BitFin 2014 – Dublin, Ireland

Introduction to Bitcoin Transactions and Bitcoin Security

July 3, 2014

Alan Reiner
Founder & CEO
Armory Technologies, Inc.

List of Topics

- Introduction
- Bitcoin basics
- The marriage of cryptography and money
- Short intro to Bitcoin Security

Who Am I?

- **Alan Reiner**
  - Creator of Armory Bitcoin Wallet
  - Founder and CEO of Armory Technologies, Inc.
  - "**e**to**the**i**pi**" on the bitcointalk.org

- **Mathematician, Statistician, SW Developer**
  - With a sprinkling of cryptography and data mining

- **Have been part of the Bitcoin community since 2011**
  - Contributed to documentation, standards, security discussions, etc, on the "*Development & Technical Discussion*" forum

- **A huge nerd!** (you have to be to do what I do)
  - Sub-category: "ultra-paranoid crypto-nerd"

What is Armory?

- **Armory Bitcoin Wallet is an free, open-source desktop application for securing Bitcoins yourself**
  - One of five such applications featured on bitcoin.org
  - Has been one of the primary alt wallets since Dec 2011
  - Windows, Linux & Mac

- **Known for "security at all costs"**
  - Sometimes "convenience" is one of those costs...
  - Currently a tool tailored to advanced/power users

Armory Technologies, Inc.

- **Develops and maintains Armory Bitcoin Wallet**
  - Focuses on innovating security best practices
  - Enterprise security consulting
    - Integration of Armory into business platform
    - Security configuration, training and best practices

- **A complete platform for enterprise Bitcoin businesses:**
  - End-to-end **cold storage** management interface
  - Flexible, decentralized **multi-signature** interface
  - Watch-only wallets for monitoring funds (CFO, auditors, etc)
  - Graphical interface for executive management
  - Daemon/API for network and services integration
  - Consulting, support and training for enterprise clients

Bitcoin Storage Options

- **With Bitcoin, now "data" <u>is</u> "money"**
  - And "money" is "data"
  - A 32-byte secret number can control **$billions**
  - Raises the stakes of computer and network security
  - Money now stored directly on phone, computer, paper, etc

- **Store Bitcoin yourself**
  - **+** Full control over your money (and its security)
  - **-** Full control over your money (and its security)
  - **+** Cannot be seized or stolen if secured properly
  - **-** It's easy to lose 32 bytes if you're careless!
- **Let someone else hold your Bitcoin**
  - **+** May be more diligent about security than you
  - **-** May hold BTC properly but use poor user auth
  - **-** Counterparty risk
  - **-** ~~No~~ Few Bitcoin insurance options

Holding Your Own Bitcoin

- **Holding your own Bitcoin is like a caveman discovering fire**
  - *Can be extremely useful*... and dangerous!
  - Keep your fires small until you are experienced

- **Sometimes the biggest threat to users is themselves**
  - Users are not used to truly irrecoverable data!
  - Not everyone makes backups
  - No one expects their hardware to fail

Educate yourself, learn the tools, learn the risks, and experiment (with small amounts)

Cryptography & Bitcoin

A Short, Non-Technical Introduction

Public-Private Key Crypto

- **On the internet, there are two main concerns:**
  - **Privacy** of communications (encrypt & decrypt)
  - **Authenticity** of communications (sign & verify)

- **Bitcoin protocol does <u>not</u> use encryption**
  - The Bitcoin protocol only uses "authentication"
  - *"Are you authorized to move this money?"*

- **All users create a private key (secret) and a public key (distributed)**

Public and Private Keys

- **Think of Bitcoin as a decentralized, public bank**

  Public Key - is like a **bank account number**

  Private Key - is the **signing authority** on that bank account
  (it is a pen with special ink needed to write and sign checks)

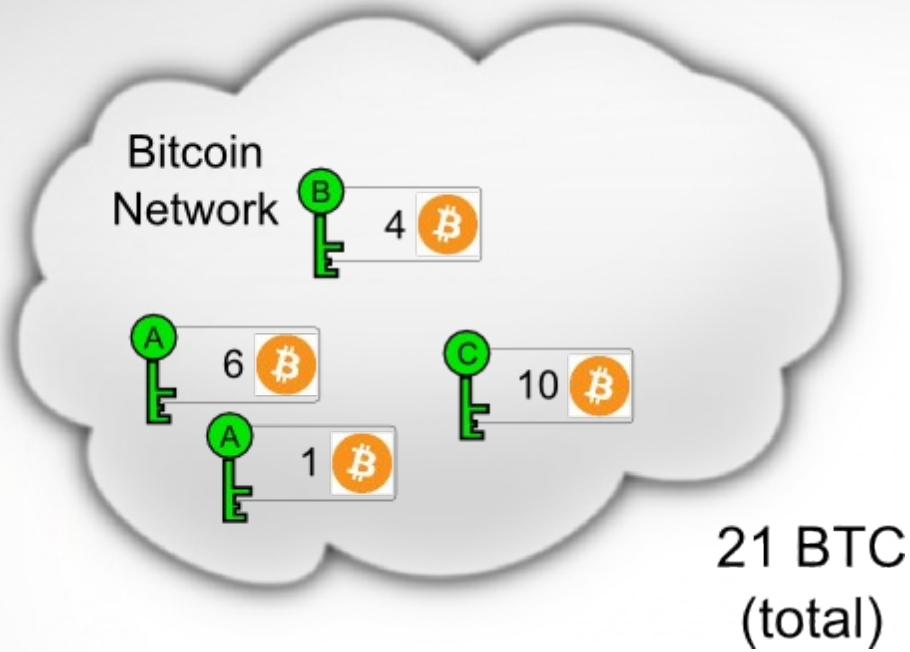
- **All users make keypairs for "account" management**

  Public Key - give to payers to deposit money in your "account"

  Private Key - keep it secret so <u>only **you**</u> can authorize payments

A Bitcoin address is basically just a public key:
Such as: "1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa"

Example Network

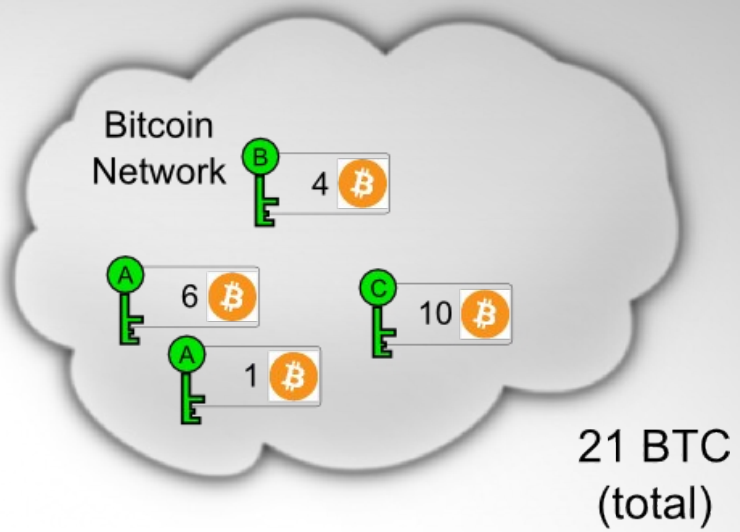


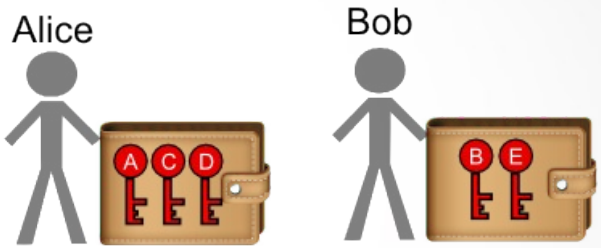
- All Bitcoins have a public "unlock" condition
- Most coins have a simple unlock condition: "**Here's a public key, only a signature from the owner can authorize moving the Bitcoins**"
- If you have the **private key**, you can create those signatures! (so your wallet includes them in your balance)

Illustrated Bitcoin Transaction Demo

Initial Conditions

- Assume the Bitcoin network has 21 coins
- All coins are locked using public keys **A**, **B** and **C** (so far)
- Alice and Bob have all the private keys associated with those coins (and a couple extra unused keys)

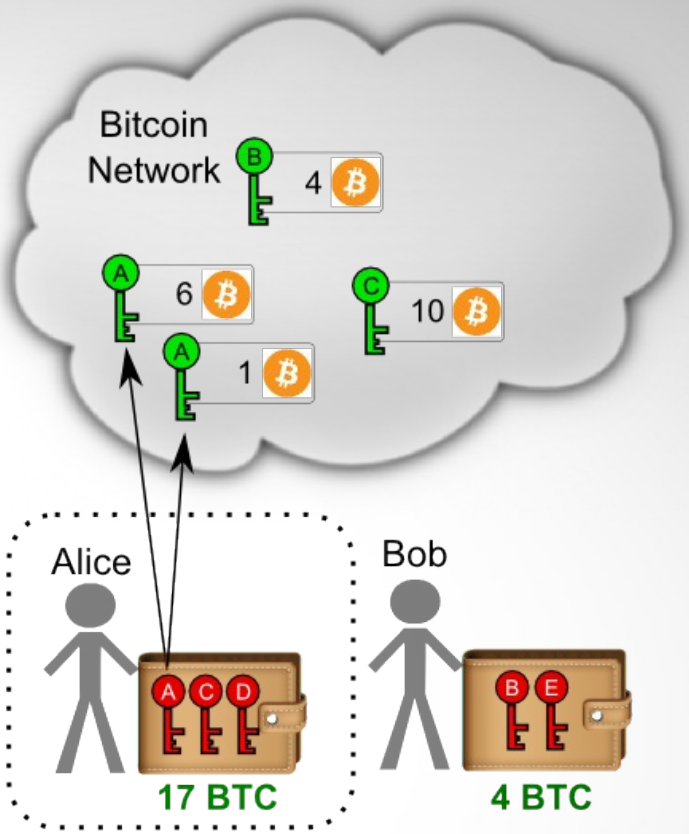




Alice's wallet has 3 private keys (**A**,**C**,**D**)

