Bitcoin Storage Security Survey: Wallets Cold Storage BIP032

Nicolas T. Courtois

- University College London, UK
Dr. Nicolas T. Courtois

1. cryptologist and codebreaker

2. payment and smart cards (e.g. bank cards, Oyster cards etc...)
LinkedIn

Your Groups (51) Reorder »

- Code Breakers

Members (712)

IACR Cryptographers
research seminar
=> In central London, runs EVERY WEEK!
public web page:
blog.bettercrypto.com / SEMINAR
or Google "UCL bitcoin seminar"

New Powerful Attacks On ECDSA In Bitcoin Systems
Posted by admin on 23 October 2014, 13:57 pm

There is a wave of new powerful cryptographic attacks on bitcoin systems.
My Whole Life:
Tried to educate people about security…
My Whole Life:

Tried to educate… AND frequently FAILED…
My Whole Life:

Crying Wolf!

51%, Elliptic Curve, OpenSSL...
It did NOT help,

The Wolf was allowed to operate
We failed to protect our DATA
We failed to protect our MONEY
Solution = Decentralized P2P
Solution = BlockChain

- Until recently, we’ve needed central bodies – banks, stock markets, governments, police forces – to settle vital questions.
  - Who owns this money?
  - ...
  - Now we have a small piece of [...] computer code that will allow people to solve the thorniest problems without reference to “the authorities”.


[11 June 2014]
John Nash - 1955

In 2012 the NSA declassified his hand-written letter:

> Now my general conjecture is as follows: For almost all sufficiently complex types of enciphering, especially where the information instructions given by different portions of the key interact complexly with each other in the determination of their ultimate effects on the enciphering, the key computation length increases exponentially with the length of the key.

He also says that:

> [...] the game of cipher breaking by skilled teams, etc., should become a thing of the past.” [...]
Elliptic Curve Crypto

“exponential security”
## ECC - Certicom Challenges [1997, revised 2009]

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECC2K-95</td>
<td>97</td>
<td>18322</td>
<td>$5,000</td>
</tr>
<tr>
<td>ECC2-97</td>
<td>97</td>
<td>180448</td>
<td>$5,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECC2K-108</td>
<td>109</td>
<td>$10,000</td>
<td>$10,000</td>
</tr>
<tr>
<td>ECC2-109</td>
<td>109</td>
<td>$10,000</td>
<td>$10,000</td>
</tr>
<tr>
<td>ECC2K-130</td>
<td>131</td>
<td>$20,000</td>
<td>$20,000</td>
</tr>
<tr>
<td>ECC2-131</td>
<td>131</td>
<td>$20,000</td>
<td>$20,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECC2K-163</td>
<td>163</td>
<td>$30,000</td>
<td>$30,000</td>
</tr>
<tr>
<td>ECC2-163</td>
<td>163</td>
<td>$30,000</td>
<td>$30,000</td>
</tr>
<tr>
<td>ECC2-191</td>
<td>191</td>
<td>$40,000</td>
<td>$40,000</td>
</tr>
<tr>
<td>ECC2K-238</td>
<td>239</td>
<td>$50,000</td>
<td>$50,000</td>
</tr>
<tr>
<td>ECC2-238</td>
<td>239</td>
<td>$50,000</td>
<td>$50,000</td>
</tr>
<tr>
<td>ECC2K-358</td>
<td>359</td>
<td>$100,000</td>
<td>$100,000</td>
</tr>
<tr>
<td>ECC2-353</td>
<td>359</td>
<td>$100,000</td>
<td>$100,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCp-97</td>
<td>97</td>
<td>71982</td>
<td>$5,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCp-109</td>
<td>109</td>
<td>$10,000</td>
<td>$10,000</td>
</tr>
<tr>
<td>ECCp-131</td>
<td>131</td>
<td>$20,000</td>
<td>$20,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCp-163</td>
<td>163</td>
<td>$30,000</td>
<td>$30,000</td>
</tr>
<tr>
<td>ECCp-191</td>
<td>192</td>
<td>$40,000</td>
<td>$40,000</td>
</tr>
<tr>
<td>ECCp-239</td>
<td>239</td>
<td>$50,000</td>
<td>$50,000</td>
</tr>
<tr>
<td>ECCp-359</td>
<td>359</td>
<td>$100,000</td>
<td>$100,000</td>
</tr>
</tbody>
</table>

**TOTAL = 725,000 USD**

Nicolas T. Courtois 2009-2014
P vs. NP

• If you solve P vs. NP it: 1 M$.

