Applying Authenticity

...to Swiss Healthcare and beyond

Georg Greve
Co-Founder & CEO
georg.greve@vereign.com
https://www.linkedin.com/in/georggreve/

Vereign
Dammstrasse 16 | 6300 Zug | Switzerland
contact@vereign.com
+41 41 541 50 63

Who are Vereign ?

Applications of authenticity, based on Self-Sovereign Identity and decentralized technologies

Founded 2017 in Zug

16 FTE and growing

One of the leading teams for SSI

Among other things contributed SSI stack to Eclipse Foundation

Vereign.com

First, we must talk about the web

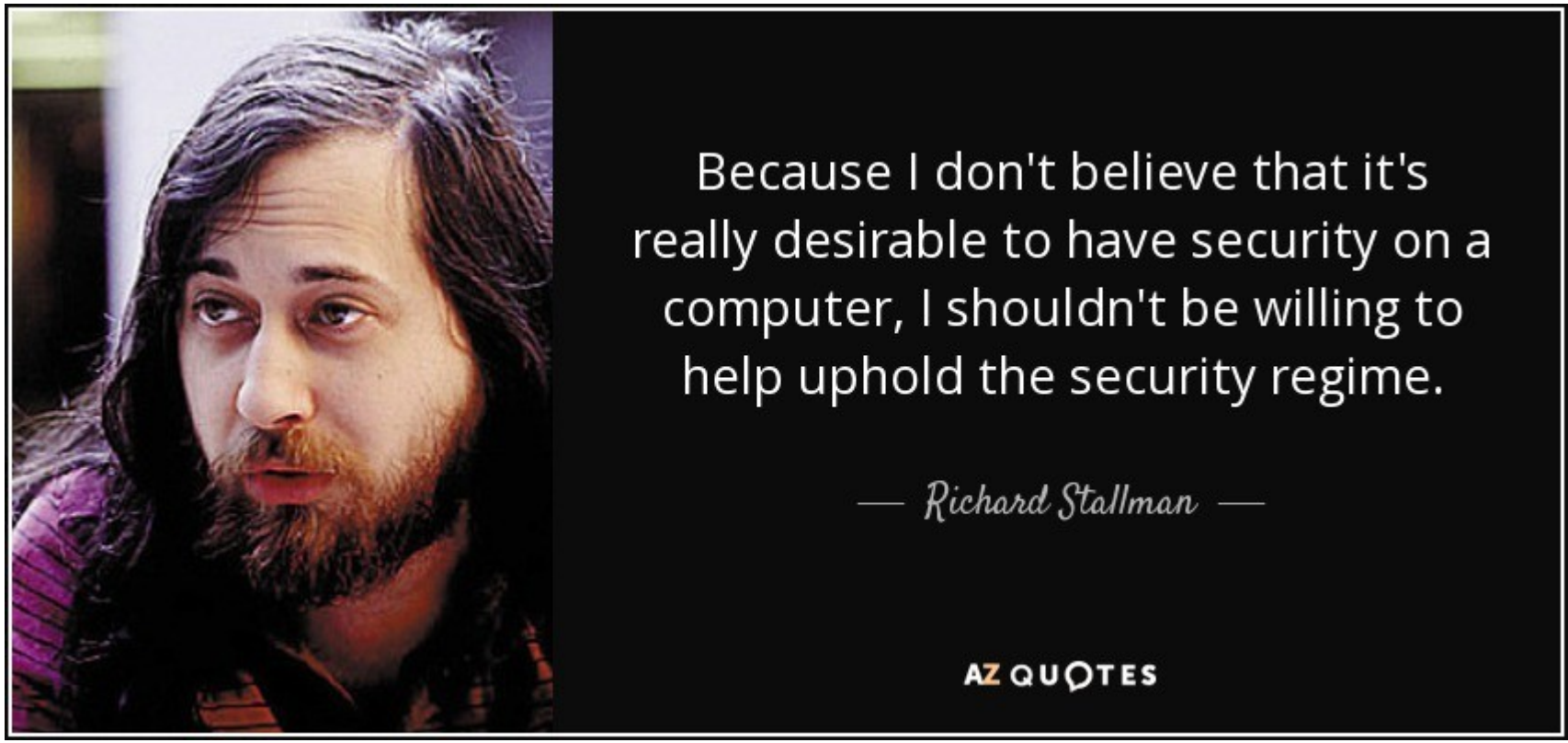
The Web was built for knowledge sharing & discovery

We collectively dreamt of the “Open Web”

Form follows function

Architecture follows vision

Remember how early computers did not have accounts, users, or passwords?



