2A 0E 00 04 D0 D1 2B 37  
Nested TLVs [ 2A 01 00 41 \\Length includes all nested TLVs  
Nested TLVs [ 2A 03 00 01 00  
Nested TLVs [ 2A 07 00 04 00 00 13 98  
Nested TLVs [ 2A 0A 00 02 00 14  
Nested TLVs [ 2A 02 00 13 \\Length includes next 3 TLVs  
Nested TLVs [ 2A 08 00 02 00 02  
Nested TLVs [ 2A 09 00 01 02  
Nested TLVs [ 2A 08 00 04 00 00 1F 40  
Nested TLVs [ 2A 02 00 13 \\Length includes next 3 TLVs  
Nested TLVs [ 2A 08 00 02 01 65  
Nested TLVs [ 2A 09 00 01 32  
Nested TLVs [ 2A 08 00 04 00 00 1F 40
```

Example of the payment packet

50 (application label) MasterCard
5F24 (card expiry) 210930
5F28 (issuer country code) GBR (United Kingdom)
5F2A (terminal currency code) GBP (Pound Sterling)
5F34 (PAN sequence number) 00
82 (AIP - Application Interchange Profile) 
 1000 (Byte 1 Bit 5) Cardholder verification is supported
 0800 (Byte 1 Bit 4) Terminal risk management is to be performed
 0100 (Byte 1 Bit 1) CDA supported
 0080 (Byte 2 Bit 8) EMV and Magstripe Modes Supported
84 (dedicated file name) A0000000041010
95 (TVR - Terminal Verification Results) 
 000008000 (Byte 4 Bit 8) Transaction exceeds floor limit
9A (transaction date) 190318