• Nobel price, Abel price in mathematics: roughly 1M$

• Break bitcoin ECC: About 3 BILLION $.
<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECC2K-95</td>
<td>97</td>
<td>18322</td>
<td>$5,000</td>
</tr>
<tr>
<td>ECC2-97</td>
<td>97</td>
<td>180448</td>
<td>$5,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECC2K-108</td>
<td>109</td>
<td>$1.3 \times 10^6$</td>
<td>$10,000</td>
</tr>
<tr>
<td>ECC2-109</td>
<td>109</td>
<td>$2.1 \times 10^7$</td>
<td>$10,000</td>
</tr>
<tr>
<td>ECC2K-130</td>
<td>131</td>
<td>$2.7 \times 10^9$</td>
<td>$20,000</td>
</tr>
<tr>
<td>ECC2-131</td>
<td>131</td>
<td>$6.6 \times 10^{10}$</td>
<td>$20,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCp-97</td>
<td>97</td>
<td>71982</td>
<td>$5,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCp-109</td>
<td>109</td>
<td>$9.0 \times 10^6$</td>
<td>$10,000</td>
</tr>
<tr>
<td>ECCp-131</td>
<td>131</td>
<td>$2.3 \times 10^{10}$</td>
<td>$20,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCp-163</td>
<td>163</td>
<td>$2.48 \times 10^{15}$</td>
<td>$30,000</td>
</tr>
<tr>
<td>ECCp-163</td>
<td>163</td>
<td>$2.48 \times 10^{15}$</td>
<td>$30,000</td>
</tr>
<tr>
<td>ECCp-191</td>
<td>191</td>
<td>$4.07 \times 10^{19}$</td>
<td>$40,000</td>
</tr>
<tr>
<td>ECCp-239</td>
<td>239</td>
<td>$6.83 \times 10^{26}$</td>
<td>$50,000</td>
</tr>
<tr>
<td>ECCp-359</td>
<td>359</td>
<td>$7.88 \times 10^{44}$</td>
<td>$100,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Field size (in bits)</th>
<th>Estimated number of machine days</th>
<th>Prize (US$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCp-163</td>
<td>163</td>
<td>$2.3 \times 10^{15}$</td>
<td>$30,000</td>
</tr>
<tr>
<td>ECCp-191</td>
<td>192</td>
<td>$4.8 \times 10^{19}$</td>
<td>$40,000</td>
</tr>
<tr>
<td>ECCp-239</td>
<td>239</td>
<td>$1.4 \times 10^{27}$</td>
<td>$50,000</td>
</tr>
<tr>
<td>ECCp-359</td>
<td>359</td>
<td>$3.7 \times 10^{45}$</td>
<td>$100,000</td>
</tr>
</tbody>
</table>

secp256k1

NOT INCLUDED

no price if you break it 😎
Timely Denial

Dan Brown, chair of SEC [Certicom, Entrust, Fujitsu, Visa International…]

``I did not know that BitCoin is using secp256k1. I am surprised to see anybody use secp256k1 instead of secp256r1'',

September 2013,
https://bitcointalk.org/index.php?topic=289795.80
<table>
<thead>
<tr>
<th>Used/recommended by</th>
<th>secp256k1</th>
<th>secp256r1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitcoin, anonymous founder, no one to blame…</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>SEC Certicom Research</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>TLS, OpenSSL</td>
<td></td>
<td>Y 98.3% of EC</td>
</tr>
<tr>
<td>U.S. ANSI X9.63 for Financial Services</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>NSA suite B, NATO military crypto</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>U.S. NIST</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>IPSec</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>OpenPGP</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>Kerberos extension</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>Microsoft implemented it in Vista and Longhorn</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>EMV bank cards XDA [2013]</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>German BSI federal gov. infosec agency, y=2015</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>French national ANSSI agency beyond 2020</td>
<td></td>
<td>Y</td>
</tr>
</tbody>
</table>
Wanna Bet?

Bitcoin Cryptography Broken in 2015

Category: Bitcoin
By NCourtois ★★★★★

Description

The digital signature scheme of bitcoin with SHA256+secp256k1 ECDSA will be broken before 1 September 2015 by cryptography researchers. The attack should allow to forge digital signatures for at least a proportion of 1/1 million bitcoin users and steal money from them. It should be done faster than $2^{100}$ point additions, a total including the time to examine the data.

