Merkle Tree

1 Bitcoin

1. Best policy: ever spend your bitcoin
2. A currency where the best policy is to never spend
3. If you buy bitcoin you have to wait a few days before you can tell your bank it was a fraud and get your money back
4. Although the transaction itself is cheap, waiting time of ten minutes has real world cost
5. Bitcoin has no tie to dollars so it fluctuates up and down
6. Every actual bitcoin transaction: turning dollars to ditcoing, doing bitcoin, turning to back to dollars
7. Using bitcoin is a political statement and is censorship resistant (drugs)
8. Silk Road was a TOR hidden service marketplace - US centric - selling drugs and accepted bitcoin - Mandatory feedback and escrow system, optional currency hedge for sellers
   (a) The Armory: you could buy guns too - amazingly illegal for everyone - black market prices - 95 percent of people just buy guns legally
9. Ransomeware: you infect a system, encrypt files, encrypt master key with extortionist's public key, tell them to pay money or they won't see there file again - hard to get paid
10. Can't store Bitcoin on an Internet connected device - or you will get robbed - store with a private key offline
11. The ECDSA signature scheme has a nonce (k-value) like r in El Gamal. If you sign two different messages with the same k, it becomes trivial to find the private key
    (a) Green Dot MoneyPak: service that allows you to ransfer money to reload prepaid credit cards for 6 dollars - this used to be preferred by ransomware but not Bitcoin
    (b) Bitcoin uses a lot of energy -more than UC Berkeley
    (c) Bitcoin is pseudonymous - every wallet is a distinct pseudonym and every transaction is public
    (d) Ethereum: "Smart" Contracts in a JavaScript-like Language
        i. Executes a small virtual machine - Paymaent to a destination can invoke the destination's program

ii. In paying someone else, it invokes code outside itself
iii. At the same time, the cryptocurrency community tends to belive code is law - the basis of some very interesting attacks:
   A. Denial of Service (DOS): Find code that is cheap in terms of "gas" but expensive in practice - like dis I/O - spend a bunch of transactions to slow down system and stop the network
   B. Distibuted Autonomous Organazation: like a mutual fund whose investments are the consensus of the participants - including the ability to split off and perform other actions - it tended to be a bit of "natural ponzi" scheme right from the start - nearly 10 percent of all Ethereum was invested in the DAO
   Bug: Attacker could propose a split that he gets all his money - split process would first transfer money the reduce the balance - in transferring the destination is simply calling another function, one being written by the attacker - the attacker can claim he followed the contract (the code) - classic TOCTOU
   This caused the Ethereum community to split in two - one group said "revise history" and simply have the miners ignore the DAO theft - the other group kept the old chain going because they didn't have their money stolen from them

12. Anything electronic must have an undo - detection and mitigation
13. Bitcoin has a limited amount of programmibility - you can pay two an address - inputs are actually small scripts
14. Allowed to set up a structure where in order for some transaction 2 or 3 signuatures must be present

1.1 How to Make Money in Bitcoin in 10 Easy Steps

1. Move to Sochi
2. Break into blockchain.info and other web-wallet services
3. Download saved web wallets for offline cracking
4. Modify wallet service javascript to leak passwords
5. Be patient and wait
6. When discovered, steal all the Bitcoin
7. Blame the victims
8. Write malcode to look for Bitcoin wallets
9. Crack away and rob everyone
10. Enjoy life

2 Key Management

Problem with sending a public key: MITM can change to be his own public key and decry pt any message.

2.1 PKI - Public Key infrastructure

Helps manage public keys and certificates

2.1.1 Trusted Directory

1. **Trusted Directory** contains mapping of User to User Public key that cant be interfered with. Trusted Directory has a secret key and public key PK_{TD} and trusted directory is accessible by anyone.

2. Trusted Directory signs transmission of data so receiving users can check if there was a Man in the Middle Attack. $Sig[SK_{TD}, PK_B]$. However there is still an issues: MITM can get give Alice his/her own public key and pretend to be bob by passing on Trusted Directory Data to Alice (Alice thinks this is safe because it was 'signed' by TD, however MITM could have sent alice his). To get around this, have TD send the following 'BOBS PUBLIC KEY IS PK_B'. Thus the TD includes data on whose key they are signing and alice can confirm whose key they receive.

3. MITM requests Eve's public key, and TD answers with $Sig[SK_{TD}, PK_M]$. Then Eve can send this back to Alice instead of $Sig[SK_{TD}, PK_B]$ Instead, send $Sig[SK_{TD}, \text{Bob's public key is } PK_B]$

4. **Replay Attack**: Bob changes keys, so he know has SK'_B and PK'_B. We imagine the MITM wants to intercept the TD saying PK'_B and replace it with PK_B. Alice must be able to determine it is the latest so we put a time stamp.

   (a) Alice says she wants PK_B and gives a nonce which she wants them to use. TD send back $Sig[SK_{TD}, \text{Bob's public key is } PK_B, nonce]$ This nonce shows the message was generated now, not earlier.

5. Caveats: central point of attack (and failure), Scale-ability: it has everyone's PK, important to authenticate users to TD when updating PK, the service has to be online at all times during PK operation because you must fetch PK

2.1.2 Digital Certificates

1. **Certificate**: a way to represent association between name and PK as certified by a third party (CA = certificate authority), like Verisign

2. Using RSA: $Sign(SK_{CA}, \text{name=PK})$

3. Alice obtains the certificate for PK_B from anyone and she knows she can trust it. This means it doesn't have to be online at all times (one advantage)

4. Scale-ability also still a problem, although central point of failure no longer is

5. **Certificate chains or hierarchical PKI**: The Gov. of CA Jerry can maintain around 100 certificates, including UCB president, who has his own 100 in turn, including Popa, etc. $Sign(SK_J, Presidential = PK_P)$

6. If Alice wants to send a message to Nick - she asks people for certificate of Nick - then she gets back $Sign(SK_{UCBPresident}, Nick = PK_N)$. Then she must get the $Sign(SK_J, Presidential = PK_P)$ to verify the UCB president. Finally, everyone has hard-coded PK_J, so she can verify him as well. Now she knows that she received Nick's PK.

2.1.3 Revoking Certificate

:

1. $Sign(SK_{CA}, PK_B = Bob$Experices on Nov 29, 2016 Checks that the client is still valid

2. Use revocation lists: browsers put together of no longer valid keys that did not expire yet - black list