Privacy based Public Key Infrastructure (PKI) using Smart Contract in Blockchain Technology

Sivakumar.P
M.Tech (CSE), Sem-III, NIT Trichy

Guide:- Dr Kunwar Singh
Assistant Professor

Problem Statement

Privacy based Decentralised (PKI) using Smart Contract in Blockchain Technology

Public Key Infrastructure (PKI)

What is PKI?

Provides secure binding of public keys and users

PKI Functions ?

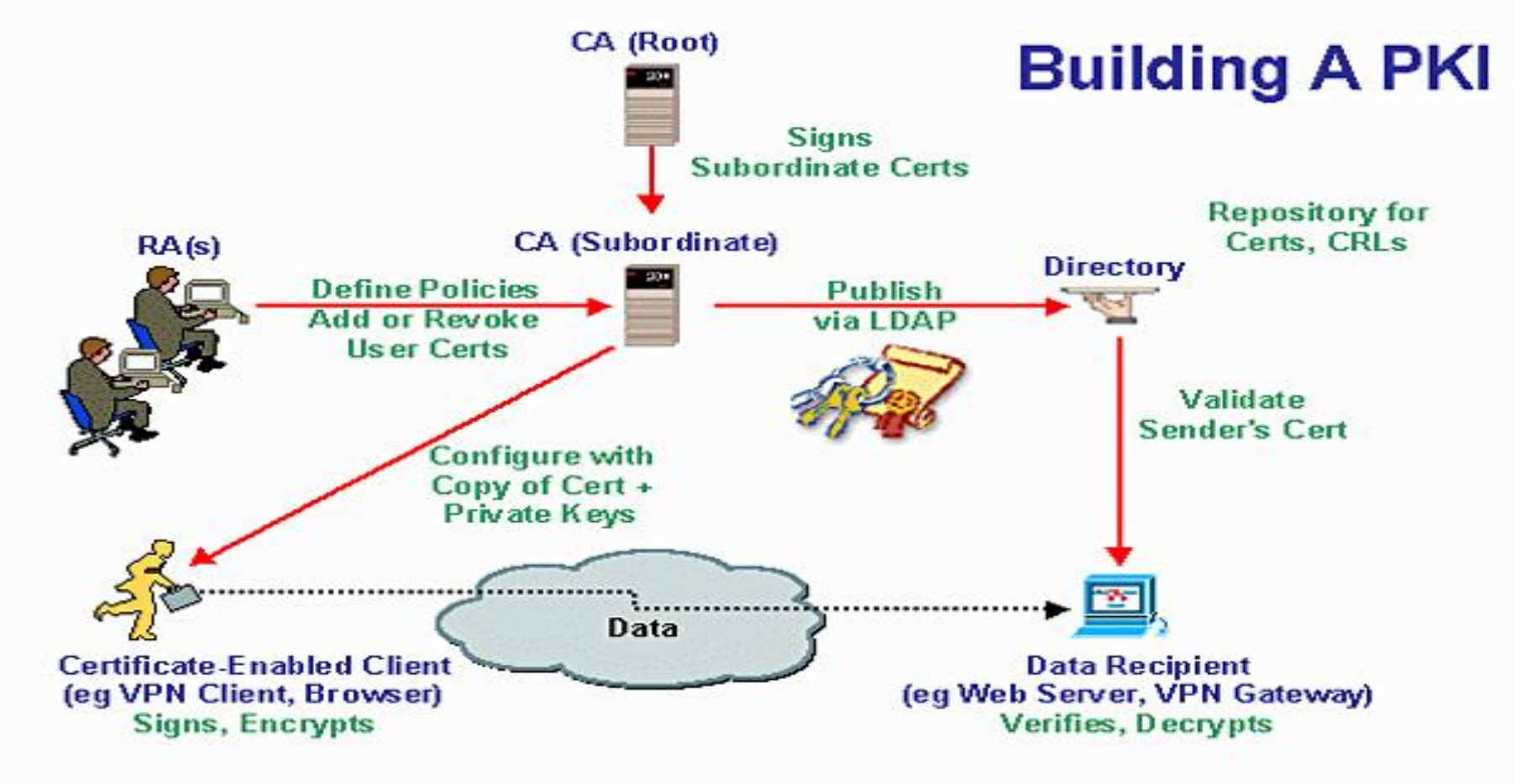
- Public Key Authentication
- User identification
- Access Control
- Authorisation

Why to authenticate Public Keys?

