Secret-in.me

A pentester design of ~~password~~ secret manager

Who am I ?

Security engineer

Working at SCRT France !

Password manager

Password	Secret
A string	*Information shared by very few people*
• You have to remember	You should find a way to share it.
• To authenticate yourself	Only the concerned people should access it.
• Others can't guess	

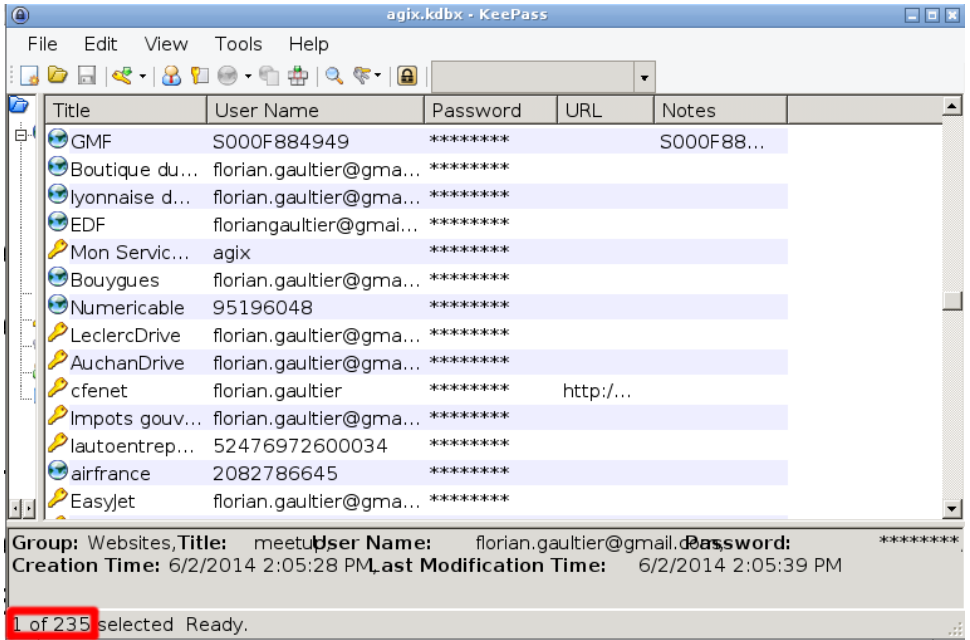
Very hard for human mind.

Try to remember
`4csVIus9TG82BXRedA5B5gAZjHKm7dNa`

Multiple services => multiple passwords

Impossible to do with your mind

Try to remember 235 random strings...



Password manager

Company's headache : managing access authorization

- Multiple equipment
- Employees in and out

SSO/LDAP binding

- Linked to the Active Directory
- Centralized management
- One private password by employee
- Multiple access to the service
- Access log rely on the service
- Service should support SSO !

KEEPASS

- Not easy to share
- Centralized management
- Work with any services
- One private password by employee
- One access to the service
- Useless access log on the service

Password manager

Pentest time

Pick your favorite vulnerability

- WPAD + weak password
- Outdated software
- Default passwords
- ...

SSO/LDAP binding

KEEPASS

- Identify user using keepass
- Wait for the keepass to be unlocked
- KeeFarce
- Do it for every users

Password manager

What's a good secret manager ? (from our point of view)

- Secret encryption with standards => Obvious
- Open source => To check claimed security
- Limited dependencies => Reduce trust surface
- Cryptography <u>not written by us</u> => Crypto is hard
- Double authentication standard => Obvious
- Sharing possibility => Needed in company
- Logging possibility => Needed in company
- Browser integration => Easier to use

Secret-in.me

TADAAAA!

Started in 2015 after Gandi 15 years anniversary
Improved **a lot** more recently

Secret-in.me

- Reduce trust surface
  - "We don't want to install new client software"
    - Maintenance, backdoor…
  - We trust the browser (I hope you do)
  - W3C wrote WebCryptoAPI
  - Browser can do cryptography !
  - For now, only Blink (google chrome and chromium engine) support every standards.

- You only have to trust your browser and secret-in.me
- *Unfortunately not if you want a pretty UI...*

Secret-in.me

- Storage
  - JSON is easy to transport
    - Write it on file anywhere (like keepass)
    - Use a server to save it for you

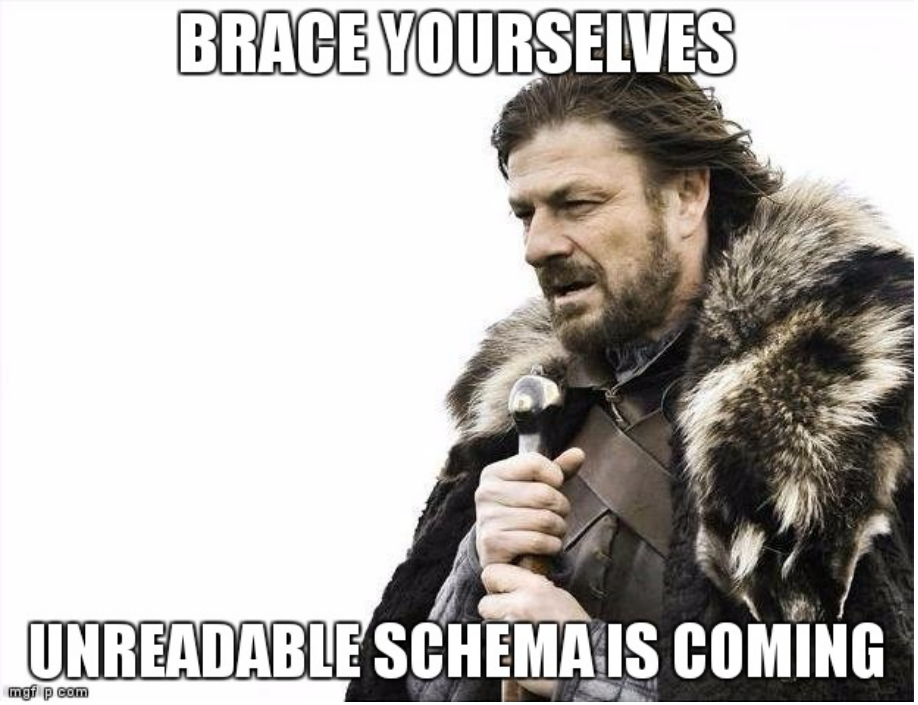
- Cryptography is done client side
  - Compromised server can't read your secrets
  - Compromised network can't read your secrets