Decision Logic

<table>
<thead>
<tr>
<th>YES</th>
<th>NO</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Volume:</strong></td>
<td><strong>Volume:</strong></td>
</tr>
<tr>
<td>B 0.140</td>
<td>B 0.189</td>
</tr>
<tr>
<td><strong># of Bets:</strong></td>
<td><strong># of Bets:</strong></td>
</tr>
<tr>
<td>3</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PAYOUT</th>
<th>ROI</th>
</tr>
</thead>
<tbody>
<tr>
<td>B 0.00</td>
<td>0%</td>
</tr>
</tbody>
</table>

*assumes current weight and volumes

Place Anonymously

SHA256, ECDSA, ECDL, secp256k1
Is Bitcoin Improving?
Bitcoin Troubles

- Crypto gets broken?
- Monetary policy: genius, weird or mad?
- 51% attacks and double spending: easy!
- P2P network in decline (XX,000=>5,000)
- Slow speed
- Poor Anonymity

• Payment fees decline/stable
Better Security Will Prevail?

NOT obvious, and even LESS obvious in financial systems.

A right amount of insecurity:
• allows you to sell insurance,
• trains our survival and cybersecurity skills,
• creates lots of interesting jobs for our students,
• possibly avoids criminals to engage in “more violent” crime…
Better “Money” Will Prevail?

Crypto engineers like us sometimes naively hope that “better” currencies will drive “not so good” currencies out of business.

In fact the Gresham-Copernicus Law [1517] says exactly otherwise!

**Bad currencies** DO frequently drive better currencies out of business.
Better “Money” Will Prevail?

The “bad” option is also happening with bitcoin: it has gained excessive popularity NOT because it was technically very good (it never was) or had solid intrinsic value, or it was fast and convenient (it never was).

It has thrived because it has created huge expectations which temporarily bitcoin competitors could not meet. Bitcoin remained the obvious choice, a sort of natural monopoly.
Network Effects!

Antonopoulos [former UCL student] points out that
"when you have a technology that is ‘good enough’ that achieves network scale [...] good enough suddenly becomes perfect"

“I don’t see any altcoin displacing it”, he says.

If bitcoin crashes, again according to Antonopoulos it will be rather because “we blow it up by accident”.

[L.A. Bitcoin Meetup Jan 2014]
Our Works on Bitcoin

cf. also blog.bettercrypto.com

- Poster: http://www.nicolascourtois.com/bitcoin/POSTER_100x_Secrypt2014_v1.0.pdf
Actually I show that quite possibly
bitcoin is EXEMPT from destruction [natural monopoly].

=> Whatever is Bad with bitcoin is even worse with most alt-coins.
Bitcoin vs.
Security Engineering
We postulate:

1. Open design.

2. Least Common Mechanism

3. Assume that attacker controls the Internet [Dolev-Yao model, 1983].

4. The specification should be engineered in such a way that it is hard for developers to make it insecure on purpose (e.g. embed backdoors in the system).
Open Design ≠ Open Source

Examples: cryptography such as SHA256 (used in bitcoin) is open source but NOT open design – it was designed behind closed doors!
Open Source vs. Closed Source and Security
Secrecy: Very frequently an obvious business decision.

- Creates entry barriers for competitors.
- But also defends against hackers.
Kerckhoff’s principle: [1883]

“The system must remain secure should it fall in enemy hands …”
Kerckhoffs’ principle: [1883]

Most of the time: incorrectly understood. Utopia. *Who can force companies to publish their specs?***

No obligation to disclose.

- Security when disclosed.
- Better security when not disclosed.
Yes (1,2,3,4):

1. Military:
   layer the defences.
Yes (2):

2) Basic economics: these 3 extra months (and not more 😏) are simply worth a lot of money.
Yes (3):

3) Prevent the erosion of profitability / barriers for entry for competitors / “inimitability”
Avoid Legal Risks

- companies they don't know where their code is coming from, they want to release the code and they can't because it's too risky!
  - re-use of code can COMPROMISE own IP rights and create unknown ROYALTY obligations (!!!)
  - clone/stolen code is more stable, more reliable, easier to understand!
What’s Wrong with Open Source?
Kerckhoffs principle:

- Rather WRONG in the world of smart cards…
  - Reasons:
    - side channel attacks,
    - PayTV card sharing attacks

- But could be right elsewhere for many reasons…
  - Example:
    - DES, AES cipher, open-source, never really broken
    - KeeLoq cipher, closed source, broken in minutes…
*Kerckhoffs principle vs. Public Key Crypto vs. Financial Cryptography

• In Public Key Cryptography one key **CAN** be made public. In practice this means that
  – some **group** of people has it
  – **NO obligation** to disclose, to make it really public (and it is almost never done in serious financial applications)

• Full disclosure for public keys is unbelievably stupid…
  – cf. next slide!
Do NOT Disclose Public Keys!