Access control of users through their Public Keys

Existing Solutions

1. Public Key Certificates issued by Certifying Authority (CA)



Existing Solutions - Limitations

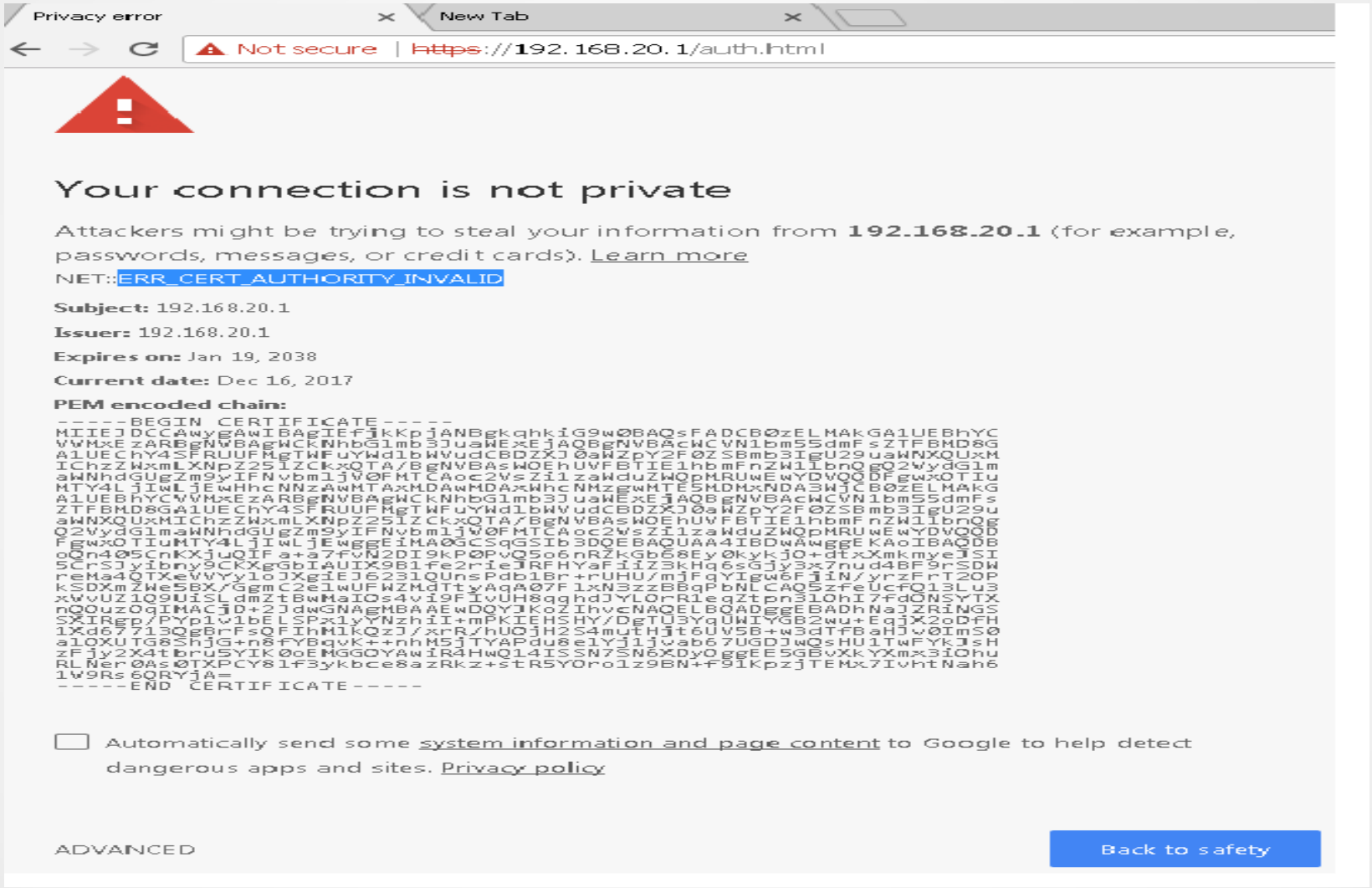
1. **Public Key Certificates issued by Certifying Authority**

Limitations

- Certificate forgery
- Man in the middle attacks using rogue certificates
- Violating identity retention
- Centralised control
- Single point failure
- Maintenance of certificates
- Unpatched Software based attacks
- False sense of security