- Using server can add privileges and logging dimension
  - Read only, Read/write, Read/Write/Share
  - Who, what, when

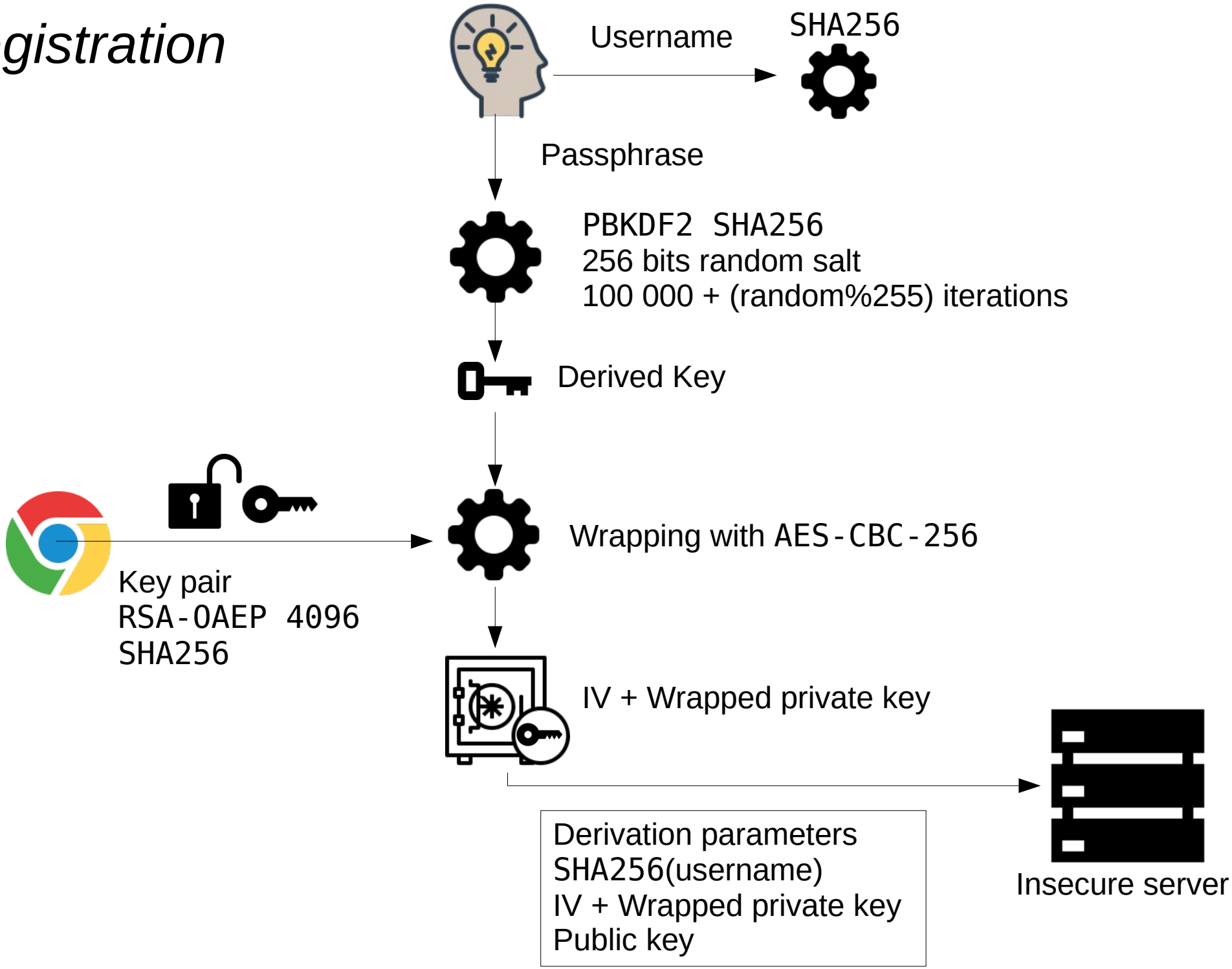
Secret-in.me

How it works

Cryptographic layer



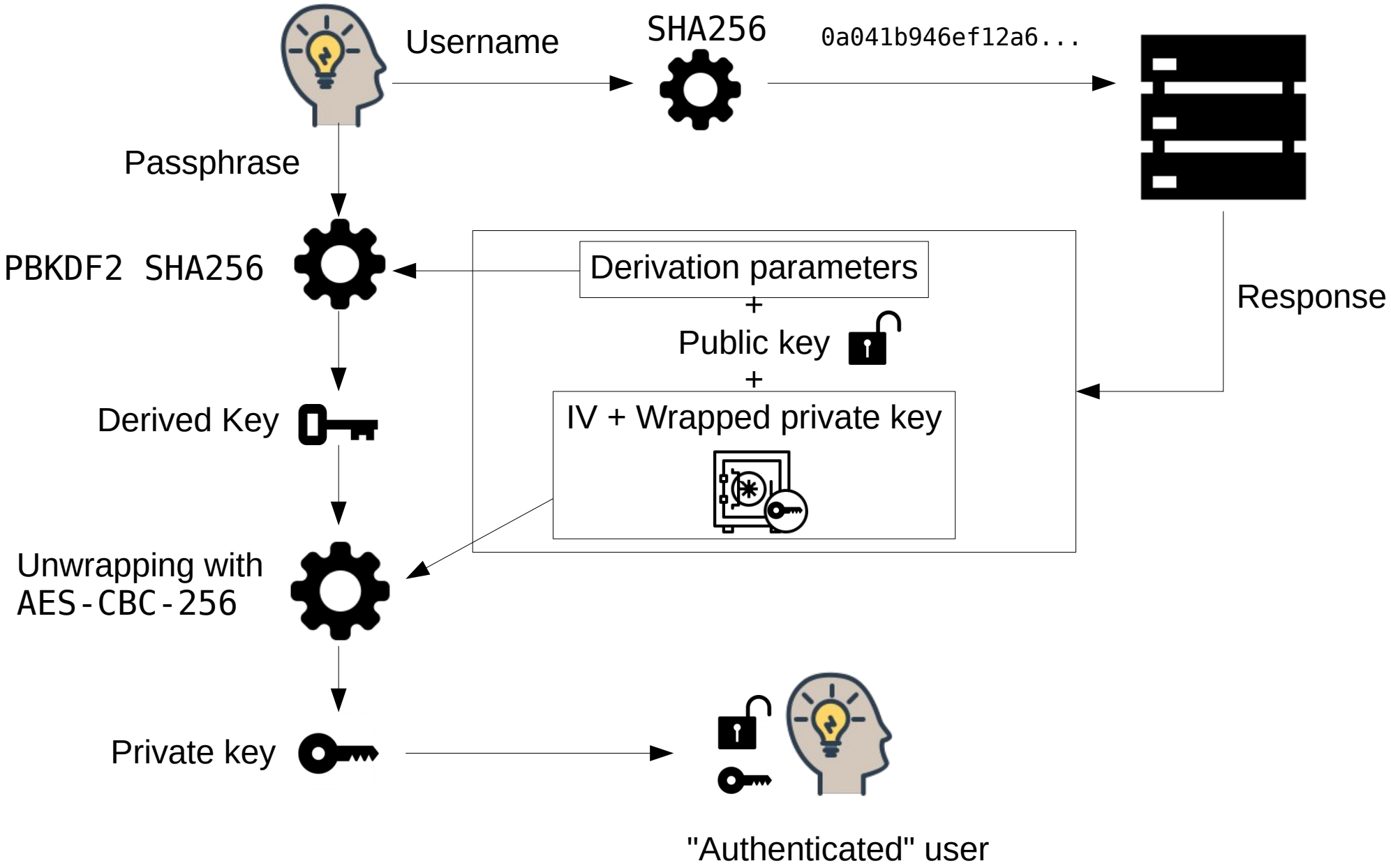
Registration



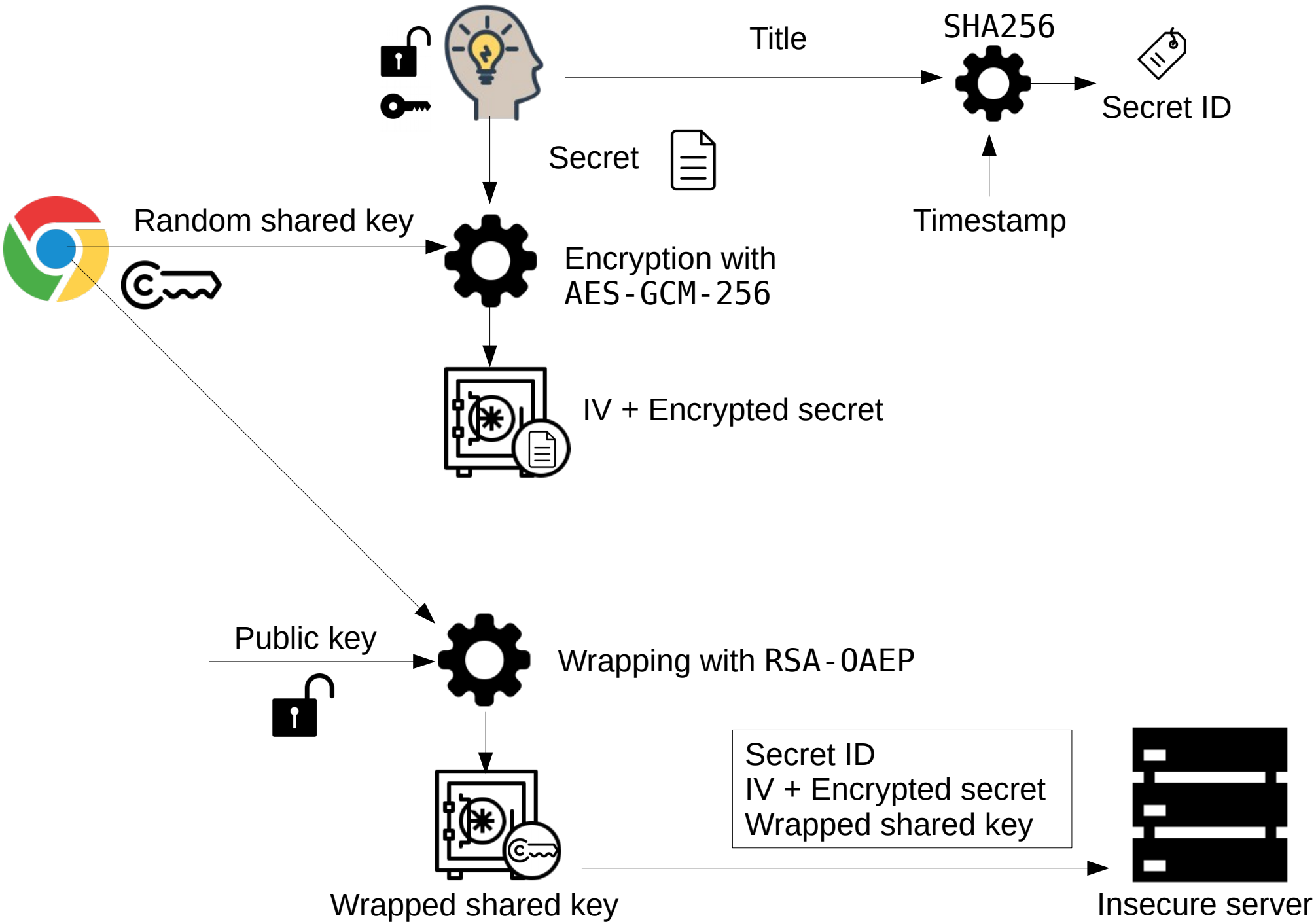
Server View


```
{
    "0a041b9462caa4a31bac3567e0b6e6fd9100787db2ab433d96f6d178cabfce90": {
        "keys": {},
        "pass": {
            "iterations": 100105,
            "salt": "1e473abdb40125b8f07b6a77959413f2fed862ffa4c81cbcb5db17de7aebcf48"
        },
        "privateKey": {
            "iv": "ee73cf663438360febc74d5d6f8720f4",
            "privateKey": "47da2b54a55198[...]9e0d64fda2db9211ad7d6394a9d7"
        },
        "publicKey": {
            "alg": "RSA-OAEP-256",
            "e": "AQAB",
            "ext": true,
            "key_ops": [
                "encrypt",
                "wrapKey"
            ],
            "kty": "RSA",
            "n": "nGGkuqrDLpqrggBzkmx-[...]hLt9wEFh5tQRbObcFFEZ8"
        }
    }
}
```