• Full disclosure for public keys is simply BAD security engineering and BAD security management.

• Examples:

  • ATMs have like 6 top-level public keys, not really public though

  • in Bitcoin: the public key can remain a secret for years, only a hash is revealed, this is BRILLIANT key management which makes Bitcoin MUCH more secure that it would otherwise be!

  • it does solve the problem raised by Diffie at CataCrypt in San Francisco:

    HOW DO YOU PROTECT AGAINST UNKNOWWN ATTACKS?
Workshop on catastrophic events related to cryptography and their possible solutions

Technical Program

Venue: Grand Hyatt San Francisco, Union Square, 345 Stockton Street, downtown San Francisco: room Fillmore A - Theatre Level  http://grandsanfrancisco.hyatt.com
October 29, 2014  (together with IEEE Conference on Communications and Network Security (CNS)

08:15 – 08:25  Opening Remarks: Jean-Jacques Quisquater (UCL, Belgium)
Breaking News

NSA Plans To Retire Current Cryptography Standards

Posted by admin on 15 September 2015, 3:26 pm

Breaking news:

the cryptography that we all know and use, such AES-128, SHA-1 and SHA-256, RSA/DH, and the most commonly used elliptic curve P-256 (a.k.a. secp256r1) are NO LONGER wholeheartedly supported by the NSA. In fact most of these, if not all, are not quite recommended anymore.

Until now and for the last 10+ years the NSA and the NIST urged everybody to use these things.

Now the NSA has a very different message:

• There will be a transition to new crypto algorithms coming very soon.
Introducing Bitcoin
Bitcoin In A Nutshell

- Bitocoin are cryptographic tokens, binary data = 010100110101010…
  - Stored by people on their PCs or mobile phones
- Ownership is achieved through digital signatures:
  - You have a certain cryptographic key, you have the money.
  - Publicly verifiable, only one entity can sign
- Consensus-driven, a distributed system which has no central authority
  - A major innovation: financial transactions CAN be executed and policed without trusted authorities.
  - Bitcoin is a sort of financial cooperative or a distributed business.
- Based on self-interest:
  - A group of some 100 K people called Bitcoin miners own the Bitcoin “infrastructure” which has costed > 1 billion dollars (my estimation)
  - They make money from newly created bitcoins and fees
  - At the same time they approve and check the transactions.
  - A distributed electronic notary system
Two Key Concepts

- Initially money are attributed through **Proof Of Work (POW)** to one public key A
  - to earn bitcoins one has to “work” (hashing) and consume energy (pay for electricity)
  - now in order to cheat one needs to work even much more (be more powerful than the whole network), more precisely:

- Money transfer from public key A to public key B:
  - like signing a transfer in front of one notary which confirms the signature,
  - multiple confirmations: another notary will re-confirm it, then another, etc…
  - we do NOT need to assume that ALL these notaries are honest.
    - at the end it becomes too costly to cheat
In Practice
WALLETS

- **Wallet**: file which stores your “money”.
- A Bitcoin client App is also called a wallet.
Digital Currency

Bitcoin is a

=> PK-based Currency:

- bank account = a pair of public/private ECDSA keys

- spend money = produce a digital signature
Main Problem:

Bitcoins can be “spent twice”.

Avoiding this “Double Spending” is the main problem when designing a digital currency system.
Block Chain
Bitcoin Mining

- Minting: creation of new currency.
- Confirmation+re-confirmation of older transactions

Ownership:
- “policed by majority of miners”:

```plaintext
HASH
```

must start with 64 zeros

data from previous transactions

miner’s public key

RNG
Block Chain

Def: 🚴‍♂️
A transaction database shared by everyone.

Also a ledger.
Every transaction since ever is public.
It is possible to almost totally separate:

- **Miner nodes** – Hashing with public keys
- **Peer Nodes** – Relay and store transactions and blocks
- **Wallet Nodes** – Store and release funds, focus on management of private keys, master keys etc...

```
public ledger
```

```
burn
```
Bitcoin Address

To: 1K2CcfWYW5sBL2xSeQWXpcmjPCgoXdi36
Amount: 1.0 BTC
SEND
Ledger-Based Currency

A “Bitcoin Address” = a sort of equivalent of a bank account.