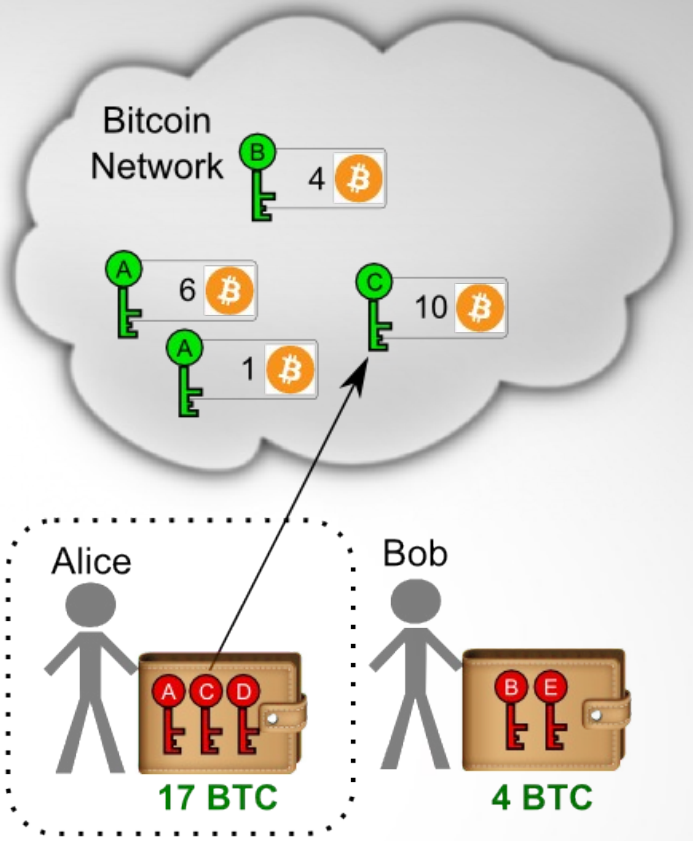
Bob's wallet has 2 private keys (**B**,**E**)

Bitcoin Transactions



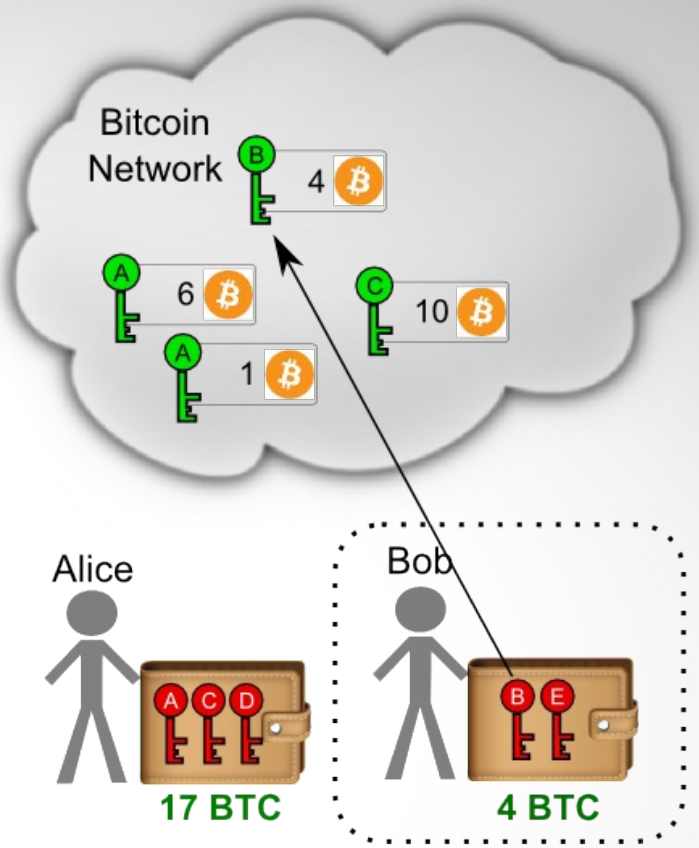
There is **7 BTC** in the network for which **A** can sign (6+1)

Bitcoin Transactions



There is **10 BTC** in the network for which **C** can sign

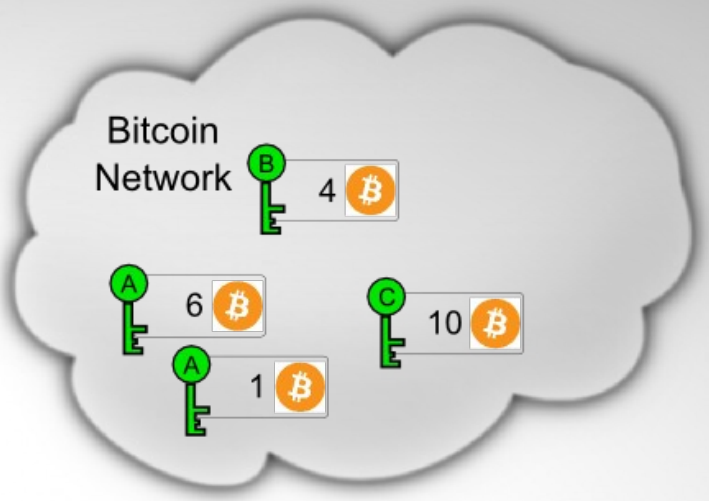
Who owns what?

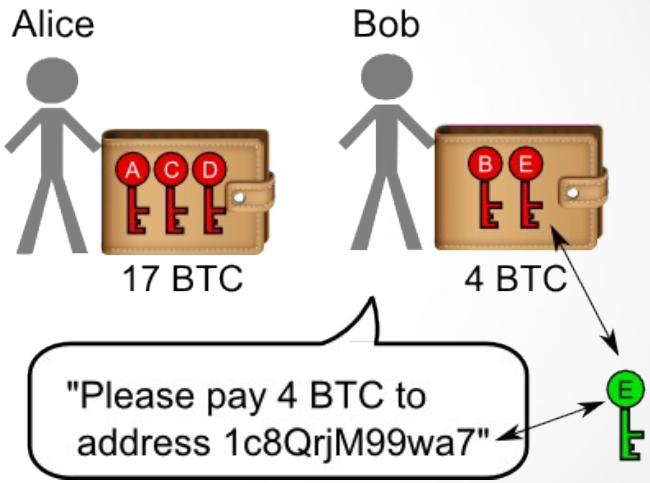


There is **4 BTC** in the network for which **B** can sign

Payment Request

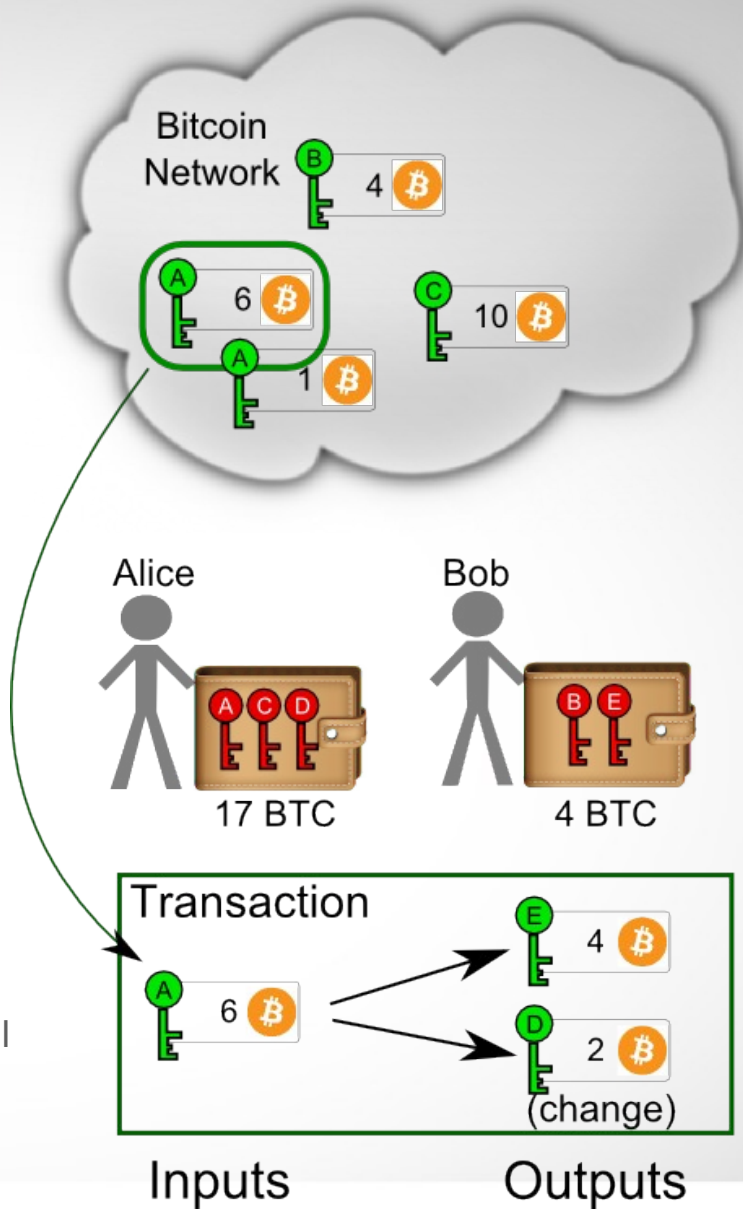
- **Bob** will request **4 BTC** from Alice
- **Bob**'s wallet will select unused private key **E**, and then create a payment address (based on the public key **E**)
- **Bob** sends the address to Alice requesting payment





Create Transaction

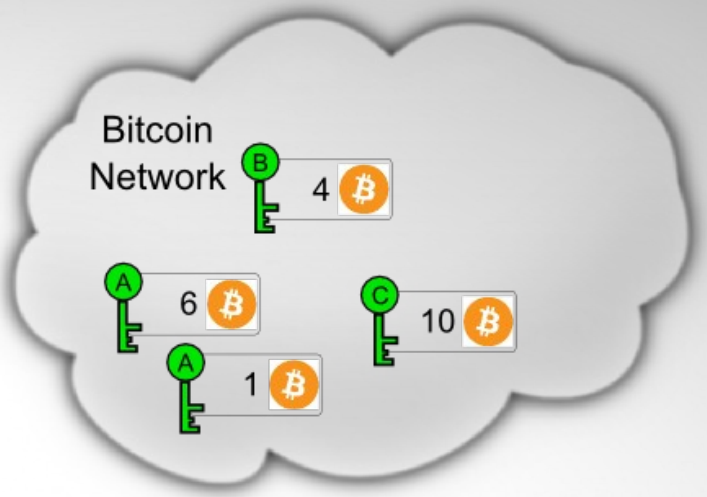
- **Alice**'s wallet selects some coins that she knows she can sign for (at least 4 BTC)
- She will use the 6 BTC associated with **A** (think of it like a $6 bill)
- **Alice** creates a transaction spending the 6 BTC
  - The software selects an unused key (**D**) to send the 2 BTC back to herself (change)