9C (transaction type) 20
9F02 (amount authorized) 000000000100
9F03 (amount other) 000000000000
9F06 (application id) A0000000041010
9F09 (terminal application version number) 0002
9F10 (issuer application data) 
 Key Derivation index 01
 Cryptogram version number 10
 Card verification results 
 Byte 1 Bit 8 = 1, Byte 1 Bit 7 = 0 Second Generate AC not requested
 Byte 1 Bit 6 = 0, Byte 1 Bit 5 = 0 AAC Returned in First Generate AC
 Byte 3 Bits 8-5 Right nibble of Script Counter = 0
 Byte 3 Bits 4-1 Right nibble of PIN Try Counter = 3
 Byte 4 Bit 6 = 1 Offline PIN Verification Not Performed
 Byte 4 Bit 2 = 1 Domestic Transaction
 Byte 5 Bit 4 = 1 Go Online On Next Transaction Was Set
DAC/ACC Dynamic Number 2 Bytes 0000
Plaintext/Encrypted Counters 00000000000000FF

9F12 (application preferred name) MasterCard
9F1A (terminal country code) GBR (United Kingdom)
9F1C (terminal id) 12345678
9F1E (terminal serial number) 10552290
9F26 (application cryptogram) CAD7779C1AA2A3C9
9F27 (cryptogram information data) AAC (Application Authentication Cryptogram - Declined)
9F33 (terminal capabilities) 
 000800 (Byte 2 Bit 4) No CVM Required
 000008 (Byte 3 Bit 4) CDA