Remarks:

• PK is NOT public!
• only $H$(public key) is revealed!
• PK remains confidential until some money in this account is spent.
• SK = private key: always keep private, allows transfer of funds.
Bitcoin Ownership

Amounts of money are attributed to public keys. Owner of a certain “Attribution to PK” can at any moment transfer it to some other PK (== another address).

Destructive, cannot spend twice:
*Multi-Signature Addresses
MultiSig = Addresses Starting with 3

Bitcoin can require *simultaneously* several private keys, in order to transfer the money.

- For example 2 out of 3 signatures are required to spend bitcoins.
- The keys can be stored on different devices (highly secure).
- Can work without backups: if one device is lost, use other devices to transfer bitcoins to a new multisig address with another set of devices...
Multi-Sig Concept is NOT new…

1993
Efficient multi-signature schemes for cooperating entities
Olivier Delos¹ and Jean-Jacques Quisquater²

1983
K. Itakura, K. Nakamura:
A public-key cryptosystem suitable for digital multi-signatures
Cryptographic Security of ECDSA in Bitcoin

BTC Transfer

To: 1K2CcfWYW5sBL2xSeQWXpcmjPCgoXdi36
Amount: 1.0 BTC
SEND
Bitcoin Transfer

Transactions have multiple inputs and multiple outputs.

Transaction Signed by All Owners with their SK

Input Bitcoin Addresses
0.2 BTC  1.3 BTC

Output Bitcoin Addresses
1.0 BTC  0.499 BTC

+ Fees
0.001 BTC
Transaction Scripts
## Cryptographic Security of ECDSA in Bitcoin

### Signed Tx / Final Tx

**byte by byte** (similar but not identical to raw blocks seen before)
(this is done twice, with different scriptSig)

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>01 00 00 00</td>
</tr>
<tr>
<td>input count</td>
<td>01</td>
</tr>
<tr>
<td>input previous output hash</td>
<td>48 4d 40 d4 5b 9e a0 d5 52 fc a8 25 8a b7 ca a4 25 41 eb 52 97 58 57 f9 ff b5 0c d7 32 c8 b4 81</td>
</tr>
<tr>
<td>input previous output index</td>
<td>00 00 00 00</td>
</tr>
<tr>
<td>input script length</td>
<td>scriptSig length 1 byte</td>
</tr>
<tr>
<td>input scriptSig</td>
<td>script containing signature</td>
</tr>
<tr>
<td>input sequence</td>
<td>ff ff ff ff</td>
</tr>
<tr>
<td>output count</td>
<td>01</td>
</tr>
<tr>
<td>output value</td>
<td>52 64 01 00 00 00 00 00</td>
</tr>
<tr>
<td>output script length</td>
<td>scriptPubKey length 1 byte</td>
</tr>
<tr>
<td>output scriptPubKey</td>
<td>script containing destination address</td>
</tr>
<tr>
<td>block lock time</td>
<td>00 00 00 00 (not widely used)</td>
</tr>
</tbody>
</table>
## Second $\text{scriptSig}$

$$\text{sign+PKey}$$

$$\text{len} = 1 + 71 + 1 + 65 = 138 \text{ BUT NOT ALWAYS!}$$

<table>
<thead>
<tr>
<th>PUSHDATA 47</th>
<th>47</th>
</tr>
</thead>
<tbody>
<tr>
<td>sequence</td>
<td>30</td>
</tr>
<tr>
<td>length</td>
<td>44</td>
</tr>
<tr>
<td>integer</td>
<td>02</td>
</tr>
<tr>
<td>length</td>
<td>20</td>
</tr>
<tr>
<td>X r</td>
<td>2c b2 55 b5 10 70 7b f4 92 46 c3 5d d3 d1 6f c4 54 61 8c 58 ec 0a 0f 44 48 a6 76 c5 4f f7 13</td>
</tr>
<tr>
<td>integer</td>
<td>02</td>
</tr>
<tr>
<td>length</td>
<td>20</td>
</tr>
<tr>
<td>Y s</td>
<td>6c 56 24 d7 52 61 fc ef 46 18 28 4e ad 6f 08 67 8a c0 5b 13 c8 42 35 f1 65 4e 5a d1 88 23 3e 82</td>
</tr>
<tr>
<td>SIGHASH_ALL</td>
<td>01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PUSHDATA 41</th>
<th>41</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>04</td>
</tr>
<tr>
<td>public key</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>14 e9 01 b2 32 8f 17 44 2c 0b 83 10 d7 87 bf 3d 8a 40 4c fb d0 70 4f 13 5b 62 1d b2 d3 ee 75 13</td>
</tr>
<tr>
<td>Y</td>
<td>10 f9 81 92 6e 53 a6 e8 c3 9b d7 d3 fc fd 57 8c 54 3c he 40 93 ba c0 63 88 f2 65 1d 2a ac bf cd</td>
</tr>
</tbody>
</table>
Is Bitcoin Secure?