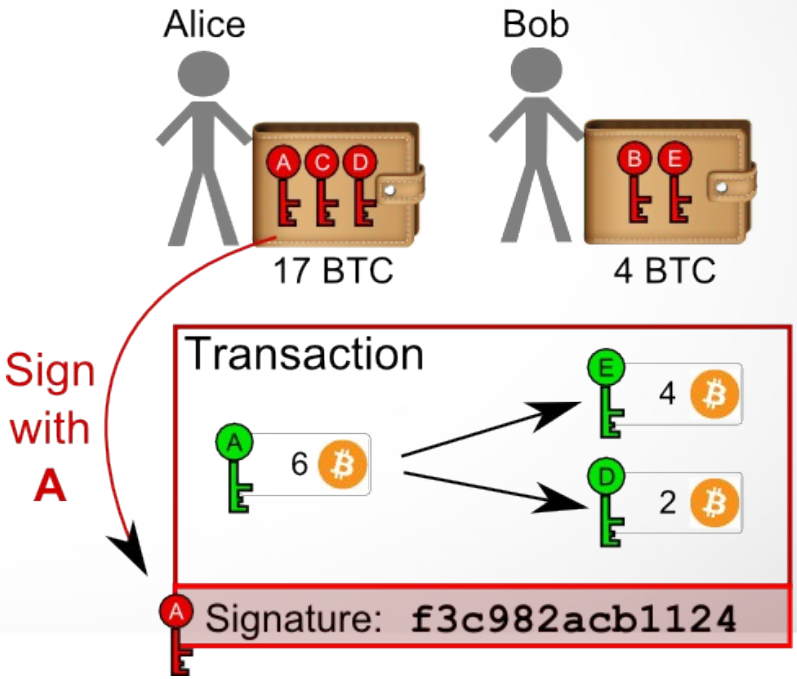


Note: The change-back-to-self process is transparent to the user. All wallet software "hides" those details because they are confusing and irrelevant to most users.

Bitcoin Transactions

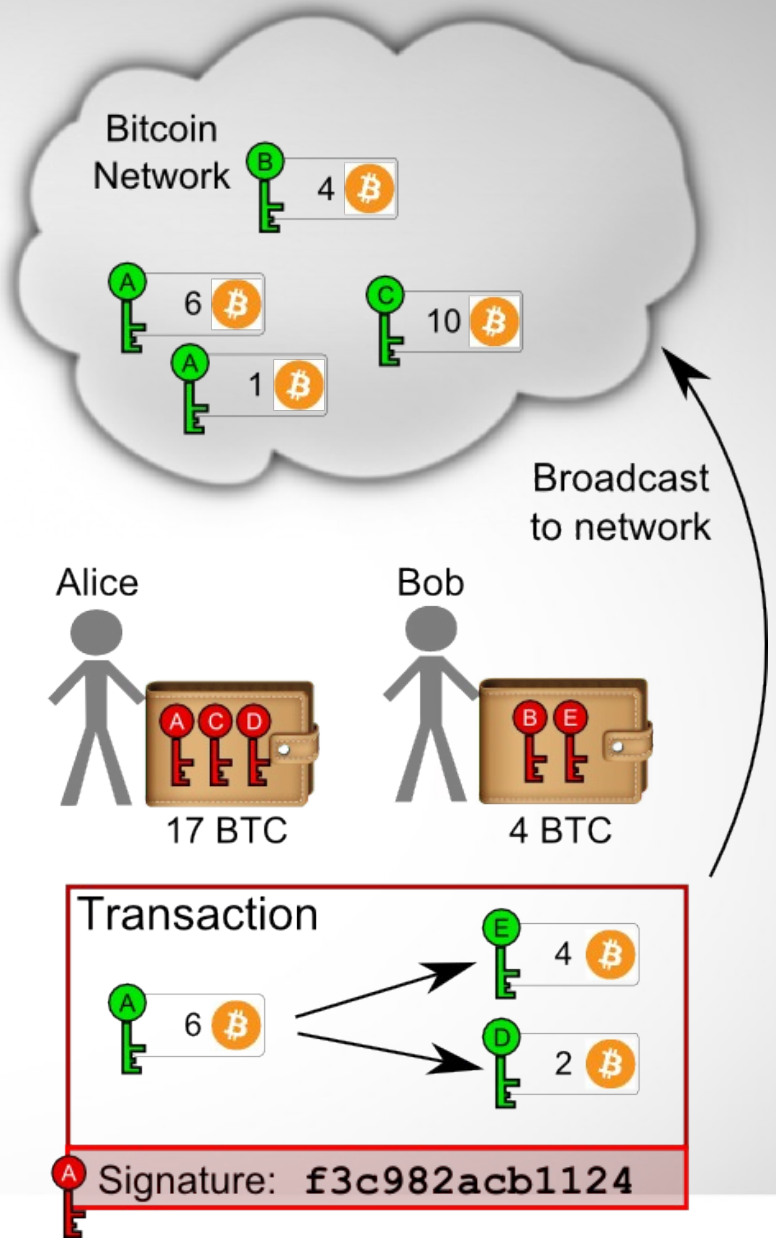
- **Alice** uses private key **A** to sign the transaction
- The signature *is* *mathematically linked* to every detail of the transaction
  - If the transaction changes at all, the signature will break (the math stops working)





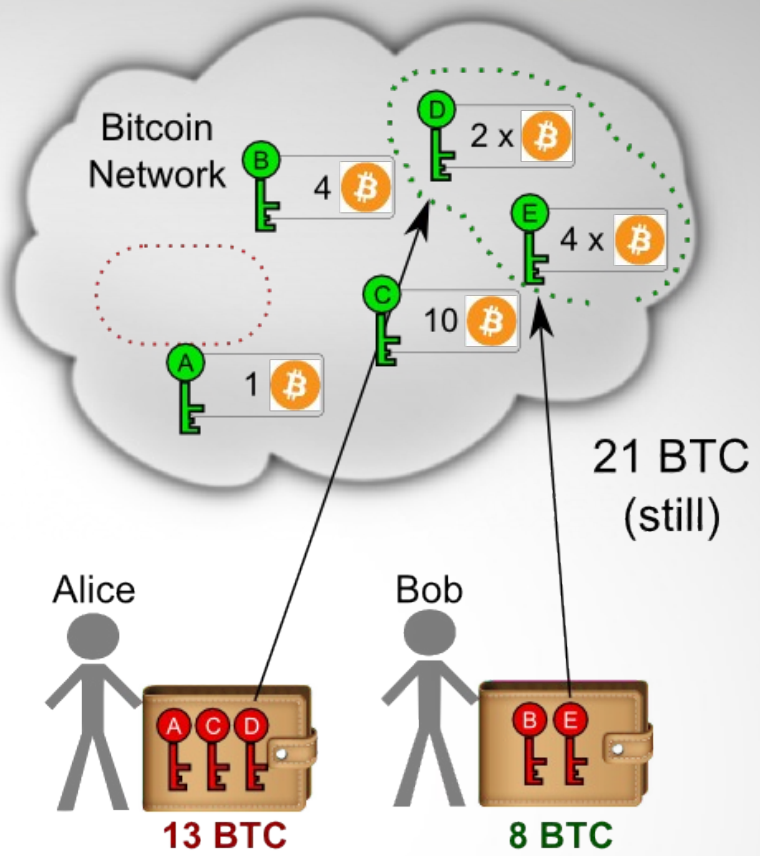
Bitcoin Transactions

- **Alice** "broadcasts" the transaction to the Bitcoin network
- Users of the network verify:
  - The 6 BTC is unspent
  - The sig corresponds to public key **A**
  - The sig is valid for this particular transaction



Final Condition

- The original 6-BTC bill is destroyed and 2 new bills totaling 6 BTC created
- All users update their databases



6 BTC consumed

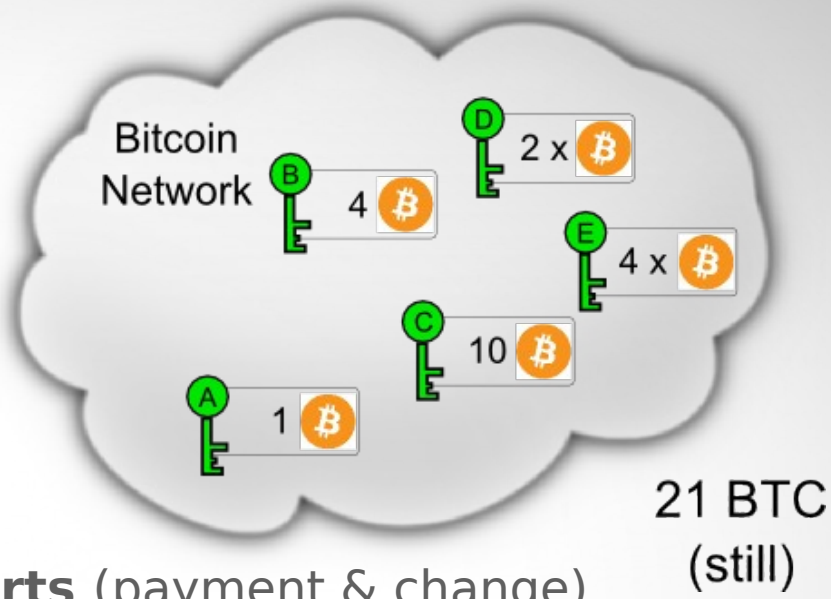
6 BTC emitted w/ new public keys

4 BTC to Bob (E)

2 BTC change back to Alice (D)

What Have We Learned?