9F34 (CVM Results - Cardholder Verification Results) 
 1F No CVM required
 03 If terminal supports CVM
 02 Successful
9F35 (terminal type) 22
9F36 (ATC - application transaction counter) 1090
9F37 (unpredictable number) A3D751EA
DF28 (?) 00
DF30 (?) 0201
DFAE02 (?) 97C2508543007DFE18BDD076C76D61DD813094C12E6AC83855232AE43CA.A05E
DFAE03 (?) FFFF9802840F88200168
DFAE5A (?) 537590FFFFFFF5611

<https://tvr-decoder.appspot.com>

PAN/Track2/Expiry date

Transaction date and time

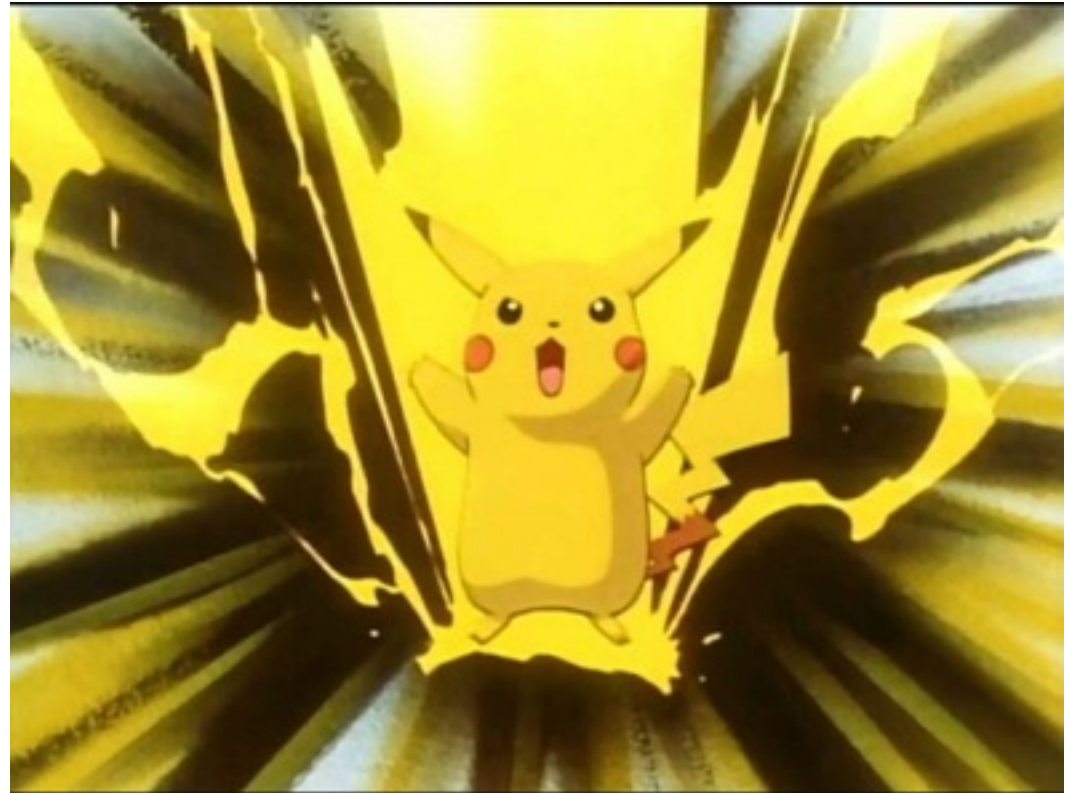
Amount and currency

Type of the operation (payment, cashback, refund, other)

Type of the cryptogram, cardholder verification method

Attacks

- Refund/reverse attacks
- Chip & PIN attacks
- Card testing



Reverse attacks

72,811 views | Nov 23, 2015, 06:40am

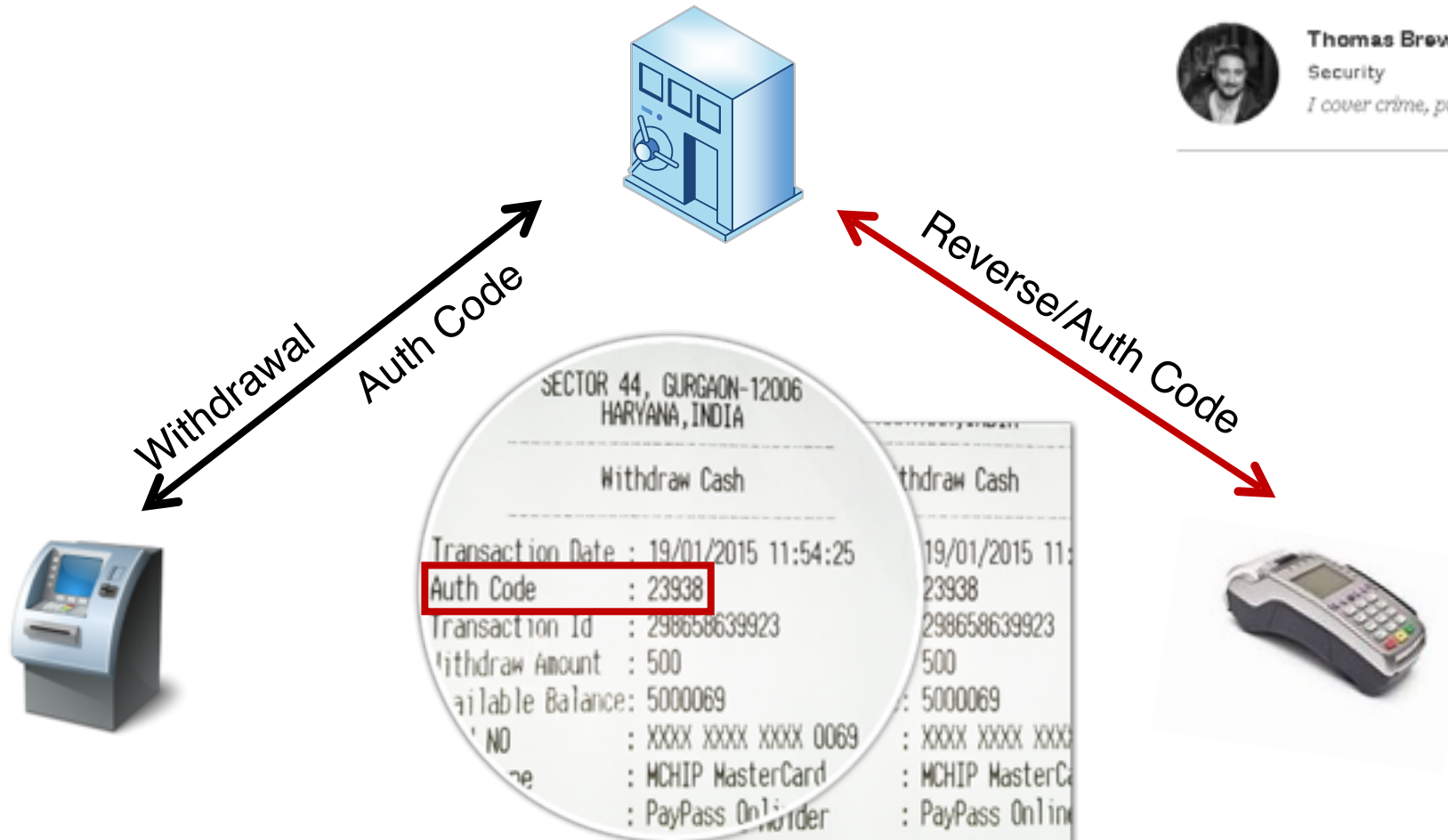
Criminals Steal \$4 Million In Cash With Novel 'Reverse ATM' Attack



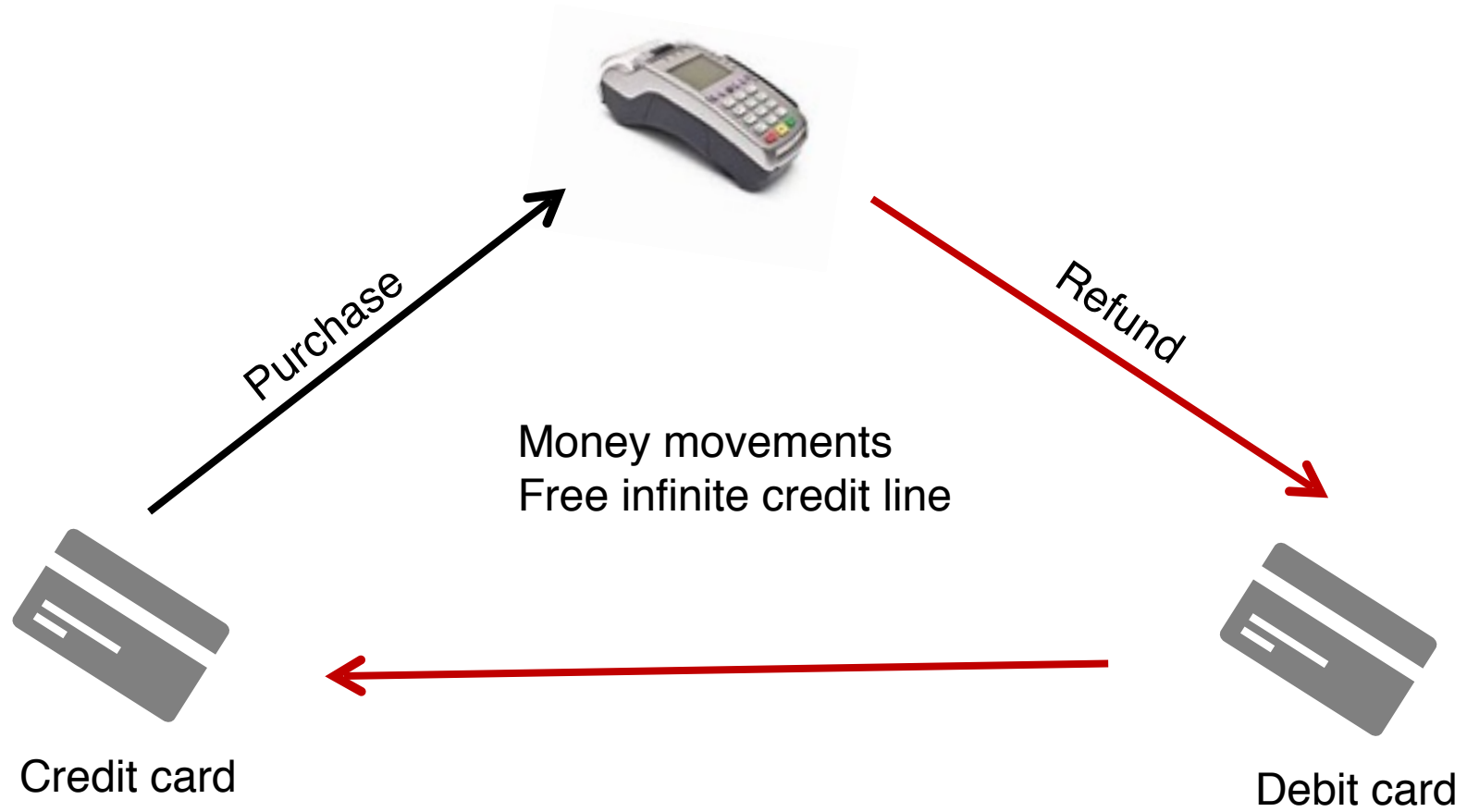
Thomas Brewster Forbes Staff

Security

I cover crime, privacy and security in digital and physical forms.



Refund attacks



Chip & PIN is still broken

- 2005 University of Cambridge, <https://murdoch.is/papers/cl05chipandspin.pdf>