Satoshi claimed it is…
Wallets
Bottom Line

Main Functionality:
- Private Key Generation
- Export public key
- ECDSA sign

-optional:
  • sign full BTC transactions
  • confirm recipient on the screen!
    (huge classical pb with all smart cards and digital signature devices,
     Ledger has a clever solution: regurgitates inputs on another device USB keyboard)

BTChip HW1
hardwarewallet.com

Ledger
ledgerwallet.com

Trezor
bitcointrezor.com
Ledger HW.1

Visit website  Source code

Control over your money
Variable validation
New app
Very secure environment
Variable privacy

HW.1 is a hardware wallet built upon a ST23YT66 banking smartcard platform. It keeps the user private keys safe, validates transactions, can be used as a secure prepaid card or a multisignature party. While not open-source, it can be deterministically validated.
**Features of USB card ST23YT66**

NESCRIPTY crypto-processor for PK crypto

- 900 ms for 1 ECDSA signature
- 900 ms for key gen
- Encrypts private keys on the card (`content` key) 3DES CBC

- Content key can be protected with "a GlobalPlatform Secure Channel" authentication mechanism
Trezor

by Satoshi Labs Prague, CZ

released March 2014

+ display: know to whom you send the money!
+- has open source firmware: https://github.com/trezor/trezor-mcu

TREZOR is a hardware wallet providing a high level of security without sacrificing convenience. Unlike cold storage, TREZOR is able to sign transactions while connected to an online device. That means spending bitcoins is secure even when using a compromised computer.
+ Trezor Lite App

Allows to see your money when you don’t have your device with you!

Based on **BIP032 audit capability**

=> quite dangerous: see

Wallets and Key Management

BIP032

\[ \text{Parent Ext. Private} \]
\[ k \]
\[ k_L, k_R \]
\[ k_{\text{Private}} \]
\[ x_{\text{chain}} \]
\[ .G \]
\[ K_{\text{public}} \parallel i \]
\[ M \]
\[ K \]
\[ HM \]

\[ \text{mod } q \]
\[ k_i \]
\[ y \]
\[ y_L, y_R \]
\[ y_{\text{Private}}, y_{\text{Chain}} \]

\[ \text{Parent Ext. Public} \]
\[ K \]
\[ K_L, K_R \]
\[ K_{\text{public}} \parallel i \]
\[ M \]
\[ K \]
\[ HM \]

\[ .G \]
\[ c_i \]
\[ \text{right key } = k_i \cdot G \]

75
Ledger

- have their own operating system!
  - closed source, their Chrome front-end is open source
  - due to the current JavaCard limitation:
    - cannot implement deterministic ECDSA (RFC6979)
- bitcoin tx processing implemented inside (unlike HW.1)
  - claimed to be a “more secure” evolution of HW.1
- communicates with Google Chrome directly, no middleware
- data retention: 30 years
- open: no NDA for any wallet to support this
It Implements:

- Standard Multisig, P2SH style (BIP016)
- BIP032: HD Wallets
  ⇒ danger?, they have fixed it!
  ⇒ solution: implements RFC 6979, deterministic signatures
- BIP039: seed mnemonic (list of words in English)
- BIP044: specific wallet structure
Security

• master backup
  – printed card with master private seed
  + long passphrase to be written on paper (used only to recover)
  – recovery also possibly if the hardware is lost
    • standard method BIP39, no lock-in, can be recovered on 3rd party soft/hard
  – enter wrong PIN 3 times=>all data are claimed to be erased
  – claimed totally anonymous
    • except browser IP address will be revealed when you send Tx to the network

• each device is paired with a printed card A=>3, to be kept with the wallet,
  – this card=second factor authn. (malware cannot use the device)
  – duo edition has the same card: can create 2 identical hardware wallets
  – Pb: PIN code is entered on a PC: BUT
    • to sign a transaction, need to enter correspondance codes A=>3
      "based on a random sampling of the payment address"
CoinKite

- card + terminal with HSM
- supports multisig

Pb.
- “each new member receives a "welcome email" which contains the "xpubkey" (extended public key) for their deposits.”
- super dangerous!
Are Known Wallet Solutions Secure?
Incidents at Operation: Bad Randoms
Bad Randoms

First publicized by Nils Schneider:
28 January 2013

D47CE4C025C35EC440BC81D99834A624875161A26BF56EF7FDC0F5D52F843AD1
⇒ repeated more than 50 times…

Used twice by the SAME user!
ECDSA Signatures

Let $d$ be a private key, integer $\text{mod } n = \text{ECC [sub-]group order}$.