- **Private keys let you "unlock" and "re-lock" coins under other public keys** (i.e. send to others)



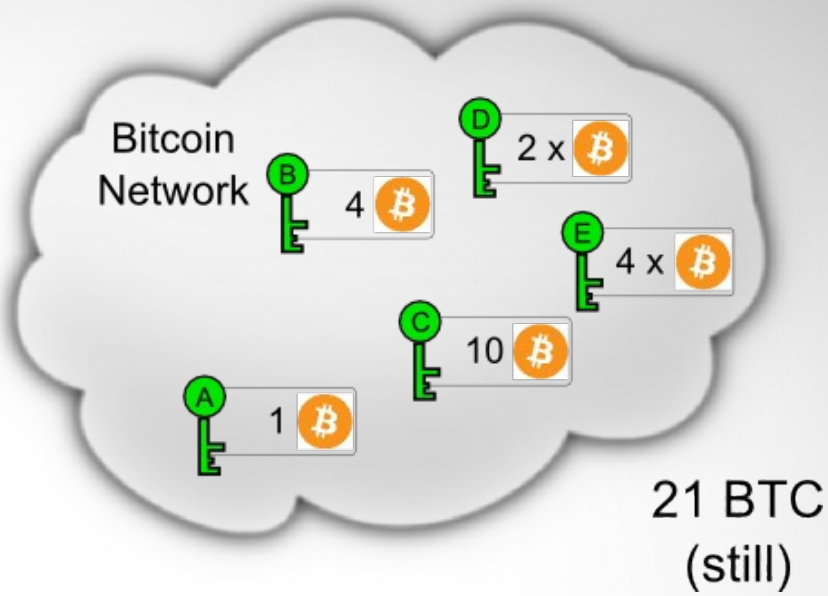
- **Public keys let you:**
  - Receive money
  - See available coins/balance

- **Transactions usually have 2 parts** (payment & change)
  - The change is handled automatically and transparently

- **Signatures are mathematically linked to the signed data**
  - Handwritten sigs can be (maliciously) transplanted between documents. Bitcoin signatures cannot.
- **Wallets contain a lot of private keys**
  - Uses a new key for every receiving operations

Cold Storage

- Now we can appreciate "**cold storage**" (aka "offline wallets")
  - Only **public keys** are required to receive payments and verify transactions
  - **Private keys** are only required to move the coins



It is possible to keep the private keys on an offline computer/device and receive money to it using the public keys

Armory was designed to do exactly this!
It is the gold standard of wallet security best practices

Some Security Best Practices

(for varying levels of paranoia)

Security vs. Convenience

- **Nearly all system become more inconvenient as you increase security!**
  - The easiest systems are usually the least secure!
  - To do security right, expect to be inconvenienced

- **A lot of users don't have the patience for this**
  - Honestly, this is why Bitcoin may not be ready for primetime!

If you're going to hold a lot of Bitcoin it's worth sacrificing some convenience to protect yourself

Backup Your Wallets

- **Risks:**
  - **The most common reason users lose coins is due to not having an unencrypted backup!**
    - You lose all your Bitcoins if your hard drive fails
    - You lose all your Bitcoins if you forget your password
    - Your family cannot inherit your money if you get hit by a bus
  - **It is critical your backup be unencrypted!**
    - An encrypted backup is useless if you forget the password
    - An encrypted backup is useless if your family would like to inherit your fortune

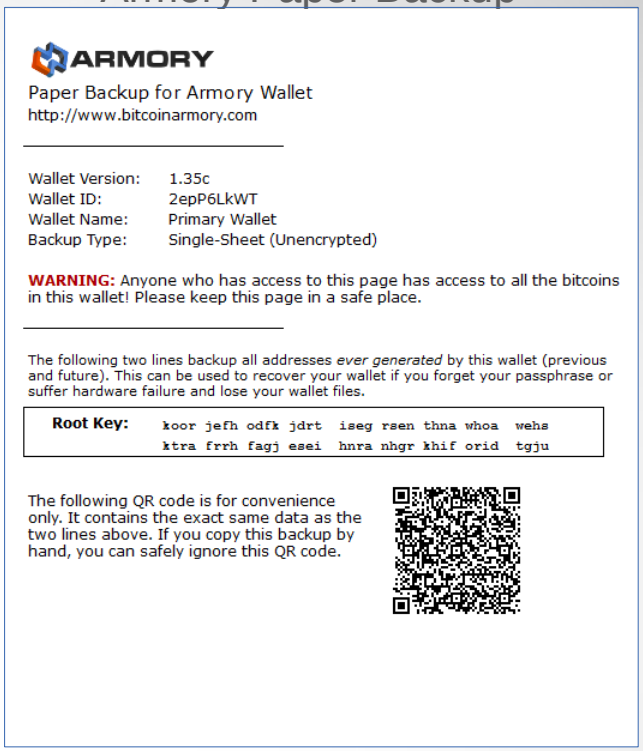
For most users, digital security is most important
For most users, physical security is not a concern

THEREFORE: Make an unencrypted backup secure it!
(paper, DVD or USB key)

Backups: Digital vs. Paper

- **How much are you willing to bet that your CD or USB key will still work in 5-10 years from now?**
- **If you use digital backups, make multiple copies**
  - Store together, at least one should still work
- **Paper *fades* over time, but the data <u>will</u> be recoverable in 100+ years**
  - Most things that destroy paper also destroy digital

Armory Paper Backup



ARMORY

Paper Backup for Armory Wallet
http://www.bitcoinarmory.com

Wallet Version: 1.35c
Wallet ID: 2epP6LkWT
Wallet Name: Primary Wallet
Backup Type: Single-Sheet (Unencrypted)

WARNING: Anyone who has access to this page has access to all the bitcoins in this wallet! Please keep this page in a safe place.

The following two lines backup all addresses *ever generated* by this wallet (previous and future). This can be used to recover your wallet if you forget your passphrase or suffer hardware failure and lose your wallet files.

Root Key: koor jefh odfk jdrt iseg rsen thna whoa wehs
ktra frrh fagj esei hnra nhgr khif orid tgju

The following QR code is for convenience only. It contains the exact same data as the two lines above. If you copy this backup by hand, you can safely ignore this QR code.

Armory Technologies recommends that you use <u>paper</u> backups whenever possible

Copy the data by hand if necessary

Backup Frequency

- **IMPORTANT:** **At the time of this writing (Apr 2014):**
  - ***Bitcoin-Qt*** wallets must be backed up every 100 transactions
  - ***Multibit*** and ***Bitcoin Wallet for Android*** require regular backups unless you always reuse addresses (not good practice!)
    - Address reuse is bad, but probably better than losing money
  - ***Armory*** and ***Electrum*** wallets only require one backup, ever
    - Infinite private keys generated from a single private seed
    - Your paper backup contains the seed in 1-2 lines of letters

If you use Armory or Electrum, make a paper backup, one time, then never worry again!

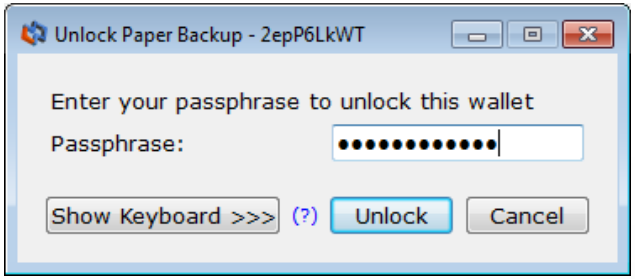
In the next few months, all wallet developers will be implementing the one-time-only backup features

Wallet Passwords

- **IMPORTANT: Your wallet password is your encryption key for your wallet!**
- **If you forget your password, your wallet will be permanently encrypted and your coins will be lost!**

  ...unless you have an unencrypted backup

- **No really: I'm serious your coins will be lost forever**
  - Users are not used to the idea of truly, irrecoverable data
  - Make an unencrypted backup!



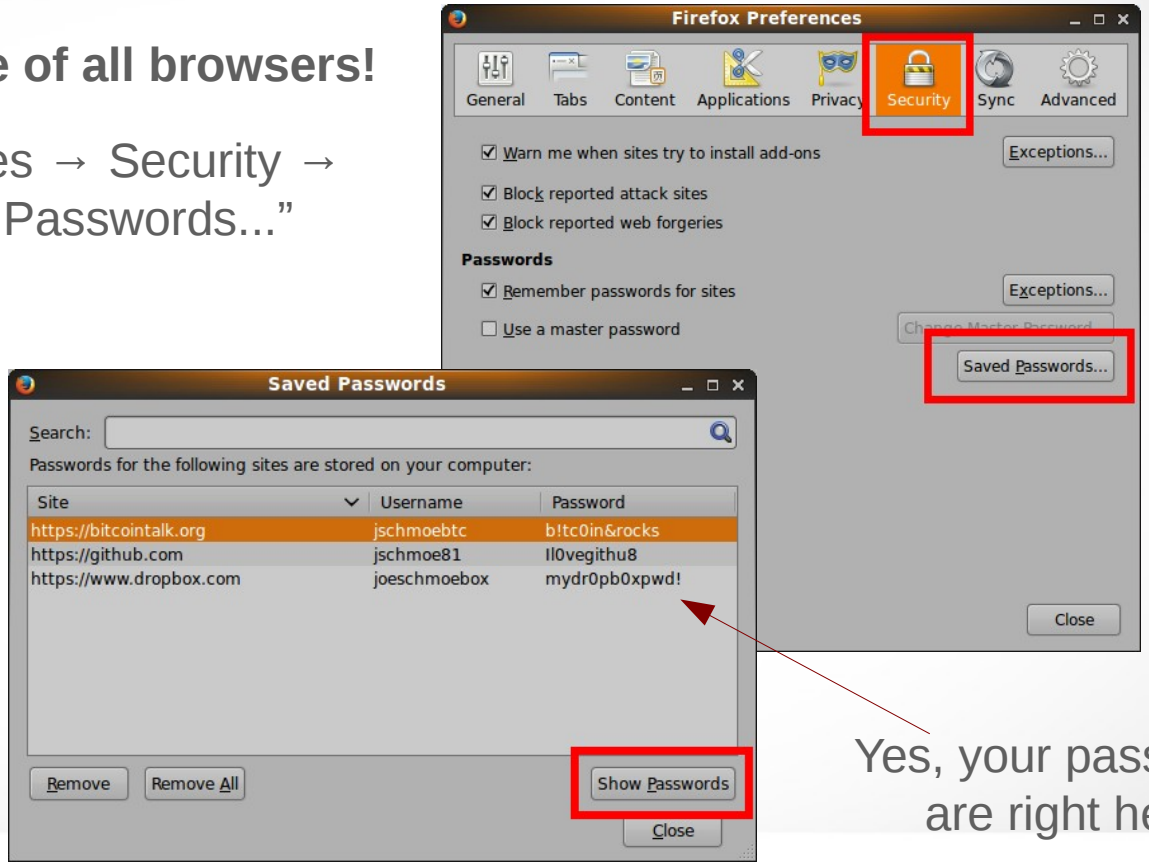
If you've ever forgotten a password, make an unencrypted backup!

Wallet Passwords

- **IMPORTANT: Did you know that all your website passwords saved in your browser are <u>really</u> saved, <u>unprotected</u> on your hard drive?**

This is true of all browsers!

Preferences → Security → "Saved Passwords..."



Yes, your passwords are right here!

Wallet Passwords

- **Try it for yourself!**
  - Preferences or Settings → Security → Saved Passwords
  - Kind of weird staring at your passwords in plaintext, huh?
- **Malware has it easy for most users:**
  - Read password databases for all web browsers
  - Try all passwords on each encrypted wallet found
  - Send all Bitcoins to the attacker's wallet
- **Anyone with physical or remote access can do this, too!**

Do not use a wallet password that is the same or similar to any of your website passwords!