- 2010 Inverse Path (F-Secure) / Aperture Labs

<https://cansecwest.com/csw11/Chip%20&%20Pin%20-%20Barisani%20&%20Bianco.pdf>

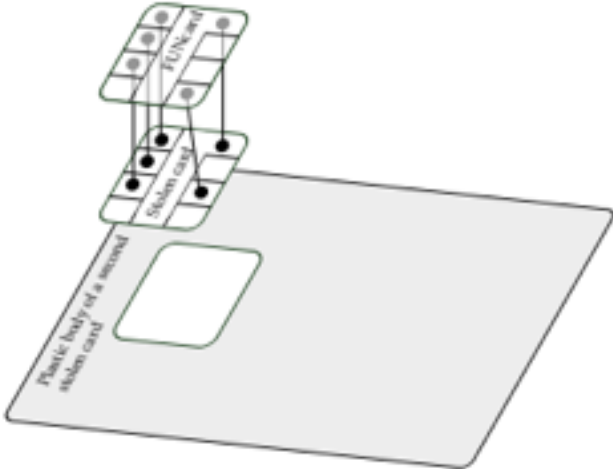
- Intercept PIN (ICC plaintext PIN verification)
- Make transactions without PIN knowledge (“PIN OK” attack)
- Downgrade to chip&signature

Chip & PIN is still broken

- CVM list – cardholder verification method list
 - CVM list is defined on the card
 - CVM List provides the terminal with four pieces of information on how an issuer wishes the cardholder to be verified:
 - CVM method (in priority)
 - Conditions of use
 - What if the CVM method is failed
 - Encrypted PIN if supports, then Unencrypted PIN if supports, the signature, than cancel
 - <https://www.spotterswiki.com/emv/cardsearch.php>
 - <https://tvr-decoder.appspot.com>
- Offline data authentication – when POS checks that card and it's data were genuine: SDA, DDA, **CDA**

When hackers come

- 2011, France <https://eprint.iacr.org/2015/963.pdf>
 - 40 cards
 - PIN-OK additional chip
 - 7000 transactions
 - 680,000 USD



Chip & PIN is still broken


- 2019, Europe
 - PIN interception, “PIN OK” attack, chip&signature downgrading
- Why?