- Pick a random non-zero integer $0 < a < n - 1$.
- Compute $R = a \cdot P$, where $P$ is the base point (generator).
- Let $r = (a \cdot P)_x$ be its x coordinate.
- Let $s = (H(m) + d \cdot r) \pmod a \pmod n$.

The signature of $m$ is the pair $(r, s)$.

(512 bits in bitcoin)
Groups and ECC

**Attack – 2 Users**

random \( a \): must be kept secret!

\[
R = a \cdot P
\]

\[
s = \left( \frac{H(m) + dr}{a} \right) \mod n
\]

\[
(r, s) \mod n
\]

same \( a \) used twice => detected in public blockchain =>

\[
\frac{s_1 a - H(m_1)}{d_1} = r = \frac{s_2 a - H(m_2)}{d_2} \mod n
\]

\[
r(d_1 - d_2) + a(s_1 - s_2) = H(m_2) - H(m_1) \mod n
\]

has already happened 100 times in Bitcoin

each person can steal the other person’s bitcoins!

=> any of them CAN recompute \( k \) used

84
Cryptographic Security of ECDSA in Bitcoin

Our Graph Model

2 users have used the same random
Cryptographic Security of ECDSA in Bitcoin

Our Online Database

```
9e199edb08bec948740e84cc6f91f0bbufe36bc5f10546e0c1a6e2655f2c6019 4x 07Jan15-07Jan15
  1x /1LR63Z94Lz29XVvnwaWi4JViREpFk4BFZf
  1x /12rdRMTZQ6unVucRnPtSmZRoqp2MVgBmh9
  1x /1BPVuwza9pDHpbzUBMLUyhyV7PnuF2iJGx
  1x /147rzbsdsqc2YKeGQRUs3jaCxyufVRz8Kh

47b1ce535f6331d07759eeaaafab4c1a276cdafsa86245a7bf61f29236619367 7x 04Jan15-04Jan15
  1x /1DDessF6x8s1RFN116aZ36PzVRrj5YUFA7
  1x /1KdpXyEtFsr9Sugf3wo5bS9328y5cZ1oXK
  1x /1GMu2kbq8sY5ZLXkPfbVJzakddHo2Vjmde
  1x /1KjLEUrdUjN7a2N6B8xY3V6bL1U1UJpCCA
```

```
337956/tx26/i3 337956/tx26/i1
337956/tx26/i2 337956/tx26/i0
337458/tx25/i1 337458/tx25/i0
337458/tx25/i5 337458/tx25/i2
```
Groups and ECC

**Attack – Same User**

random \(a\): must be kept secret!

\[
RNG \\
\rightarrow \text{random } a \\
\rightarrow R = a \cdot P \\
\rightarrow r \\
\rightarrow s = \frac{(H(m) + dr)}{a} \mod n \\
\rightarrow (r,s)
\]

same \(a\) used twice by the same user \((d_1 = d_2)\). In this case we have:

\[
(s_1 a - H(m_1)) = r d = (s_2 a - H(m_2)) \mod n
\]

\[
=> a = \frac{(H(m_1) - H(m_2))}{(s_1 - s_2)} \mod n \text{ AND now }
\]

\[
d = \frac{(sa - H(m))}{r} \mod n
\]

anybody can steal the bitcoins!

has also happened 100 times in Bitcoin
Have These Problems Stopped in 2013?

Lots of problems in May 2012, fixed.