Login

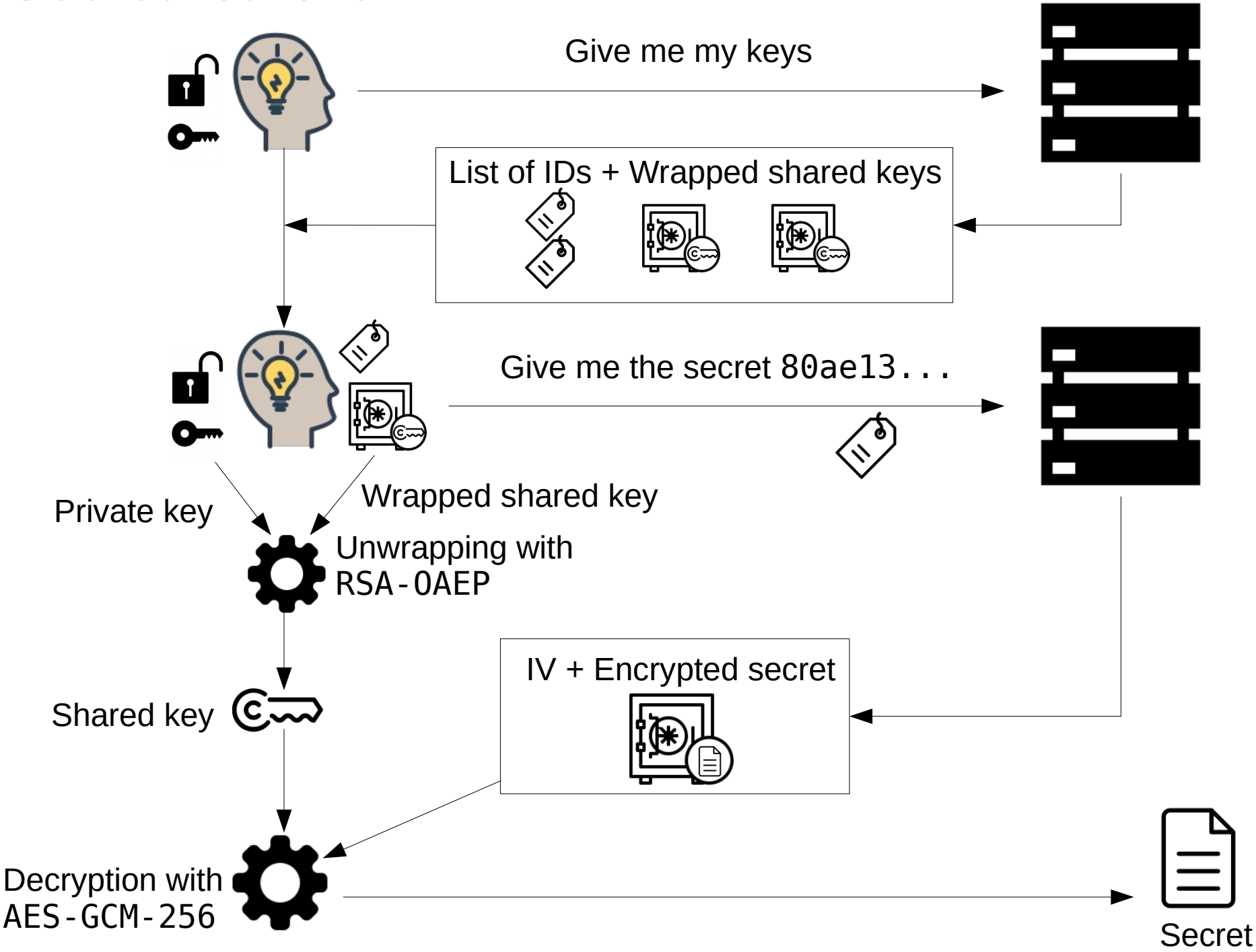

Username
SHA256
0a041b946ef12a6...
Passphrase
PBKDF2 SHA256
Derivation parameters
+
Public key
+
IV + Wrapped private key
Response
Derived Key
Unwrapping with
AES-CBC-256
Private key
"Authenticated" user