 - “Nowadays CVM is signed” (c) Inverse Path - **CDA**
 - Weak CVM Lists: PIN Online if unattended, PIN Offline elsewhere
 - Visa cards do not provide Offline Data Authentication
 - Card supports (DDA,CDA), terminal supports (DDA,CDA):
 - Terminal choose DDA
 - Terminal goes online if the offline authentication is failed

Card testing

- Balance testing for stolen cards
- <https://www.zdnet.com/article/hackers-abuse-magento-paypal-integration-to-test-validity-of-stolen-credit-cards/>


Payment methods > Add payment method

Add a payment method


 Add credit or debit card

Card number


#


MM / YY CVC 


29 June 2019

 **Google Services**
29 Jun, 11:25


26 June 2019


 **Google Services** £0
26 Jun, 10:50, Please unfreeze your card to make this transaction

 **Google Services** £0
26 Jun, 10:50, Please unfreeze your card to make this transaction




EXSBILLING.COM

 -7,22 \$



scupsm.com

 -47,99 \$

When hackers come first

- Nov, 2016, 40,000 accounts, 9,000 successfully

Tesco Bank says attack cost it £2.5m and hit 9,000 people

© 8 November 2016

[f](#) [t](#) [m](#) [✉](#) [Share](#)



Was it hacked?

Tesco did not use the "H" word in its statement and in interviews

Card testing

- 1 Dec 2016, Newcastle University
- https://eprint.ncl.ac.uk/file_store/production/230123/19180242-D02E-47AC-BDB3-73C22D6E1FDB.pdf
- Consecutive enumeration:
 - BIN (public DB)
 - PAN (online banking registration)
 - Expiry Date (refund, recipient of funds)
 - CVV (regular payment)
 - Postcode for AVS (different error)

TABLE I. VARIATION IN PAYMENT SECURITY SETTINGS OF ONLINE PAYMENTS WEBSITES

Number of attempts allowed	Sites with 2 fields (guess expiry date)	Sites with 3 fields (guess CVV2)	Sites with 4 fields (guess address postcode)	Sites with 3D Secure (safe from attack)	Total
0 to 5	2	23	2	-	27
6 to 10	20	238	18	-	276
11 to 50	2	28	3	-	33
Unlimited	2	2	2	-	6
3D Secure	-	-	-	47	47
Total	26	291	25	47	389

Card testing

- 1 Dec 2016, Newcastle University
- https://eprint.ncl.ac.uk/file_store/production/230123/19180242-D02E-47AC-BDB3-73C22D6E1FDB.pdf

FCA fines Tesco Bank £16.4m for failures in 2016 cyber attack

Press Releases | Published: 01/10/2018 | Last updated: 01/10/2018

- Consecutive
- PAN
- Expiry Date (refund, recipient of funds)
- CVV (regular payment)
- Postcode for AVS (different error)

SECURITY SETTINGS OF ONLINE SERVICES

	allowed	(guess expiry date)	(guess CVV2)	(guess address postcode)	Sites with 3D Secure (safe from attack)	Total
0 to 5		2	23	2	-	27
6 to 10		20	238	18	-	276
11 to 50		2	28	3	-	33
Unlimited		2	2	2	-	6
3D Secure		-	-	-	47	47
Total		26	291	25	47	389

Card testing

- July 2018, Monzo

<https://www.theguardian.com/money/2018/jul/07/heres-how-scammers-get-away-with-it>

Transaction attacks

On the Monday morning I visited Monzo's offices, just 12 hours earlier there had been a "pan enumeration" attack on its computer systems. This is where fraudsters, often based overseas, bombard a bank's computers, trying to guess passwords and logins, or attempting to do transactions by generating card expiry dates and three-digit CVCs (card verification codes) in the hope that some might break through.