2013: Android bug was fixed…

And then there was another MASSIVE outbreak…

And then another…
Dec. 2013

At 30C3 conference in Germany on 28 Dec 2013, Nadia Heninger have reported that they have identified a bitcoin user on the blockchain which has stolen some 59 BTC due to these bad randomness events,

The money from the thefts is stored at:
https://blockchain.info/address/1HKywxiL4JziqXrzLKhmb6a74ma6kxbSDj

Still sitting there, he is NOT trying to spend it… too famous? Afraid to be traced and caught?
Cryptographic Security of ECDSA in Bitcoin

Second Major Outbreak – May 2014

Android RNG bug
Cryptographic Security of ECDSA in Bitcoin

Bad Randoms in Bitcoin 02May11-05Jan15
cf. eprint.iacr.org/2014/848

\[ y = \text{public key} \]

Third Major Outbreak
December 2014
200,000 USD stolen
by an “ethical thief”
at Blockchain.info
Cryptographic Security of ECDSA in Bitcoin

Dodgy Security Advice By A Thief

'Good Samaritan' Blockchain Hacker Who Returned 267 BTC Speaks Out

Jon Southurst (@southtopia) | Published on December 12, 2014 at 14:41 GMT

“johoe recommends a client that employs HD (hierarchical deterministic) wallets, such as Bread Wallet on iOS and Armory, Electrum or Wallet32 on Android.”
'Good Samaritan' Blockchain Hacker Who Returned 267 BTC Speaks Out

Jon Southurst (@southtopia) | Published on December 12, 2014 at 14:41 GMT

“johoe recommends a client that employs HD (hierarchical deterministic) wallets, such as Bread Wallet on iOS and Armory, Electrum or Wallet32 on Android.”

Is he not aware that these solutions can lead to thefts at a much larger scale?
'Good Samaritan' Blockchain Hacker Who Returned 267 BTC Speaks Out

johoe recommends a client that employs HD (hierarchical deterministic) wallets, such as Bread Wallet on iOS and Armory, Electrum or Wallet32 on Android.

Is he not aware that these solutions can lead to thefts at a much larger scale?

=> see our paper 2014/848.
Cryptographic Security of ECDSA in Bitcoin

Most Recent Bad Randoms

From my own scan:

c471b1ce535f6331d07759eeaaefab4c1a276cdafa86245a7bf61f29236619367

Appears 7 times in block 337458
4 January 2015

Used by different users…
New Risks
So What?

Previous attacks:

- Classical bad random attacks typically concern only very few bitcoin accounts, and only some very lucky holders of bitcoins can actually steal other people's bitcoins.

- Only a few hundred accounts in the whole history of bitcoin were affected until today.
Advanced Attacks
October 2014
eprint/2014/848
The Really Scary Attacks

New attacks [Courtois et al. October 2014]

=> under certain conditions
   ALL bitcoins in cold storage
can be stolen

=> millions of accounts potentially affected.
Private Key Recovery Combination Attacks: On Extreme Fragility of Popular Bitcoin Key Management, Wallet and Cold Storage Solutions in Presence of Poor RNG Events

Nicolas T. Courtois\textsuperscript{1} \quad Pinar Emirdag\textsuperscript{2} \quad Filippo Valsorda\textsuperscript{3}

\textsuperscript{1} University College London, UK
\textsuperscript{2} Independent market structure professional, London, UK
\textsuperscript{3} CloudFlare, London, UK

Abstract. In this paper we study the question of key management and practical operational security in bitcoin digital currency storage systems. We study the security two most used bitcoin HD Wallet key management solutions (e.g. in BIP032 and in earlier systems). These systems have extensive audit capabilities but this property comes at a very high price. They are excessively fragile. One small security incident in a remote corner of the system and everything collapses, all private keys can be recovered and ALL bitcoins within the remit of the system can be stolen. Privilege escalation attacks on HD Wallet solutions are not new. In this paper we take it much further. We propose new more advanced combination attacks in which the security of keys held in cold storage can be compromised without executing any software exploit on the cold system, but through security incidents at operation such as bad random number or related random events.

In our new attacks all bitcoins over whole large security domains can be stolen by people who have the auditor keys which are typically stored in hot systems connected to the Internet and can be stolen easily. Our combination attacks allow to recover private keys which none of the
HD Wallets = Trees
2 Trees Connected Due to Bad Randoms

2 users have used the same random

[cycle]
More Cycles
Bad RNG ECDSA Bitcoin Etc

Even More
Is There a Fix?

**Solution:** RFC6979 [Thomas Pornin]

**BOTTOM LINE:**
If you have NOT implemented RFC6979, you should be scared by this talk…
RFc6979 [Pornin] = 5+ applications of HMAC

\[ \text{HMAC-SHA256} \]

(normal normally a loop BUT not needed for 256 bits output k)

ECDSA
Which Systems Are Affected?

Solution: RFC6979 [Pornin]

- Already applied by
  - Electrum, Multibit, Trezor

- Patched very lately:
  - blockchain.info – insecure,
  - Bitcoin Core – patch was applied 18M after being approved…