Secret creation

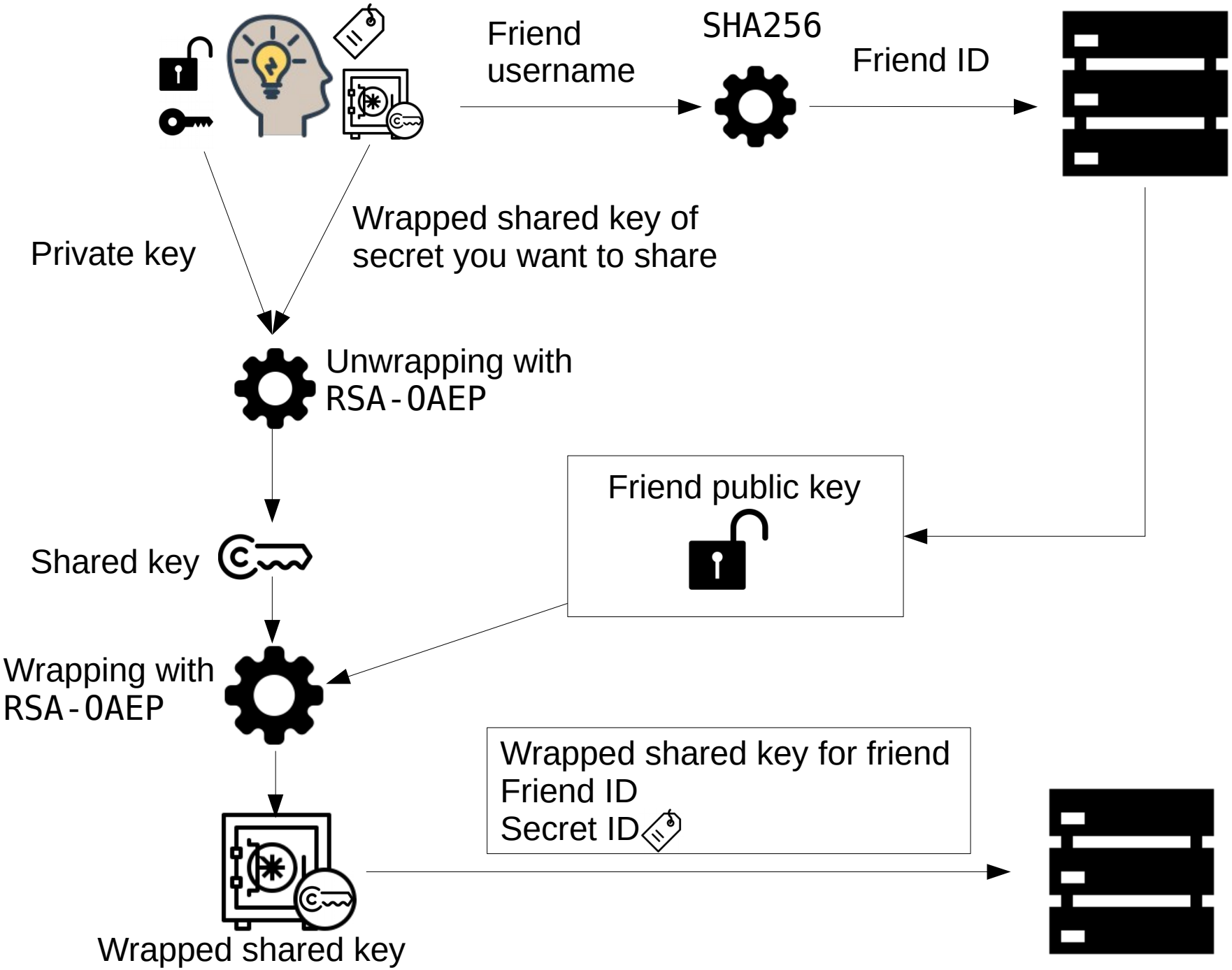


```
{
    "secrets": {
        "0839fb4655ea32255f60e4e37fe07e207be65774d8a9255bc9344403faeaead7": {
            "iv": "2e16d955f86c6589d821c7a1",
            "secret": "873c828e20ef4909cf[...]5640ac4b",
        },
    },
    "users": {
        "0a041b9462caa4a31bac3567e0b6e6fd9100787db2ab433d96f6d178cabfce90": {
            "keys": {
                "0839fb4655ea32255f60e4e37fe07e207be65774d8a9255bc9344403faeaead7": {
                    "key": "98fef3afc43e7f3d[...]26b2f833b972b3d54",
                },
            },
            "pass": {
                "iterations": 100024,
                "salt": "5dd0c60727bc84e49f0fa271bb4e7188d750e10eb0ae868df008d39464541634"
            },
            "privateKey": {
                "iv": "23ddc5828a2533c1b23ca5ffa7eb4cb0",
                "privateKey": "6fa526a3c515068537a8e033[...]8e9d8937c21db55b"
            },
            "publicKey": {
                "alg": "RSA-OAEP-256",
                "e": "AQAB",
                "ext": true,
                "key_ops": [
                    "encrypt",
                    "wrapKey"
                ],
                "kty": "RSA",
                "n": "vON4sq1SWK9bKEqXWMkG7n[...]drK24TkxJXHJ1vxLDjiIM"
            }
        }
    }
}
```