Password Length

- **Your password is your encryption key: use a good one!**
  - If your wallet is valuable enough, a botnet of 1,000,000 computers may be used to try to brute-force the password
- **Brute-forcing is exponentially harder with more letters**
  - A password that is easy to remember, is easy to brute force!
  - **Make one that's hard to remember, make unencrypted backup!**
- **Consider using words for your password!**
  - Tend to be easier to remember, have lots of entropy
  - But still make them random!

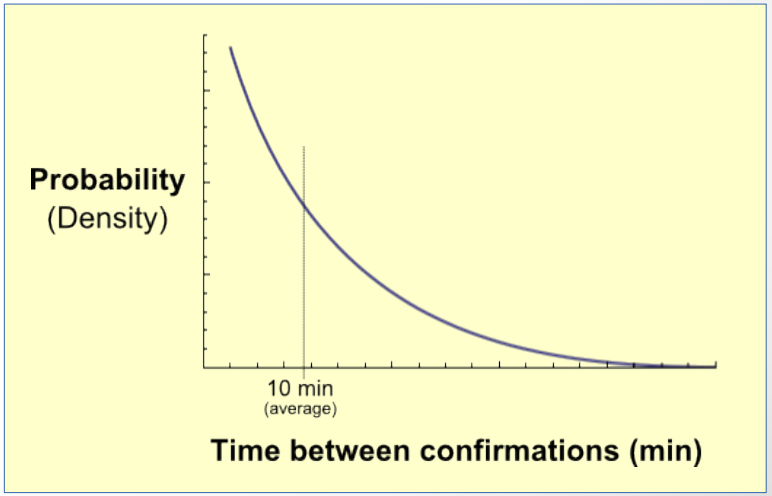
Use at least 14 <u>random</u> letters or 6 <u>random</u> words

Then make an unencrypted backup!

What are Confirmations?

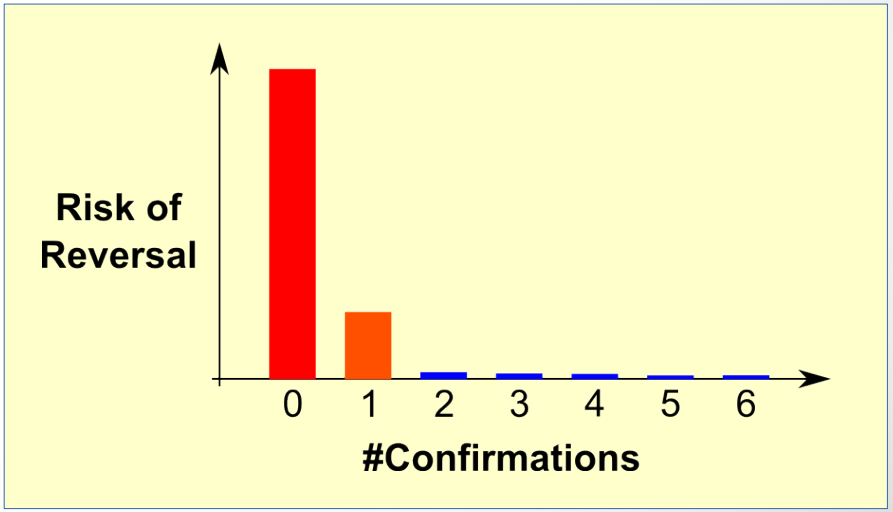
- **Bitcoin transactions are <u>not</u> instantaneous**
- **Each confirmation is increased consensus that the transaction actually happened**
  - The first confirmation is the most important
  - <u>Six</u> confirmations is generally considered irreversible
  - For $1,000,000+, wait 20-30 confirmations

- **Confirmations come <u>on average</u> every 10 minutes**
  - Actually random: usually **10 sec** to **45 min**



Confirmation Risks

- **Do not trust zero-confirmation transactions unless there is pre-existing trust!**
  - Or, you're willing to eat the loss when reversed
- **Attacks on zero-confirmation tx are easy and cheap**
  - Just not that many people doing it right now
- **Attacks on one-confirmation tx require a bit more resources**
  - But possible!
  - **Two is a good number for non-critical transactions**



Call-to-Verify Addresses

- **If you are sending <u>large</u> amounts of Bitcoin:**
  - You want to make sure you send it to the right place!
  - An attacker could replace the correct address with his own on its way to your wallet software

- **This is a serious security issue!**
  - The “payment protocol” hopes to solve this by using SSL concepts to prevent address tampering
  - This will not work in all environments (not everyone has an SSL certificate)

- **Pick up the phone and call the other parties**
  - Make sure they are who you think they are!
  - Manually verify the address before execution
  - <u>This is much more reliable with an offline computer</u>

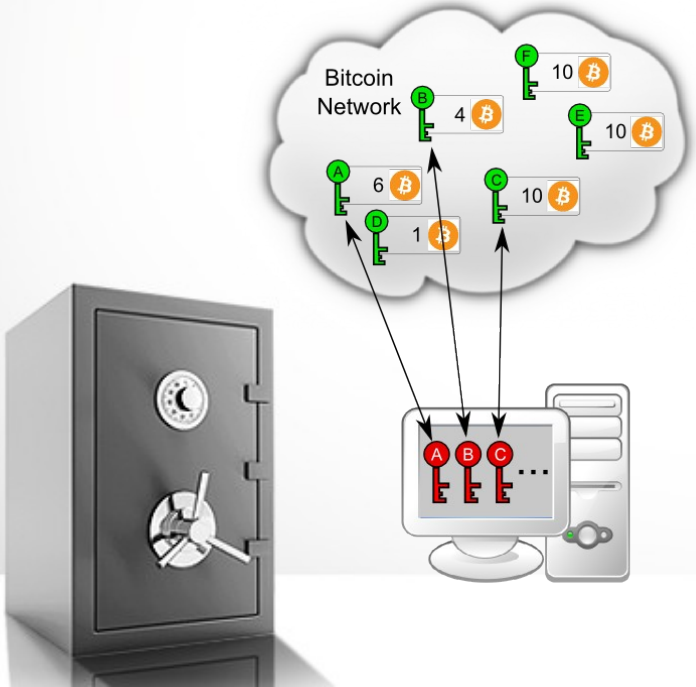
Cold Storage

and the Holy Grail

Hot vs. Cold

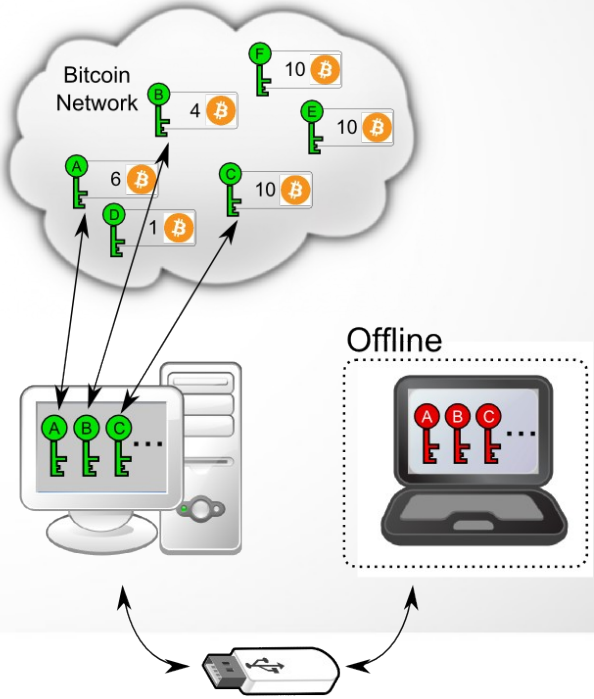
“Hot” Wallet

- The private keys are on an internet-attached system
- All wallets are “hot” by default



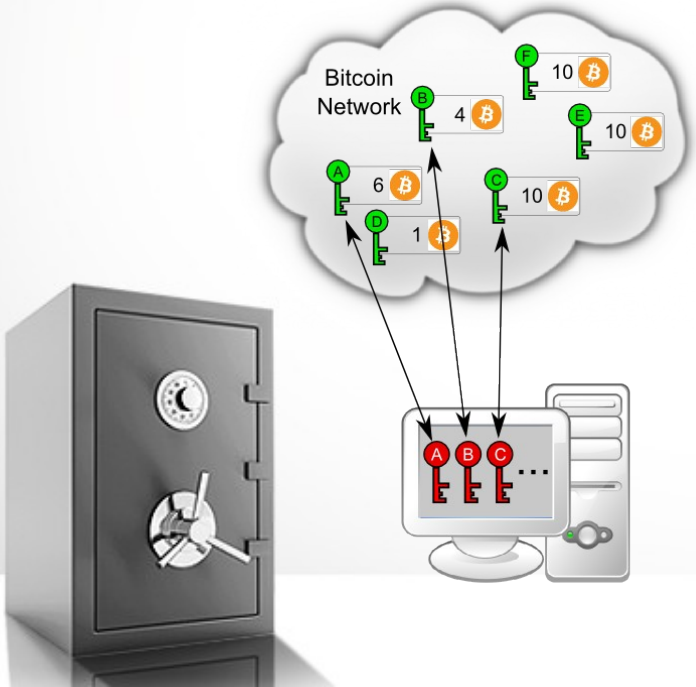
“Cold” wallet (“offline wallet”)

- Gold standard of security
- Private keys created and never leave the offline computer
- Transactions are signed offline

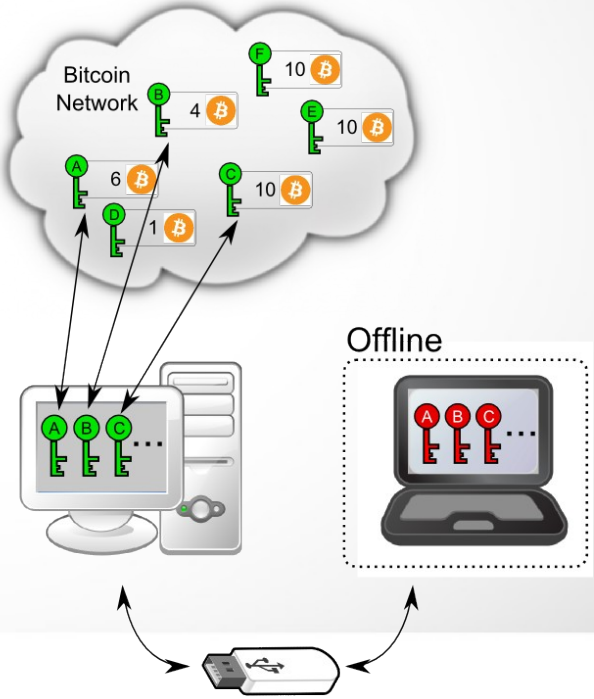


Hot vs. Cold Security

- **All <u>known</u>, <u>major</u> Bitcoin breaches to date:**
  - Coins stored on a hot wallet
  - Or unencrypted backups stored on an “hot” computer



- **Compromising a cold wallet requires one of the following:**
  - Physical access
  - Extremely advanced USB viruses
  - User accidentally installing malicious software



Setting up the Offline Computer

Online computer	Offline computer
• (1) Install Armory	• (1) Install Armory • (2) Create new wallet • (3) Create paper backup – Copy by hand, if necessary • (4) Create "watching-only" copy of wallet • (5) Copy to USB drive
• (6) Import "watching-only" wallet	

Your "watching-only" wallet has only public keys, no private keys!