Existing Solutions - Limitations

Public Key Certificates issued Certifying Authority



Existing Solutions - Limitations

1. **Public Key Certificates issued by Certifying Authority**

DigiNotar CA Security Breach

- CA Servers were hacked
- Multiple SSL certificates were stolen
- Issued rogue certificates to all Google internet Domains
- Fraudulent issue of dozens of public key certificates by hackers
- Stolen certificates were used in **Man in the Middle attacks**

Trustwave CA breach

- Sold digital certificate for a customer to eavesdrop on encrypted employee traffic
- Customer acted as a CA hijacked and issued new certificates
- Browsers were updated and started using rogue certificates

Existing Solutions

2. **Pretty Good Privacy** (PGP)

Decentralised Model based on Web of Trust

Private Key Ring

Timestamp	Key ID*	Public Key	Encrypted Private Key	User ID*
• • •	• • •	• • •	• • •	• • •
T_i	PU_i mod 2^{64}	PU_i	$E(H(P_i), PR_i)$	User i
• • •	• • •	• • •	• • •	• • •

Public Key Ring

Timestamp	Key ID*	Public Key	Owner Trust	User ID*	Key Legitimacy	Signature(s)	Signature Trust(s)
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •
T_i	PU_i mod 2^{64}	PU_i	$trust\_flag_i$	User i	$trust\_flag_i$		
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •

* = field used to index table

Existing Solutions - Limitations

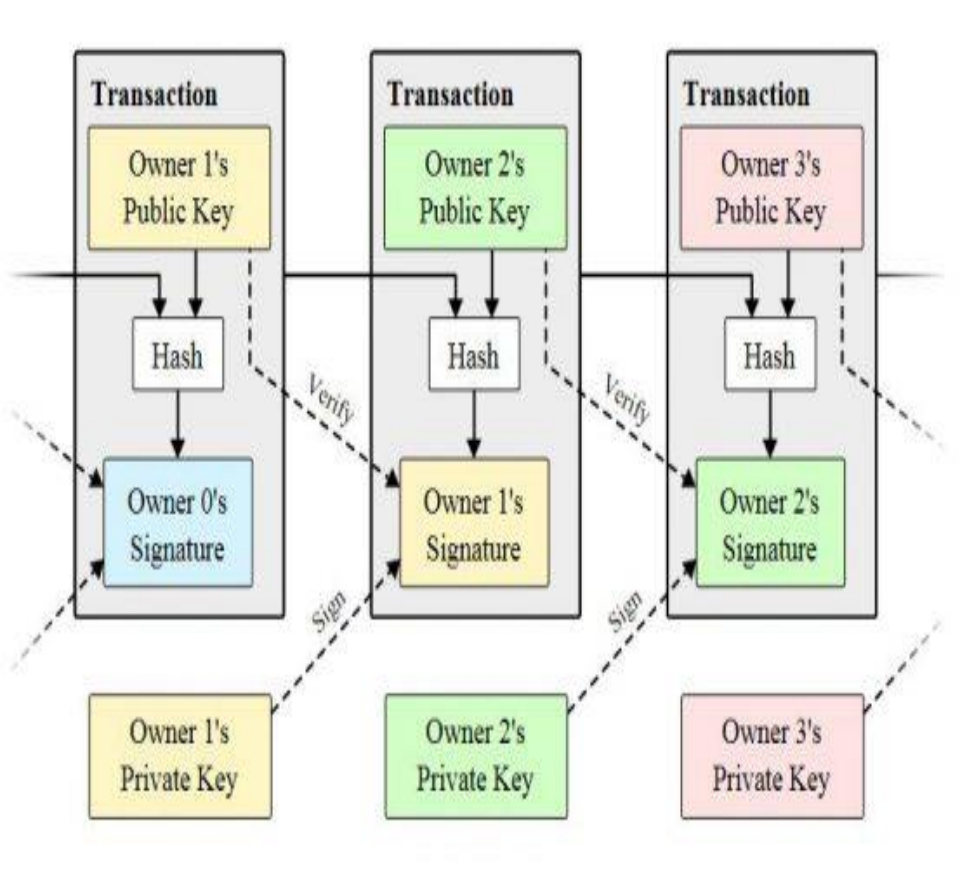
2. **Pretty Good Privacy** (PGP)

Limitations

- Authentication problems
- Honest policy cannot be trusted
- PGP key maintenance overhead
- Compatibility issues
- Limited access to additional security tools
- Cost of maintaining client software
- Web of trust is not scalable