Secret retrieval



Secret sharing

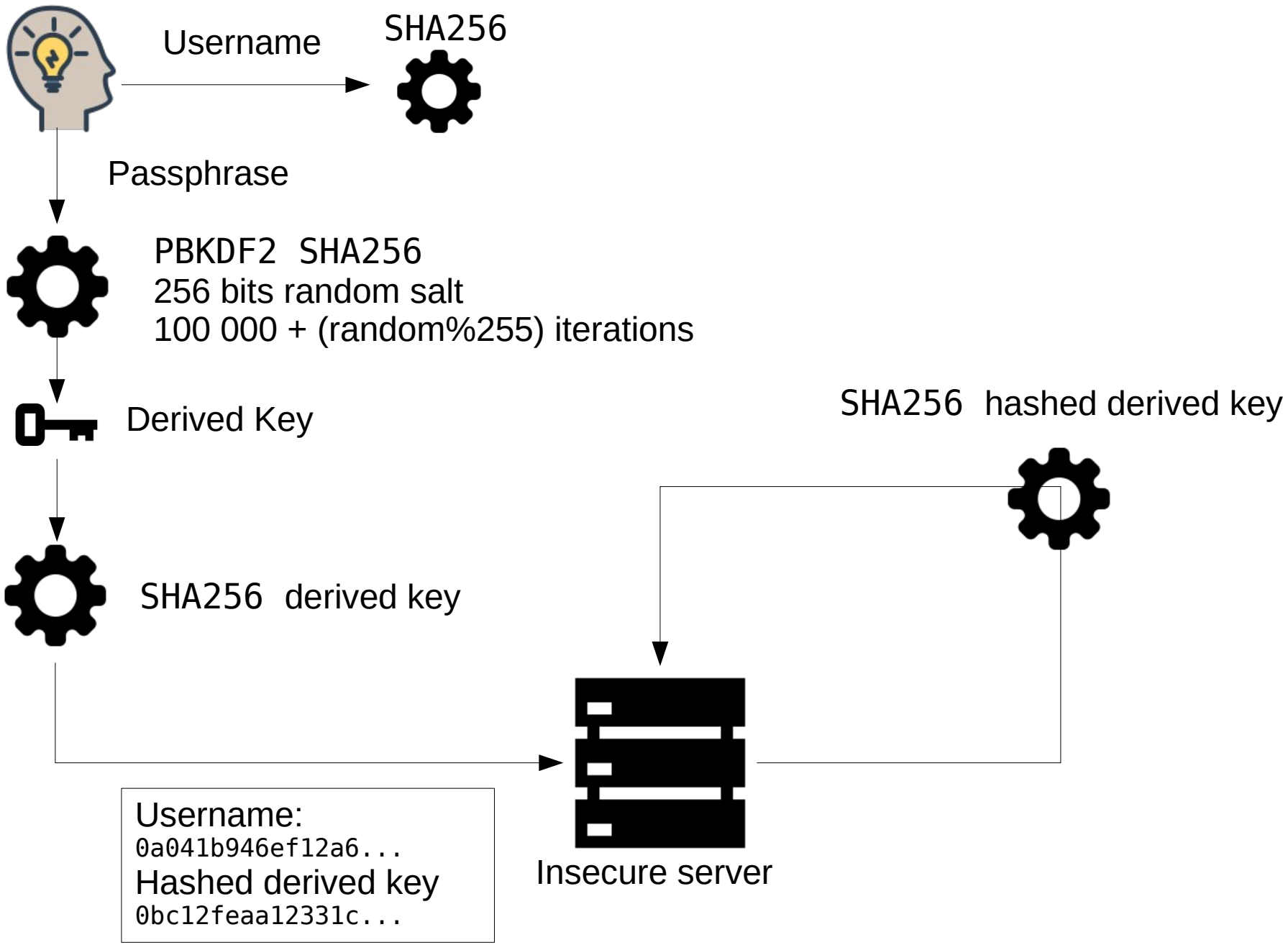


Secret-in.me

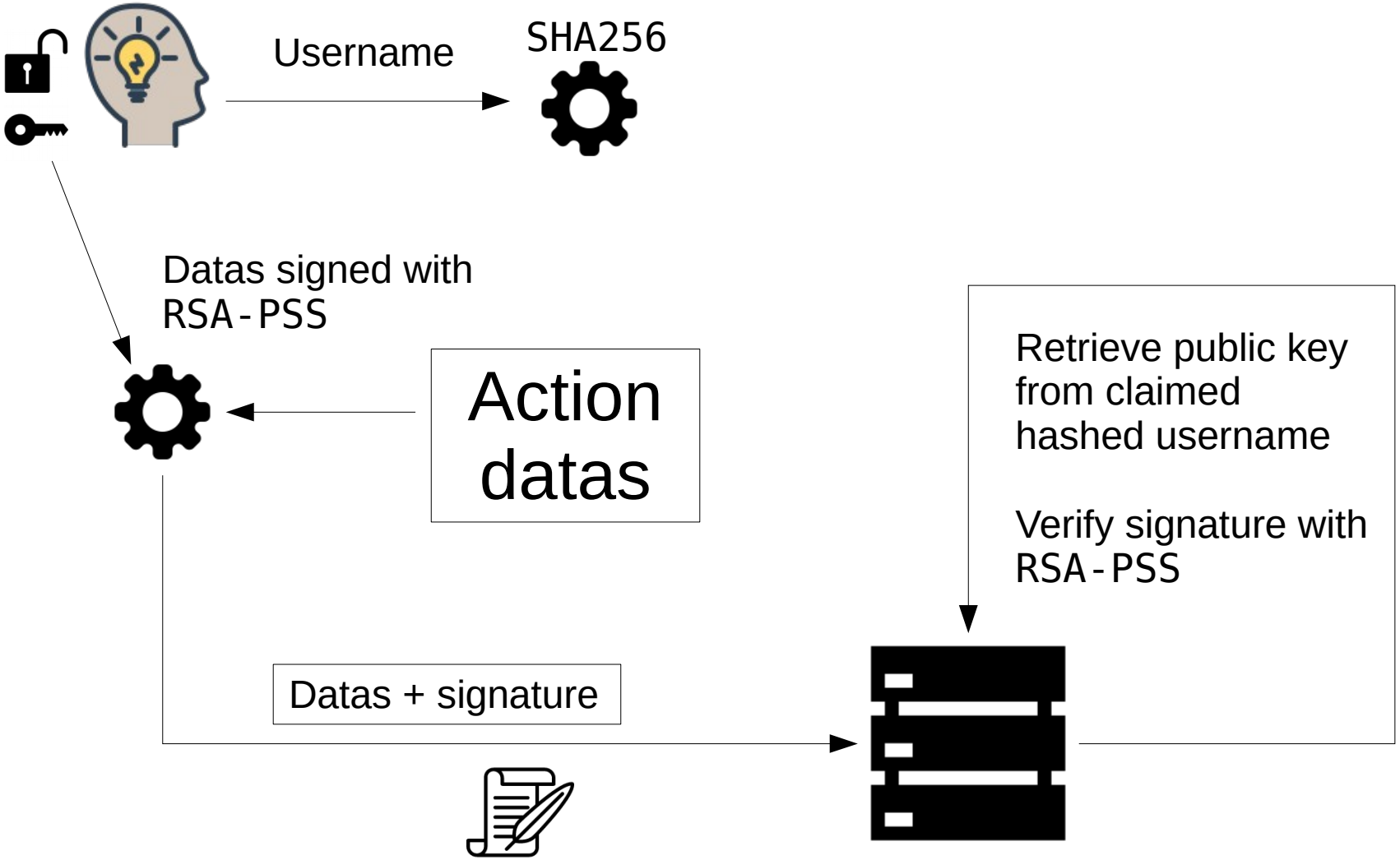
How it works

Logic layer