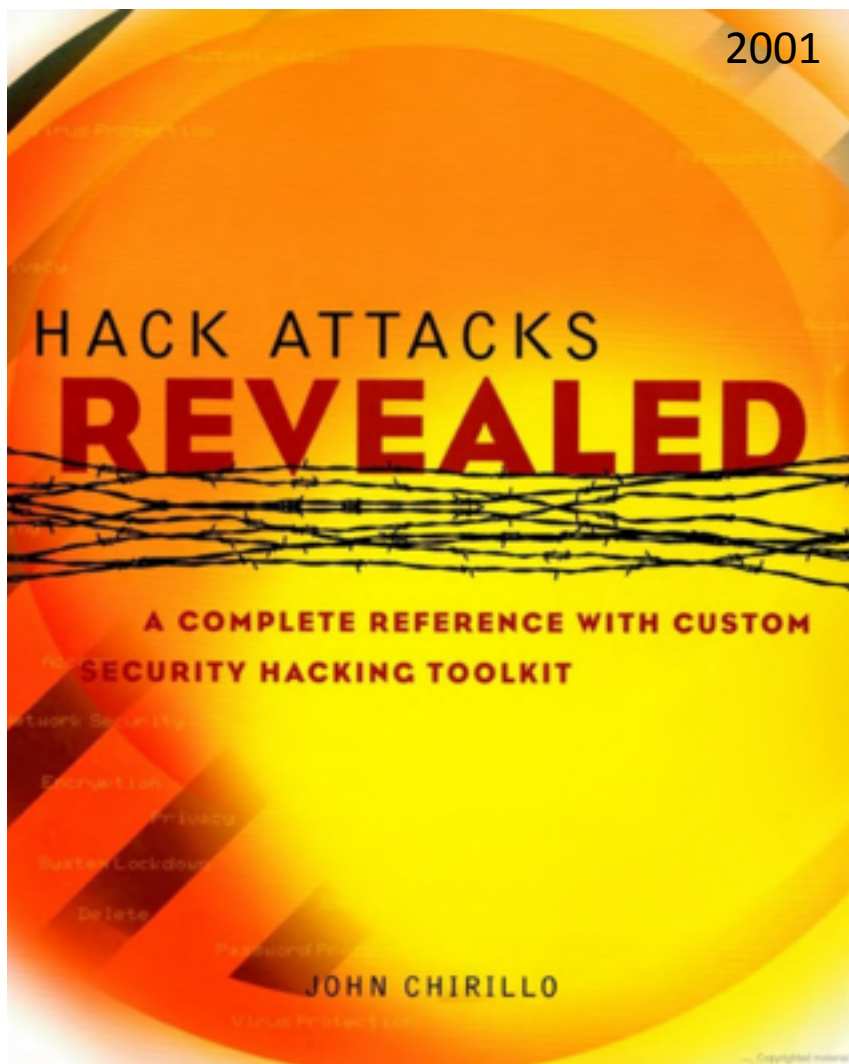
Advertisement



Meet the hub for teamwork in
Office 365 Business Premium

[CLICK HERE →](#)

Rounding



ZeroNights 2013



Practical exploitation of rounding vulnerabilities in internet banking applications

Adrian Furtună, PhD, OSCP, CEH
adif2k8@gmail.com

Y Hacker News [new](#) | [past](#) | [comments](#) | [ask](#) | [show](#) | [jobs](#) | [submit](#)

▲ Round error issue - produce money for free on itBit bitcoin exchange (hackerone.com)
70 points by waffle_ss on **Mar 3, 2017** [hide](#) | [past](#) | [web](#) | [favorite](#) | [60 comments](#)

Rounding

- 1 GBP = 1,30 USD
- 0.02 USD => float(0.0153; 2) == 0.02 GBP
- 0.02 GBP => float(0.026; 2) == to 0.03 USD
- Profit = 0.01 USD

Rounding

- 1 GBP = 1,30 USD
- 0.02 USD => float(0)
- 0.02 GBP => float(0)
- Profit = 0.01 USD



Rounding

- 1 GBP = 1,30 USD
- 0.02 USD => float(0.0153; 2) == 0.02 GBP
- 0.02 GBP => float(0.026; 2) == to 0.03 USD
- Profit = 0.01 USD

x10,000

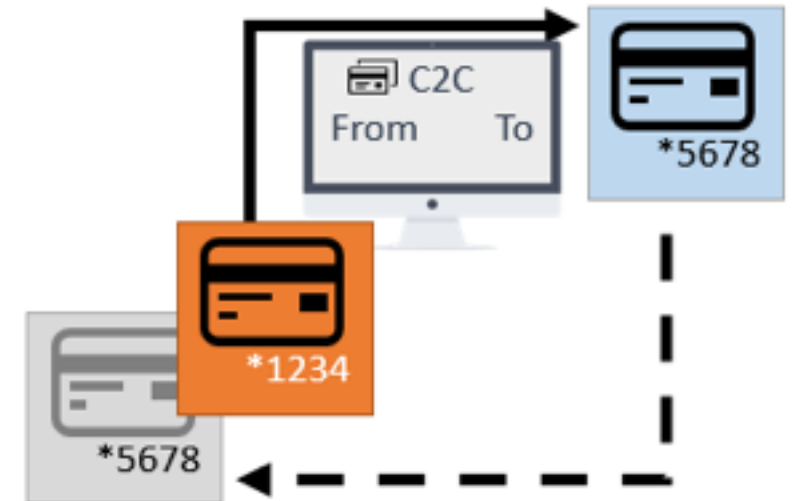
- OTP bypass
- Antifraud bypass
- Don't need to do everything manually

Stat

- Maximum amount per project – \$463,843 in 3 days (in live)
- In 2019 – 8/8 banks in Europe were *potentially* vulnerable to rounding,