Proposed Solution to overcome the limitations

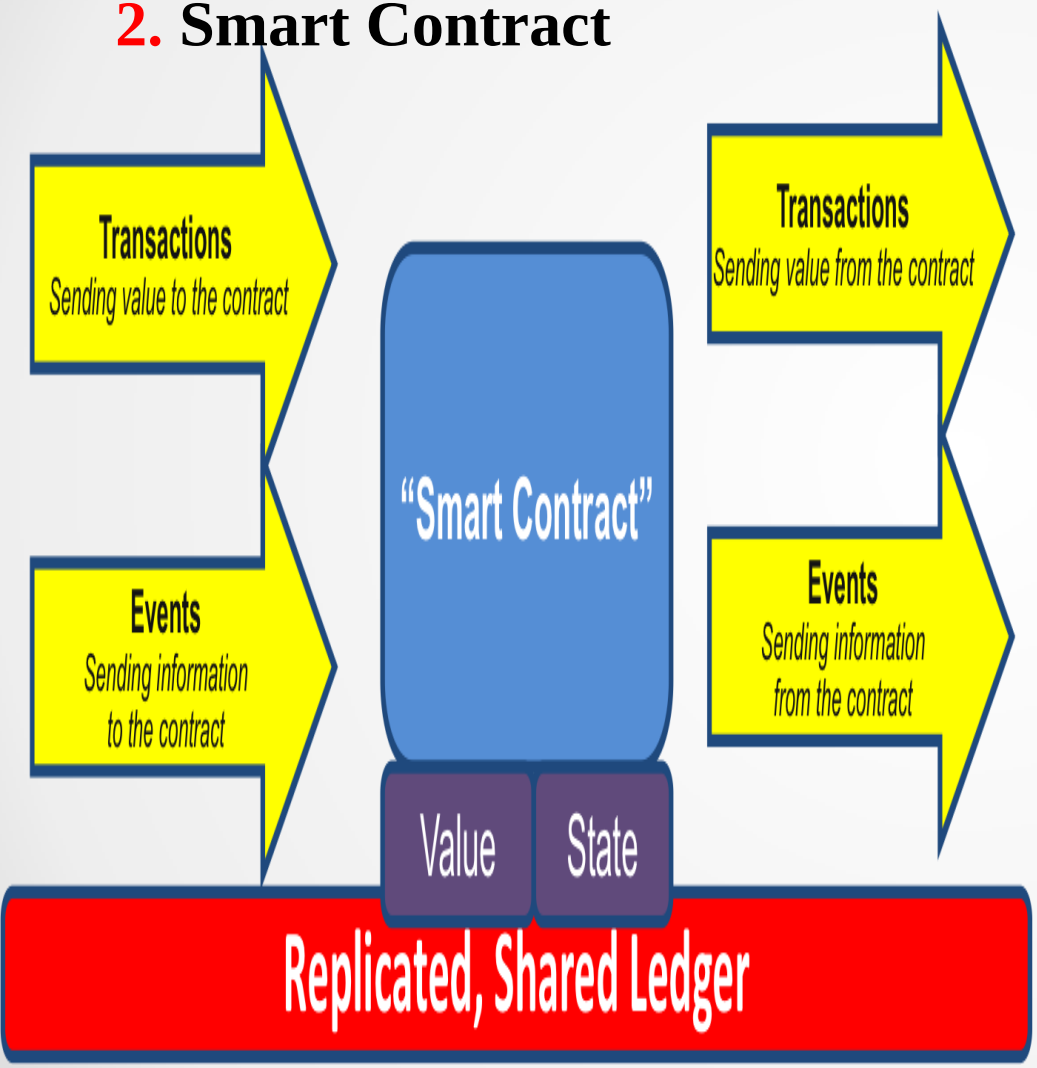
1. Block chain - decentralized ledger



- Decentralized applications (DApps)
- Immutability
- Peer-to- peer network
- DApps have a built-in cryptocurrency
- Proof-of-work consensus protocol
- IPFS (Inter Planetary File System)
- Proof of authenticity
- Proof of existence
- Technology of Bitcoin