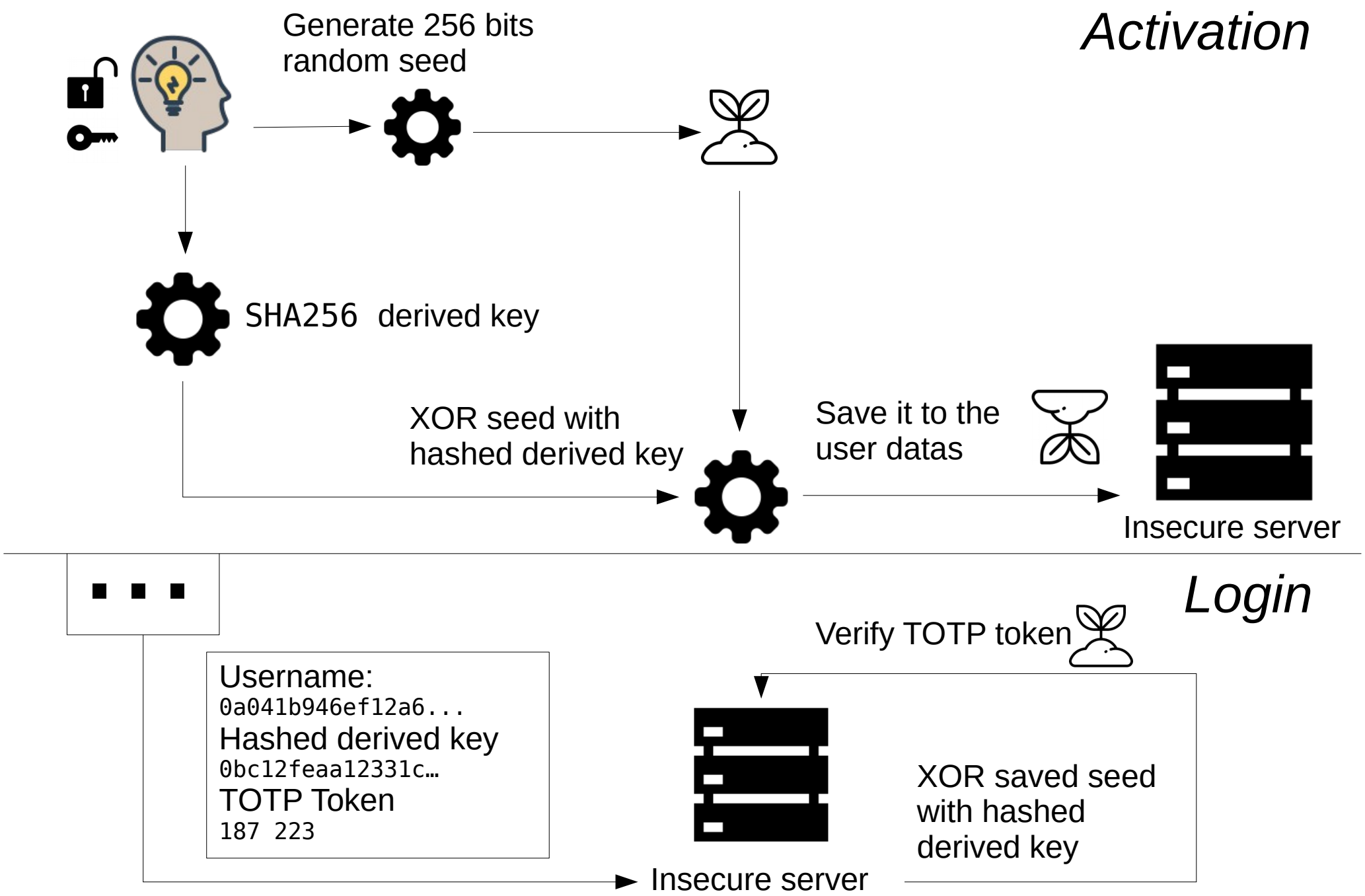
Registration / Login



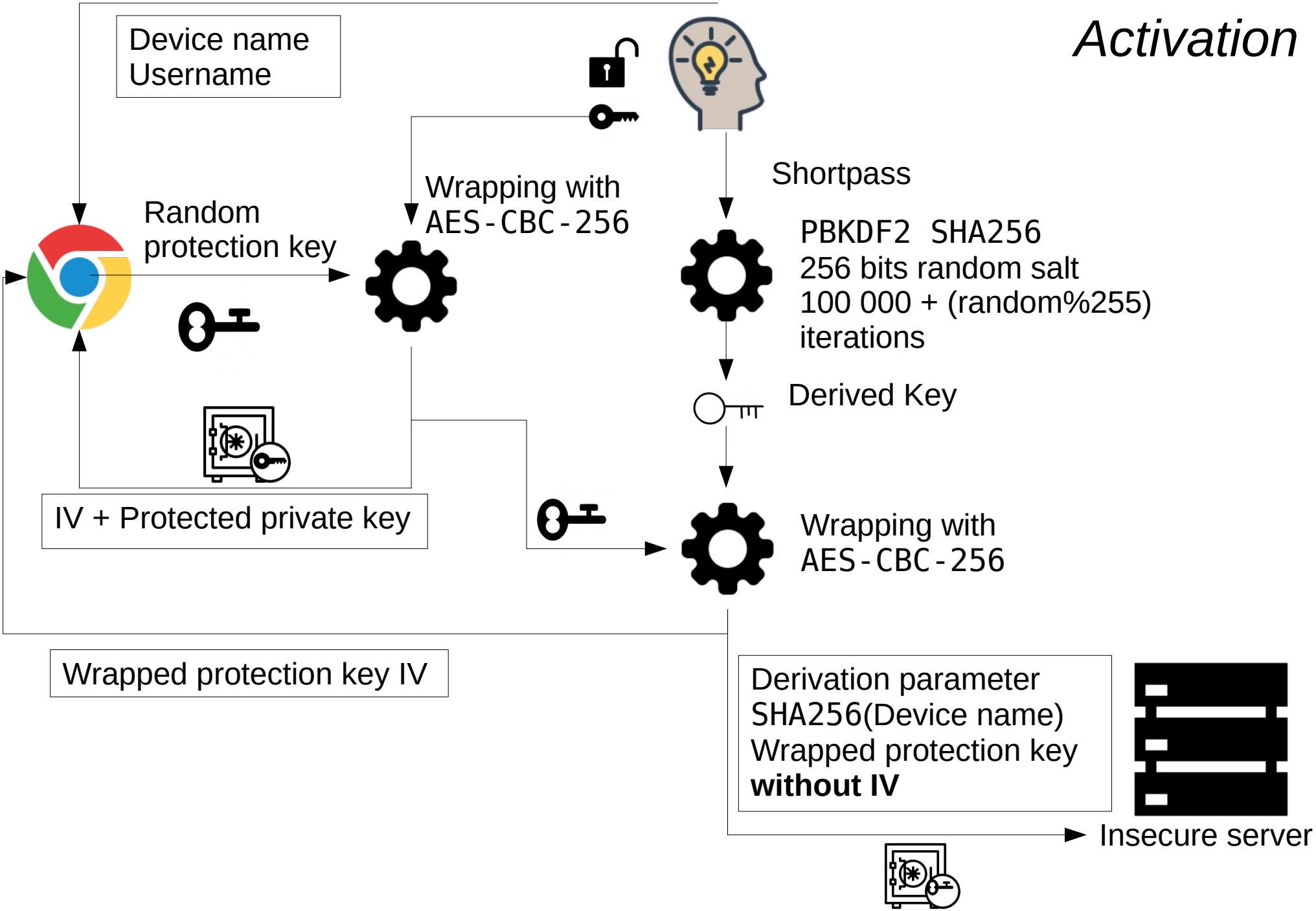
Authenticated actions



Double authentication (TOTP)