Doing an Offline Transaction

Online computer

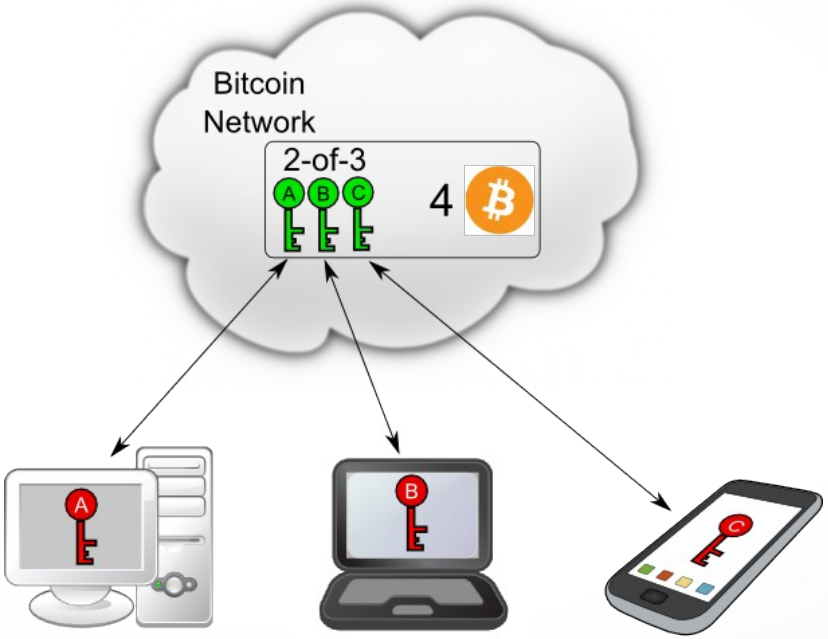
- **(1) Create transaction**
  - Same as you would with a hot wallet
- **(2) Save unsigned transaction to USB**

Offline computer

- **(3) Load tx from USB**
- **(4) Review for accuracy!**
  - All benefit is lost if you don't review on the clean, offline computer
- **(5) Sign the transaction, save to USB**

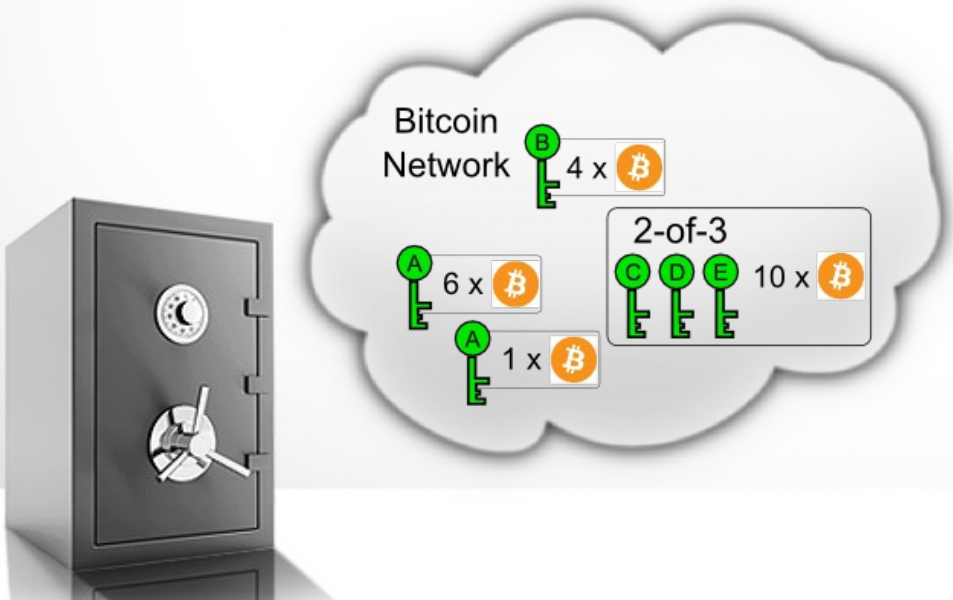
- **(6) Load signed transaction, broadcast to network**

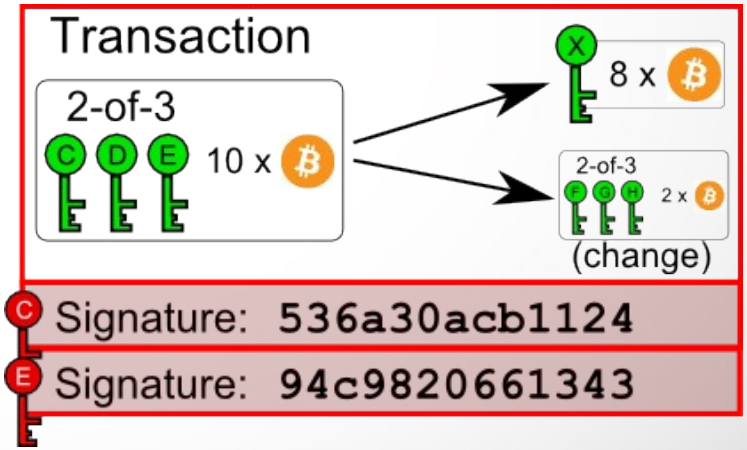
Multi-Signature Transactions



Multi-Signature Transactions

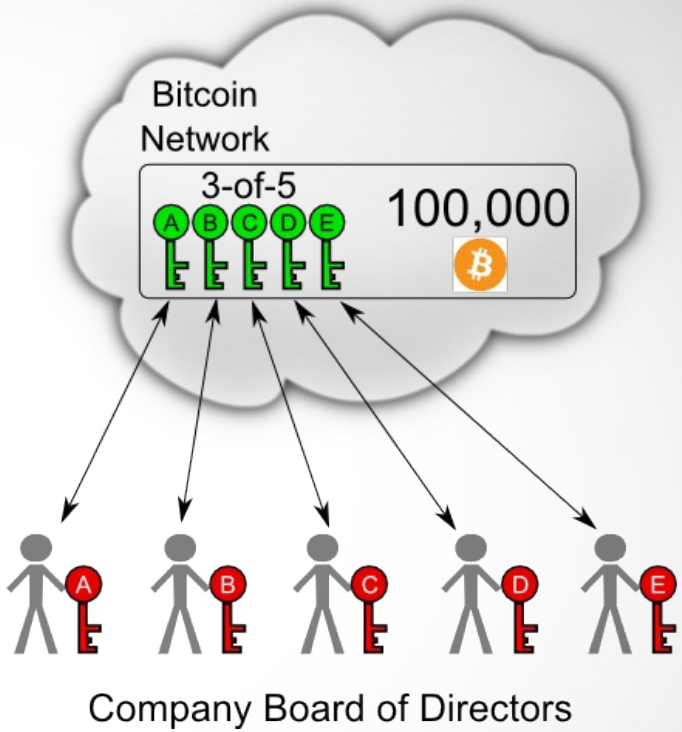
- **Most coins have a simple unlock condition:**
  - Here's a **public** key, sign with its **private** key to move
- **Much more complex conditions are possible:**
  - Here's **3** **public** keys, sign with any **2** **private** keys
  - This is a **2-of-3** multi-signature transaction





A Critical Puzzle Piece

- **Multi-signature transactions are critical for large organizations**
  - Wallets are managed by employees, who may steal
  - All wallets currently have a single point of failure
    - **You can have:**
      - Five board members of a company create wallets
      - All money handled by the company goes into 3-of-5
      - All transactions requires 3 signatures to be moved