Proposed Solution to overcome the limitations

2. Smart Contract

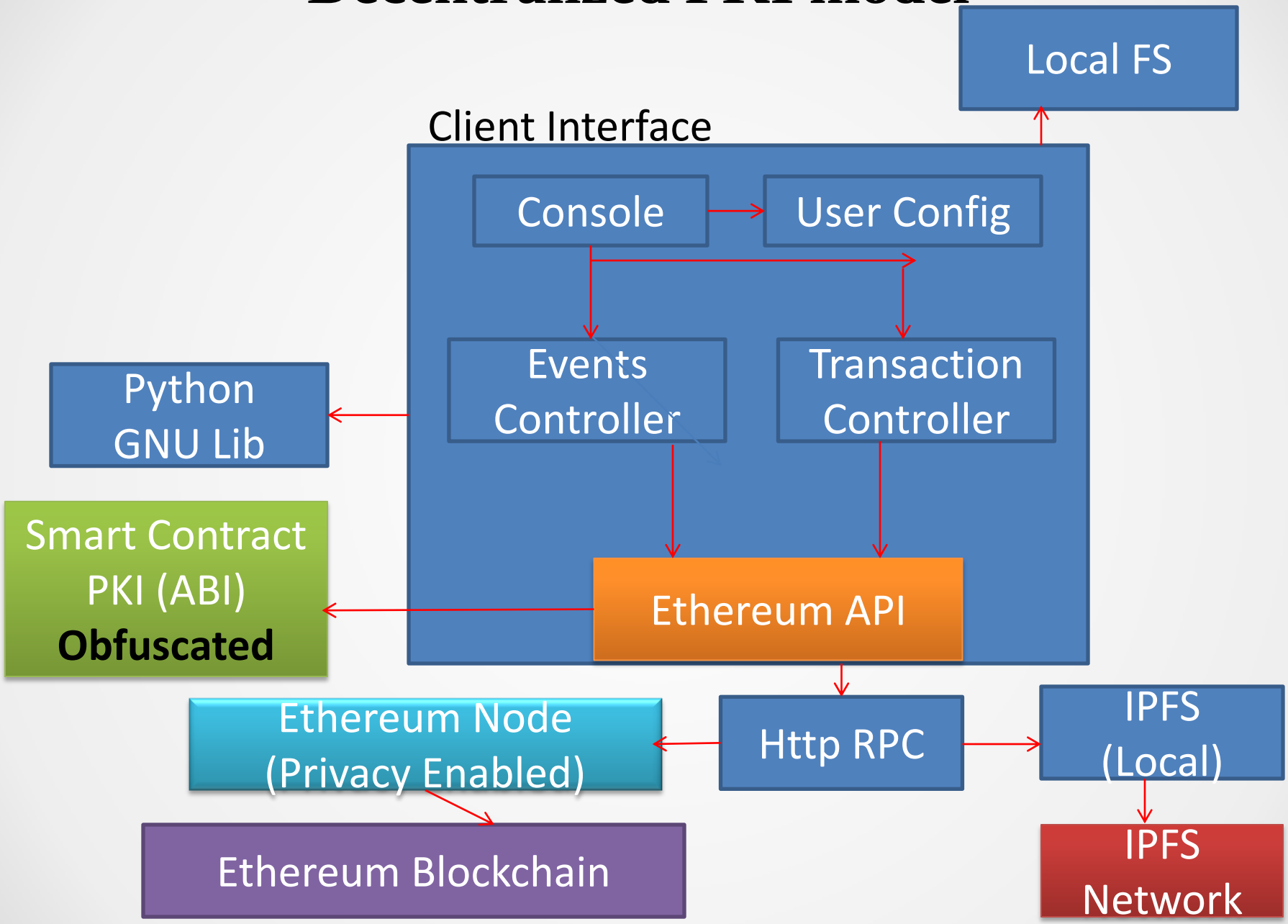


- Automatically enforces business rules
- Enforce compliance
- Transfer of assets agreements
- Legal Processes
- Rules as contractual clauses
- Cannot alter rules
- Flow executed automatically after validation
- Low level stack-based bytecode
- Raise events