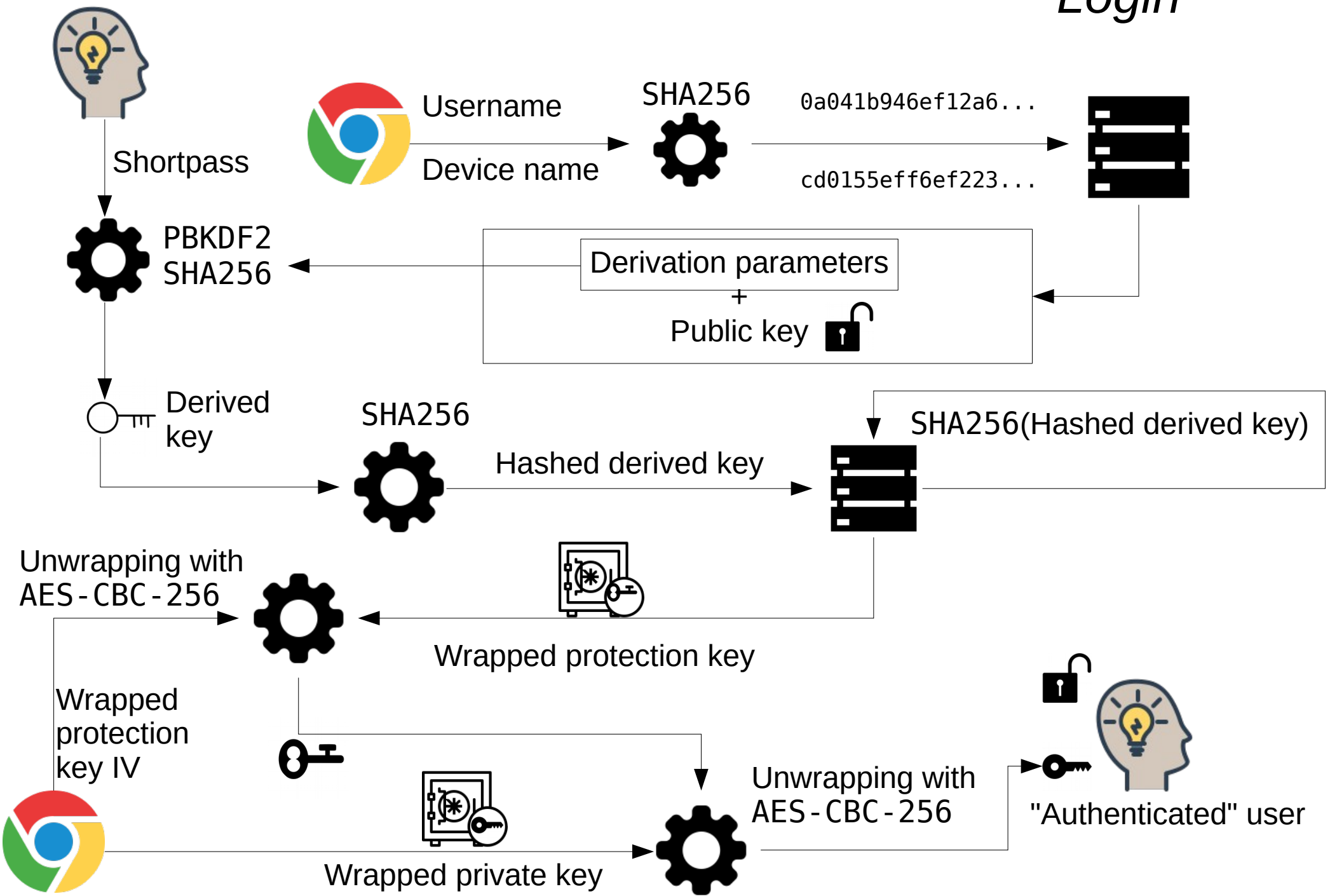


Double authentication (Trusted device)



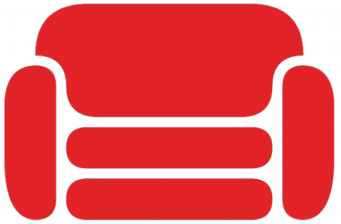
Double authentication (Trusted device)

Login



Secret-in.me

- Technologies
  - Server in nodeJS to stay in JavaScript world
  - CouchDB Database
    - Smart conflict management
    - Made for easy replication



  - Client side library without any dependencies
  - Client app using ReactJS

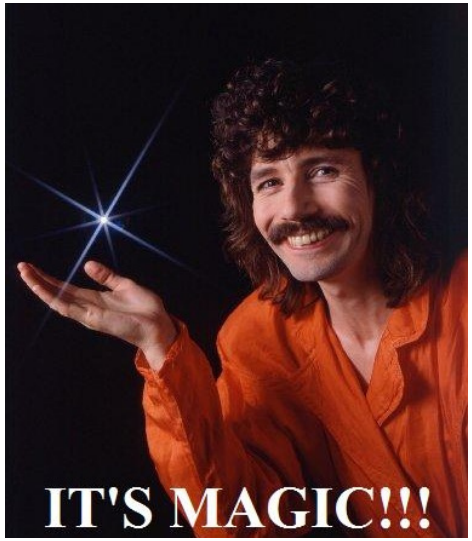
Secret-in.me

DEMO

Secret-in.me

- Problem
  - How can I save my windows password in it ?
  - I need windows access to launch my browser

- Solution



Secret-in.me

- Available on https://secret-in.me
  - Server (redis+couchdb+api) bundled by docker-compose
    - github.com/secretin/secretin-server
  - Library shipped in npm
    - github.com/secretin/secretin-lib
  - Client
    - github.com/secretin/secretin-app
  - Windows black magic
    - github.com/secretin/secretin-windows

Secret-in.me roadmap

- Find a logo !
- Offline mode (in beta)
- React-native app for iOS
- Improve UI:UX
  - Add loading information
  - Add error information
- Improve documentation for easy self hosting
  - How to setup couchdbv2 with master master replication...
- Add application settings (auto close, secret generation options...)
- Obfuscate private key in memory when decrypted