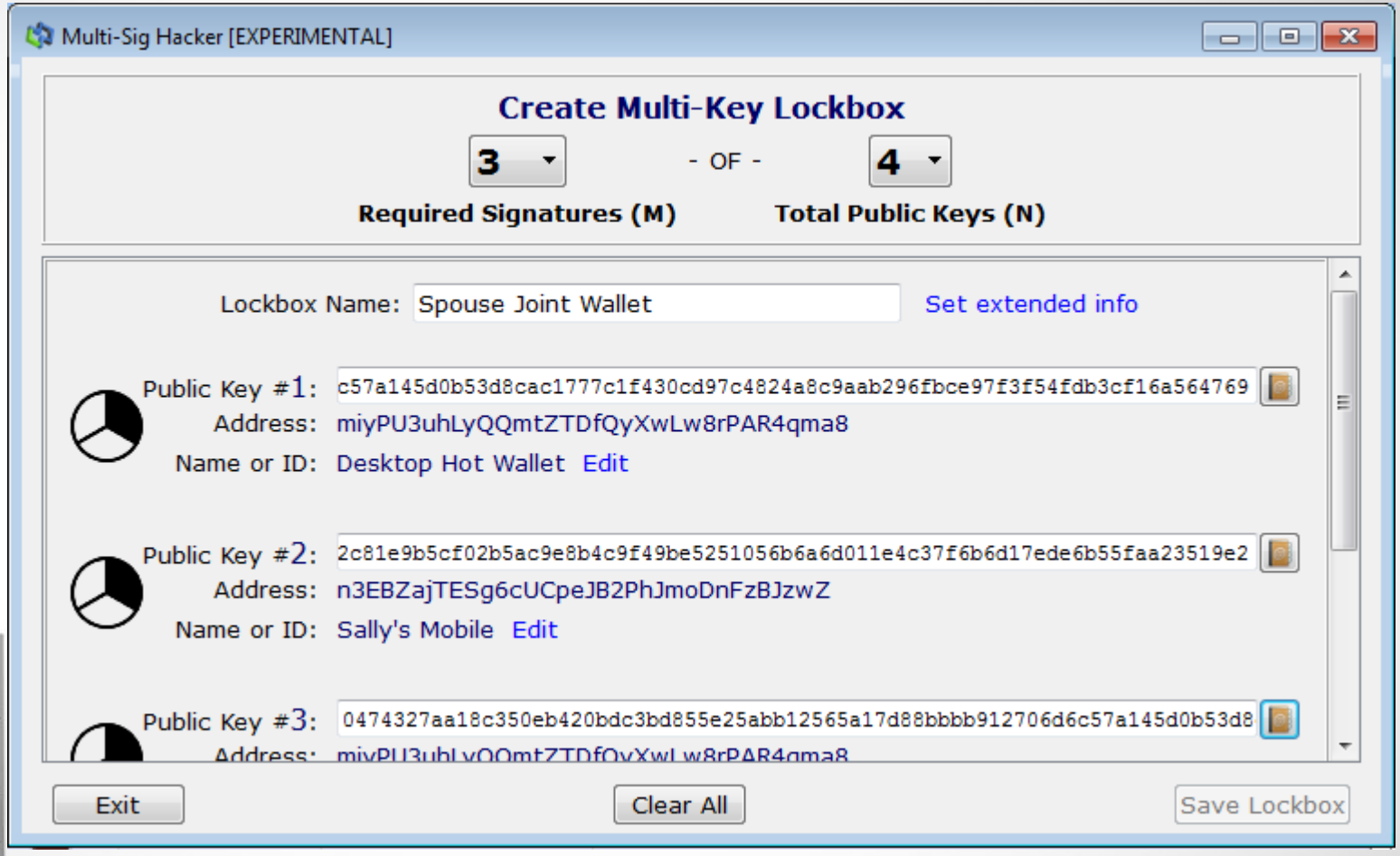


- **The Bitcoin network supports any M-of-N up to 20-of-20 !**

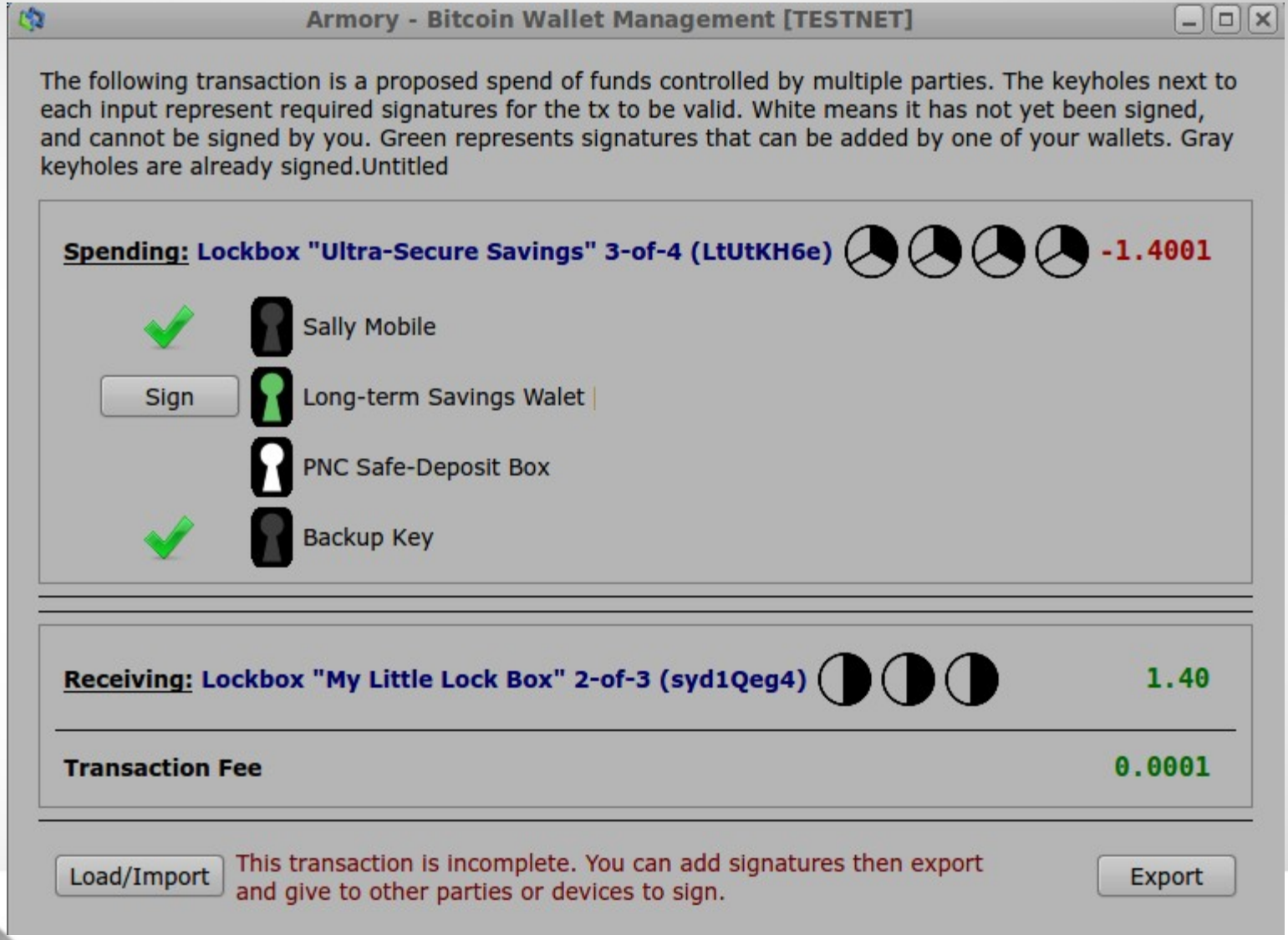
Armory “Lockboxes”

- **Armory just unveiled a multi-sig interface**
  - Collect public keys to create **lockboxes**
  - Deposit money in lockboxes like any other address
  - To spend from a lockbox:
    - Create a transaction
    - Each other party signs it
    - Last party broadcasts it (finalizes it)
- **Multi-sig transactions are inherently complex!**
  - Armory has made them about as easy as possible…
  - …just like it did with cold storage!

Collect Public Keys



Review and Multi-sign



Multi-Signature Services

- **Many other multi-signature services have popped up, such as BitGo, CryptoCorp, Xapo, Ciphrex**
- **The most common service is the 2-of-3 with the service providing two-factor authentication**
  - Service generates one key
  - User generates a hot key kept on the computer
  - User generates a cold key kept offline (backup)
    - **To spend money:**
      - User's software generates a transaction, adds one sig
      - Partially-signed transaction sent to service
      - Service sends text message or calls phone to confirm
      - Service adds second signature and broadcasts
    - **If the service goes out of business, the user still has two keys and can recover the funds.**

The Future

(coming soon)

Consumer Hardware Wallets

- A great tradeoff for security and convenience
- Hardware wallets hold the **private** keys and **sign on the device**
  - The **private** keys cannot be read from it
  - It will only emit the **public** keys

Trezor Hardware Wallet

- The **Trezor** is the most anticipated HW wallet
- Shipping soon!

Enterprise Hardware

- **Hardware Security Modules (HSMs)**
- **Many levels of FIPS 140-2 HSM certification:**
  - **Lowest:** simple verification of secure storage and computation on standard HW
  - **Highest:** security-hardened HW, physical and electronic tamper-resistance, self-destructing, $25,000+
    - **Used for both secure storage and cryptographic acceleration**
    - **Only a few major manufacturers**
      - Ultra Electronics AEP
      - SafeNet
      - Thales
      - Ultimaco

Armory Technologies, Inc.
8160 Maple Lawn Blvd, Ste 200
Fulton, MD 20759
https://bitcoinarmory.com

enterprise@bitcoinarmory.com

Twitter: @armory

Extras

Other Useful Stuff

GPG-Verify Your Installers

- **Risks:**
  - You download installers from a malicious website
  - An attacker tampered with the installers on the real website
  - The installers are replaced during or after download

- **Mitigation:**
  - All wallet devs sign their installers using known GPG keys
    - Most devs keep a special offline GPG **private key** just for this!
  - Get the developer's GPG **public key** and verify!

If the installer has a valid signature from the correct GPG key, it does not matter where you got it from!

There are slides at the end that explain in detail

This is very easy in Linux & Mac, but a lot of work in Windows!

Verify Your Installers

- **GPG is a powerful, thoroughly-trusted crypto tool**
  - Presintalled in Linux & Mac; takes effort in Windows
  - Because it's hard in Windows, I verify my Windows installers in Linux

Steps

- **(1) Get GPG and file-hashing tool**
  - Linux & Mac: Do nothing, it's all pre-installed!
  - Windows: Download **gpg4win** and SHA256 file hash tool (**HashCalc** is good)
- **(2) Import the GPG keys to your keyring (only done once)**
  - Most tools have search & import function (Linux & Mac: "`gpg --recv-keys <keyID>`"
  - Each developer's "keyID" should be well-known: mine is 98832223
- **(3) Download the installers and signed hash files***
  - Hash the installer file (Linux & Mac: "`sha256sum <filename>`")
  - The result looks something like this: `f98c7a798122167c98c0a798122167f9030a7`
  - Compare to the hashes in the signed file
- **(4) Verify the signature on the hashes file**
  - Win: use right-click gpg4win menu; Linux & Mac: "`gpg -v sha256hashes.txt.asc`"
  - **MAKE SURE THE FINGERPRINT MATCHES THE EXPECTED KEY**
  - Anyone can create a "valid" signature – but not from the developer's key!

*If it is a .deb installer (Linux), it may be signed directly, only need "`dpkg-sig --verify *.deb`"

Do Not Reuse Addresses

- **Risks:**
  - Bitcoin is actually not very good at anonymity
  - When you reuse addresses you make it **far** worse
  - Reusing addresses can hurt other users' privacy as well

- **Mitigation:**
  - **Bitcoin-Qt, Armory** and **Electrum** do not reuse addresses by default
  - Some users force reuse due to lack of understanding or simplicity of backups

  - **Multibit** & **Android Bitcoin Wallet** reuse addresses by default
  - Usually have an option to explicitly create new addresses, but not default

If you are using Bitcoin-Qt, Multibit or Android Bitcoin Wallet, you may want to reuse addresses anyway if you do not create backups regularly.

(lack of privacy is usually preferred to losing coins)

Address Reuse & Privacy

- **Discussion:**
  - Address reuse is mostly a privacy issue, not a security issue
  - Reusing the same public-private keypair is expected & safe throughout the rest of internet security
  - But it is egregiously bad for privacy in Bitcoin

- **There are contexts in which it is okay, but not standard**
  - Donation addresses: all users donating know it is heavily reused, and accept being linked to it

Users do not realize just how much privacy information is leaked by interacting with heavily-reused addresses!