Why Smart Contract for Decentralized PKI

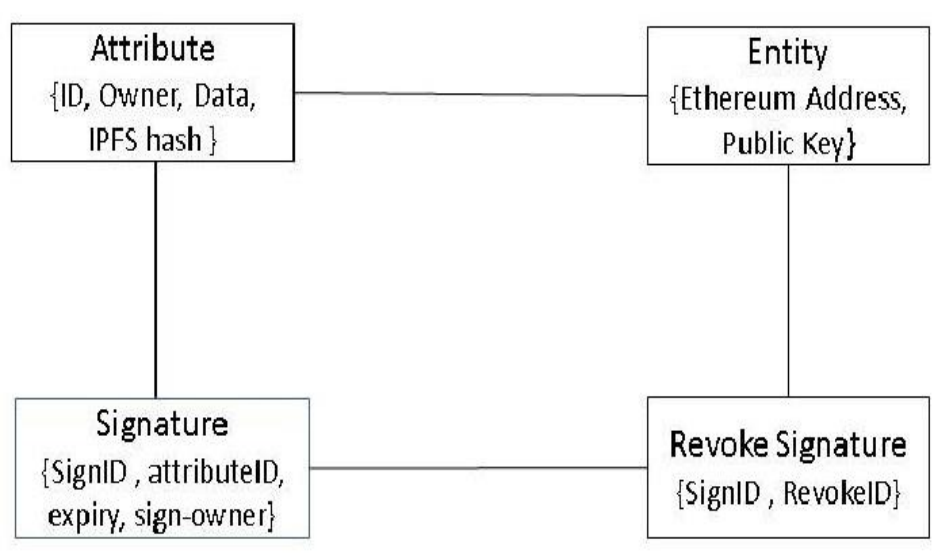
- Ethereum is a programmable blockchain
- To enable Plug and Play flexibility
- Structured way to handling methods and events to control blocks
- Obfuscating of smart contracts
- Stack based bytecode injected in EVM, no forgery of contract clauses
- Act as black box for internal logic
- Reduces transaction cost
- Averts Decompiling

Decentralized PKI model



Decentralized PKI Functions

1. **Registration of Entity**
2. **Signing of attributes**
3. **Revocation of sign**
4. **Retrieval and search**
5. **PGP for WoT model**



Privacy in Decentralized PKI

I. Key Management in Ethereum

- **Private key** is randomly generated of size 32 bytes using ethereum standard secp256k1 curve.

- **Public key** is of 64 bytes is derived from private key using Elliptic Curve Digital Signature Algorithm (ECDSA).

- **Ethereum address** of the node is generated using Keccak256 hash of the public key and the last 20 bytes of that resulted hash is the address of the node.

Privacy in Decentralized PKI

II. Address Translation for Anonymity

Consider the Receiver Node A has a public key '**A**' and private key '**a**' such that

$\mathbf{A} = \mathbf{a.G}$, where G is the generator of an elliptic curve (1)

Node B generates an ephemeral key pair with '**B**' an elliptic curve point and '**b**' is a 256 bit integer. Node B sends '**B**' to Node A and now both can calculate share secret

$$\mathbf{B} = \mathbf{b.G} \quad (2)$$

$$\mathbf{b.A} = \mathbf{B.a}$$

$\mathbf{b.a.G} = \mathbf{b.G.a}$, from (1) and (2)

Privacy in Decentralized PKI

II. Address Translation for Anonymity

Now Node B can generate public key using this shared secret

$$\mathbf{Keccak256[A + H(b.A).G]}$$

Node A can spend the money using private key

$$\mathbf{[a + H(a.B)]}$$

III. Smart contract obfuscation enables data privacy

Transaction Cost

- Smart contracts can send multiple payments and optimizes cost
- Linear consumption of gas for attributes
- Gas cost varies with data size
- Large amount of data to be stored in IPFS and hash of IPFS link can store in blockchain
- Gas cost of IPFS links found constant

Tools/Language

- ✓ Python
- ✓ Node.js
- ✓ Solidity (For smart contracts)
- ✓ Go (For Blockchain)
- ✓ Ethereum (Cryptocurrency Virtual Machine)

Conclusion and Future work

- Transparent system
- No single point failures
- System can be integrated to ethereum based Dapps easily
- Testing for heavy traffic
- Obfuscation of smart contract could affect performance of system
- Privacy can be improved using Ring signatures, Zero Knowledge Schemes

References

- Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system, 2008." (2012): 1-9.
- Conner Fromknecht, Dragos Velicanu, and Sophia Yakoubov. Certcoin: A namecoin based decentralized authentication system,. 2014.
- G. Wood. Ethereum: A secure decentralised generalised transaction ledger. http://gavwood.com/paper.pdf.
- A Next-Generation Smart Contract and Decentralized Application Platform. https://github.com/ethereum/wiki/wiki/%5BEnglish%5D-White-Paper
- Andreas M Antonopoulos. Mastering Bitcoin: Programming the open blockchain. O'Reilly Media, 2017.
- IPFS reference. https://ipfs.io/docs/commands/
- https://www.ethereum.org/
- https://blockchain.info

Thank You…