one bank has confirmed the vulnerability


How to lose money during payment research


- Startup, which “allows you to spend money from any of your accounts using just one * Card” - *1234
- Connect any of your cards in the mobile app
- When you pay from the card *1234,
money will be withdrawn from the card you’ve chosen and connected (*5678)
- What if we will use Card2Card and send
From *1234 To *5678
- Just a regular transaction for *5678
- We will get a cashback!




How to lose money during payment research

- Send £100
- Money were withdrawn twice!
- Waited 5+ days
- Used 3 different card2card services
- Used 3 different cards, connected in the app

 **Waitrose** - £25.48
 12 Apr, 20:06


 **paysend** - £99
 12 Apr, 16:33

 **Paysend** - £100
 12 Apr, 16:27



i We have received your payment!
 We have received your payment, your money transfer is now being processed.

Send amount: 5.00 GBP to Timur Iunusov in United Kingdom
 Receive amount: 5.00 GBP
 Fee: 0.00 GBP
 Total to pay: 5.00 GBP
 Delivery method: Card Deposit
 Delivery details:
 Card Number: *5611
 Card Type: mastercard
 Cardholder Name (as printed on card): iunusov timur
 Payment method: Bank Card
 Payment reference:

Today

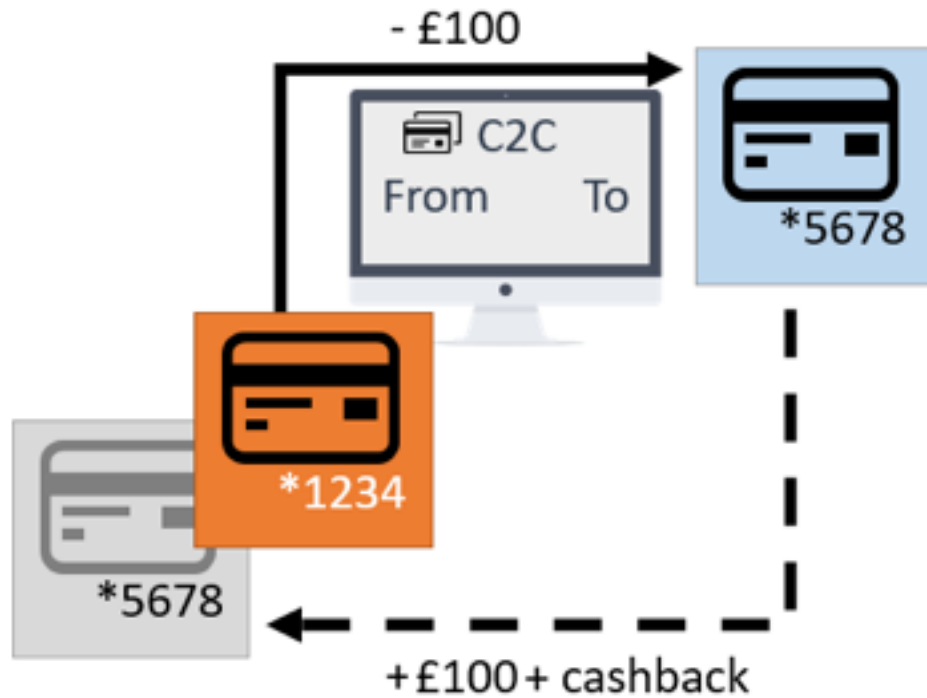
	Send Money 24 Ltd	5.00
--	-------------------	------

Saturday 15th June

	Sendmoney24	5.00
	Send Money 24 Ltd	0.01

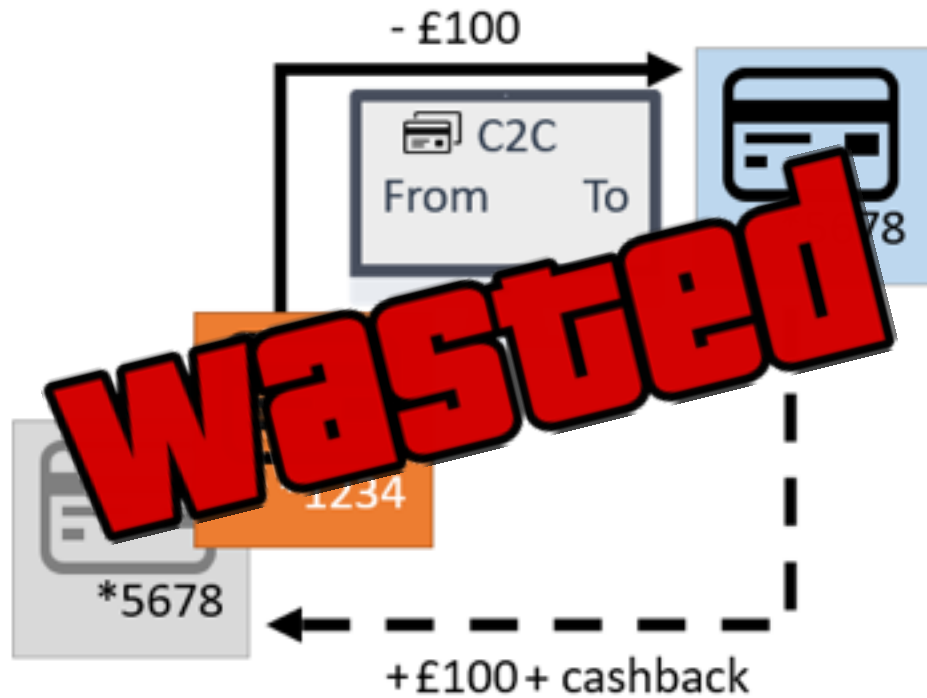
Date and time of operation	Amount in operation currency	Amount in card currency	Description
03.05.2019 13:35	100.00 RUB	100.00 RUB	Payment ir TINKOFF BANK CARD2 london GBR
03.05.2019 16:35	100.00 RUB	100.00 RUB	transfer c2c another bank

How to lose money during payment research




How to lose money during payment research

https://medium.com/@Tim_Y/how-to-lose-money-during-payment-research-or-in-searching-for-financial-ombudsman-5047bff89bc2



• 15 min ago ▾

Refunded £99.00 - Paysend  Met

Manage notifications



Who will pay?

- Not all vendors/banks are the same
- Risk-based model doesn't care "where's the money", but "how much money"

Bugbounty company from Google

1. Found vulnerability
2. Reported with lowest CVSS/out of scope
3. Thanks, \$\$\$
4. Now vulnerabilities won't be used in the wild

Bank "A"

1. Found vulnerability
2. Reported medium CVSS
3. It's not been used in the wild
4. Vulnerabilities still can be used in the wild



<https://www.cardpayments.fail>



info (at) cardpayments (dot) fail



@a66ot