Doing it Right

- **If you are running any kind of online Bitcoin business, offline-wallets are an invaluable tool**
  - **Keep bulk of your funds in an offline computer**
    - You can even keep it in a safe-deposit box!
  - **All webservers and on-site computers *should only use watching-only wallets!***
    - Securely collect payments to the offline wallet
    - Track your wallet balance
    - Track and verify all payments/transactions
    - **No one who gains access to the server can steal it!**
      - Includes employees

If you need a hot wallet, keep it small, periodically refill from the cold wallet

Use Linux

- **Once you go down the “cold storage” path you are implementing serious security**
- **As of this writing, the best way to move data between online & offline computer is USB drives**
  - Linux has a much better history of resisting USB-based attacks
  - We are working on better methods for secure transfer

- **Armory website has Ubuntu “Offline Bundles”**
  - Will install and run on the first boot of a fresh install of Ubuntu 10.04 or 12.04
  - The offline computer needs no other software at all!

Extra Credit

- **Dedicate a small USB key for offline transactions**
  - Minimize exposure to potential viruses
- **Dedicate a computer for the creating transactions**
  - Minimize exposure to potential viruses
  - Make it exclusive for Bitcoin processing
- **Use full-disk encryption to protect privacy**
  - Without it, someone not authorized can still see the wallet value and transaction history
  - Also adds an extra layer of security

Did I mention, make unencrypted backups?

Armory “Lockboxes”

- **The lockbox interface is the first step towards a more user-friendly version**
  - Decentralized: no third-party services
  - All data can be exchanged via email, chat, USB
- **Armory (and others) will create server-assisted version that handles most complexity for you**
  - Create a spending transaction from a 2-of-2
  - Other party or device gets notification, confirms

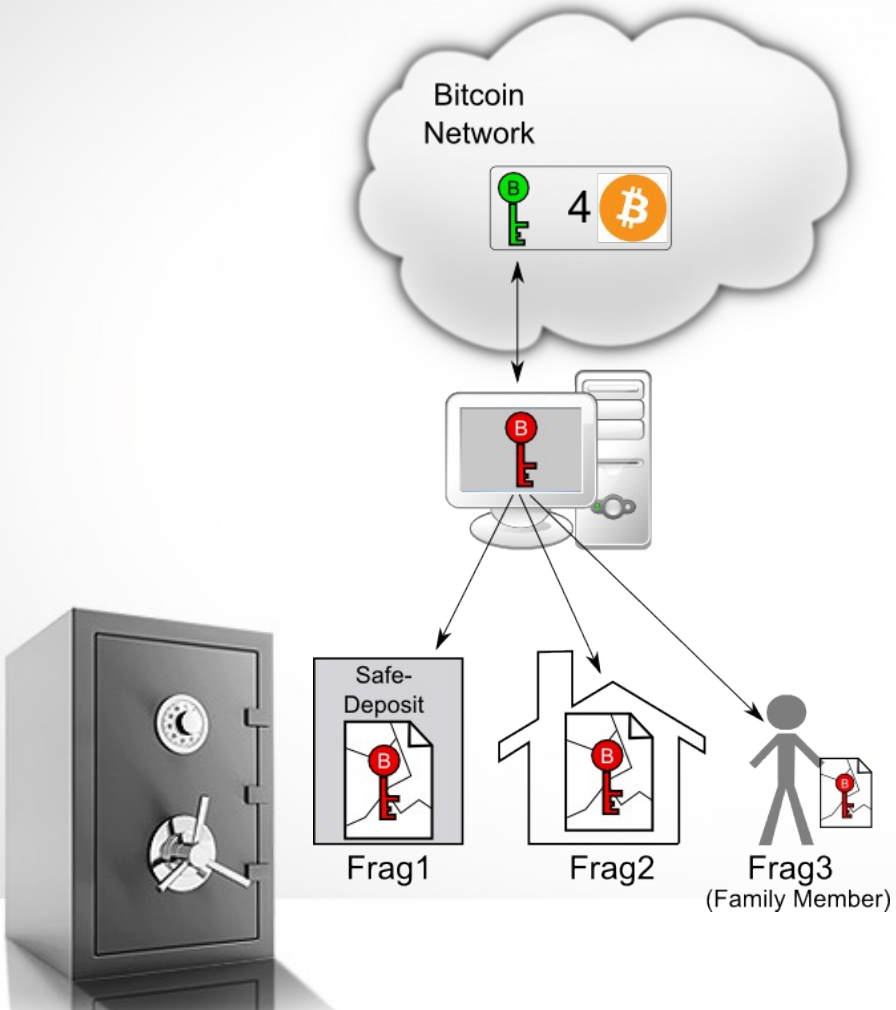
GPG Keys of Major Wallets

- The most sensitive part of using GPG keys is the fingerprint distribution
- So here they are! (most GPG apps only show last 8 chars)

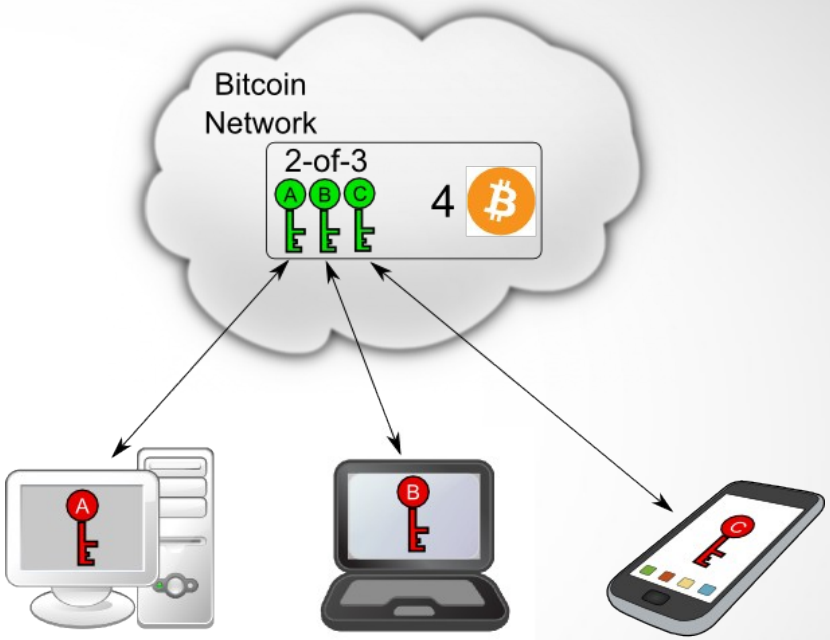
Wallet	Core Developer	GPG Fingerprint
Bitcoin-Qt	Gavin Andresen	29d9ee6b **1fc730c1**
Armory	Alan Reiner	4ab16aea **98832223**
Multibit	Jim Burton	c1972aed **79f7c572**
Electrum	ThomasV	2bd5824b **7f9470e6**
Bitcoin Wallet for Android	Andreas Schildbach	ca662be1 **8b877a60**

Multi-Sig vs. Fragmenting

M-of-N Backups



M-of-N Multi-Sig



Splitting roles

Online computer

- **The watching-only wallet is identical to a regular wallet, but cannot sign/spend**
- **An attacker getting the online wallet is a breach of privacy, not security**

Offline computer

- **Offline computer cannot display balances**
- **Remember, the offline wallet is the signing authority.**
- **The offline computer is a pen with specially-identifiable ink, for writing and signing checks**
  - The pen doesn't know or care what it's signing – it's up to you to verify what you're signing

Multi-Sig vs. Fragmenting

M-of-N Backups

Fragmented backups are for securing your backup

All transactions still require a single signature, from a single computer

The fragments only need to be collected if wallet is lost

M-of-N Multi-Sig

Multi-signature transactions are **network-enforced**

Multiple public keys are included **in the unlock conditions** of the coins

Network expects multiple sigs for every transaction

Brainwallets (don't use them!)

- Humans are really bad at memorizing things
- You will lose coins
- Your family will never recover your coins if you die
  - You literally take your wealth with you to your grave
- Any system that requires your brain to be useful is essentially a brainwallet
- This is why Armory hates encrypted backups:
  - If all your wallets are encrypted
  - And all your backups are encrypted
  - You have a brainwallet!

Segregate Funds by Security

- **Risks:**
  - Having all your funds in a single wallet, means all funds have the same security
  - Usually means funds are super-secure-but-inconvenient, or not properly secured

- **Mitigation:**
  - Use multiple wallets (Armory & Multibit have native support)
  - Exercise all the best practices on the majority of your funds
  - Keeps most of your funds secured, periodically refill low-security wallets

Sweep vs. Import

- **Definitions:**
  - "**Sweeping**" an address/key means sending all the coins owned by that key to a new address (one you control)
  - "**Importing**" an address means to add the **private** key to your wallet – usually so it can be reused

- **When to sweep vs import**
  - Sweep if anyone else has ever had access to the **private** key
  - Importing really only makes sense with address reuse
    - I already told you not to do that!
    - **Serious Security to consider**
      - You import a key that someone else has
      - That person pays you for services/goods
      - They sweep the key after you have delivered

When in doubt, SWEEP

Third-Party Risk

- **The history of Bitcoin is filled with users trusting third-parties to hold their money**
  - Most Bitcoin services have no FDIC-equivalent

Date	Service	Service Type	BTC Lost / Stolen	USD Value (at time of loss)
June 2011	Mt. Gox	Exchange	2,000	$47,000
June 2011	MyBitcoin	Wallet	79,000	$1,100,000
May 2012	Bitcoinica (#1)	Exchange	38,000	$91,000
July 2012	Bitcoinica (#2)	Exchange	40,000	$305,000
Sep 2012	Bitfloor	Exchange	24,000	$250,000
Oct 2013	Inputs.io	Wallet	4,100	$1,200,000
Nov 2013	GBL (China)	Exchange	4,100	$4,100,000
Feb 2014	**Mt. Gox**	**Exchange**	**850,000**	**$500,000,000**
Mar 2014	Flexcoin	Wallet	900	$600,000

Holding Your Own (cont)

- **Most users should <u>not</u> be holding life-changing amounts of Bitcoin themselves**
- **Most users should <u>not</u> be trusting third-parties to protect life-changing amounts of BTC for them**
- **Wait... so, what are users supposed to do ?!?**

<u>Answer</u>: Most users should <u>not</u> be putting life-changing amounts of money into Bitcoin yet!

- **Bitcoin is still the Wild West of money**
- **People like me are building safer tools & infrastructure**
  - But we're not done yet