Digital Signature on Blockchain

UTBI

History

- What's in a signature?
- Principal characteristics of paper signatures
  - always equal
  - non transferable

Requests

- Authenticity
  the signature must convince that the signer actually wanted and in fact did sign the thing
- Non Reusability
  each paper must have its own signature
- Non Repudiability
  a signer could not repudiate a signature he made

Requests

- Unforgeability
  nobody should be able to forge a signature
- Unalterability
  nobody can alterate a document without invalidating the document

Digital Signatures

- Cryptographic way to extend paper signature in digital realities
- At least same level of security
- Born with asymmetric cryptography

Applications

Digital Signature are used in:

- Software veryfication and updates
- Private emails
- Private data exchange
- Online games in consoles (such as PlayStation)

Asymmetric Cryptography

- One key for encrypting, one for decrypting
- One key is secret (or private)
- The other is public - everyone can see it
  - It is impossible to recover the secret key from the public one

A simple protocol - Signing

In order to produce a signature, Alice will

1. Produce a message
2. Sign the (hash of the) message - encrypt the (hash of the) message with her secret key
3. Send the message *and* the signature to Bob

A simple protocol - Verification

In order to verify the message, Bob will:

1. Receive and read the message
2. Decrypt the signature - with Alice's public key
3. Signature valid $\Leftrightarrow$ (function of the) decryption equal to the (function of the) message

Consequences

Two points from the previous example:

- The signature is a different file
- The signature is dependent from the message:
  for each message and for each secret key there is one and only one signature

Signcryption

In general:

- Not every signature algorithm must encrypt
- Every asymmetric encryption algorithm can be used to sign and verify messages

General Overview

Components of a Digital Signature Scheme (DSS):

- Preliminary phase
  - Prearranged parameters
    set of constants and functions decided by parties before the use of the scheme
  - Generation Algorithm
    creates the keys
- Signature Phase
  - Signing Algorithm
    sign the messages
  - Verifing Algorithm
    verify if the signature is valid

Examples list

Digital Signature Schemes used today:

- Shnorr signature (bitcoin?)
- DSA
  - ECDSA (bitcoin, ethereum, . . .)
- Ring Signature (old monero)

Schnorr Signature - Preliminary phase

- Parameters
  - a predefined generator g in a predefined set G
  - a hash function H
- Key Generation
  - Choose x from an allowed set as private key
  - Compute the public key $y = g^x$

Schnorr Signature - Signature phase - Signing

To sign a message M

- Choose a random k from the allowed set.
- Compute $r = g^k$
- Compute $e = H(r \parallel M)$
- Compute $s = k - xe$

The signature is the pair (s, e) and the private element are (k, x)

Schnorr Signature - Signature phase - Verifying

- Compute $r_v = g^s y^e$
- Compute $e_v = H(r_v \parallel M)$

The signature is verified if $e_v = e$:

$$r_v = g^s y^e = g^{k-xe} g^{xe} = g^k = r$$

and so

$$e_v = H(r_v \parallel M) = H(r \parallel M) = e$$

Shnorr signature - Attacks

Note that k must change for every message, otherwise for a couple of messages M and M':

- $r = g^k$
- $e = H(r \parallel M)$ and $e' = H(r \parallel M')$
- $s = k - xe$ and $s' = k - xe'$

And the attacker will recover the private key x:

$$s - s' = (k - k) - xe + xe' = x(e' - e)$$

$$\Rightarrow x = \frac{s - s'}{e' - e}$$

DSS in Blockchain technologies

The use of digital signatures in blockchain:

- Sign transactions
- Multisignature and escrow services
- Timestamp (via block number)
- The blockchain itself is a DSS

Privacy

- Pseudo-anonymity is not anonymity
- Some signatures (e.g. ring signatures) can hide the sender of a transaction giving more privacy
- Other method involve zero-knowledge proofs, but that does not involve DSS directly

Security

- Key length
- Strong random generators

The Future

- Bulletproofs
  - Bulletproofs based smart contracts
- Mimblewimble
- Quantum resistant signatures
  - lattice based
  - hashbsed