The invention of the Web echoed this sentiment.

Open knowledge sharing was to save the world.

Architecture matters.

Retrofitting security has limits.

And consequences.

Trust Service Providers

The best answer we have

TLS, DNSSEC… stacking band-aids

Security drove centralization

Resulting in rent-seeking

So now Let’s Encrypt is the single point of failure.

Unintended Consequences...

Platform & Surveillance Industry

Consequences of the underlying architecture in combination with networking effects and economies of scale

https://thesocialdilemma.com

It’s easy to think this is natural.

But it’s really not.

It’s by DESIGN.

More layers create more problems.

Yet, we keep adding layers.

Including for the most important personal data.

Whose health records might be available here?

https://georggreve.net/JcWmYPdD8azB1cZYaY6F

Paper is peer to peer.

Let’s design our systems to be more like paper.

What if we start building our applications on top of a security, privacy, authenticity layer that is decentralized?

Quest for European Sovereignty: Gaia-X

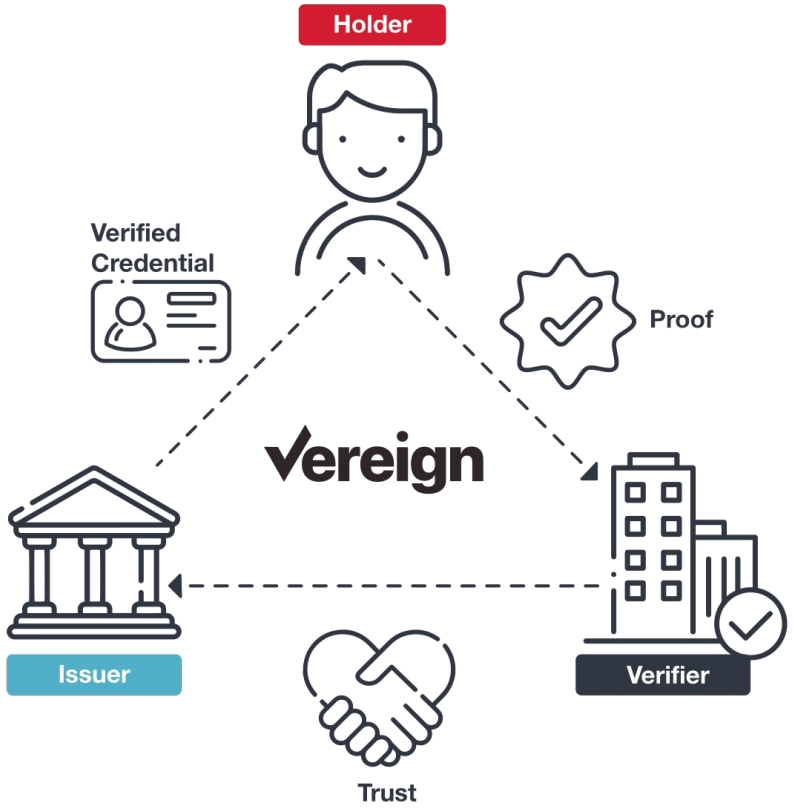
💡 Idea: Instead of rebuilding a broken design, let’s build a better architecture.

😒 Experts: ”This did not deliver the broken design I am used to. It has failed.”

💸 Reality: Death by committee, top-down, lobbyists, and some corruption.

But it led to some useful learnings and technology.

Decentralized Key Management with autonomous identifiers (AID) (also known as Self Sovereign Identity)



Translating SSI

Self → Interaction, peer to peer

Sovereign → Control, privacy

Identity → Authentic data and applications

Whose health records might be available here?

did:svdx:z6MknHKiY477mH97qryHv3zjuHaTLvBbbp6tHS5SvZv67uR4:
QmecqVGBxvW7gjffxmYTGFZNPmJcWmYPdD8azB1cZYaY6F

did:svdx

W3C DID documents

A truly decentralized identifier

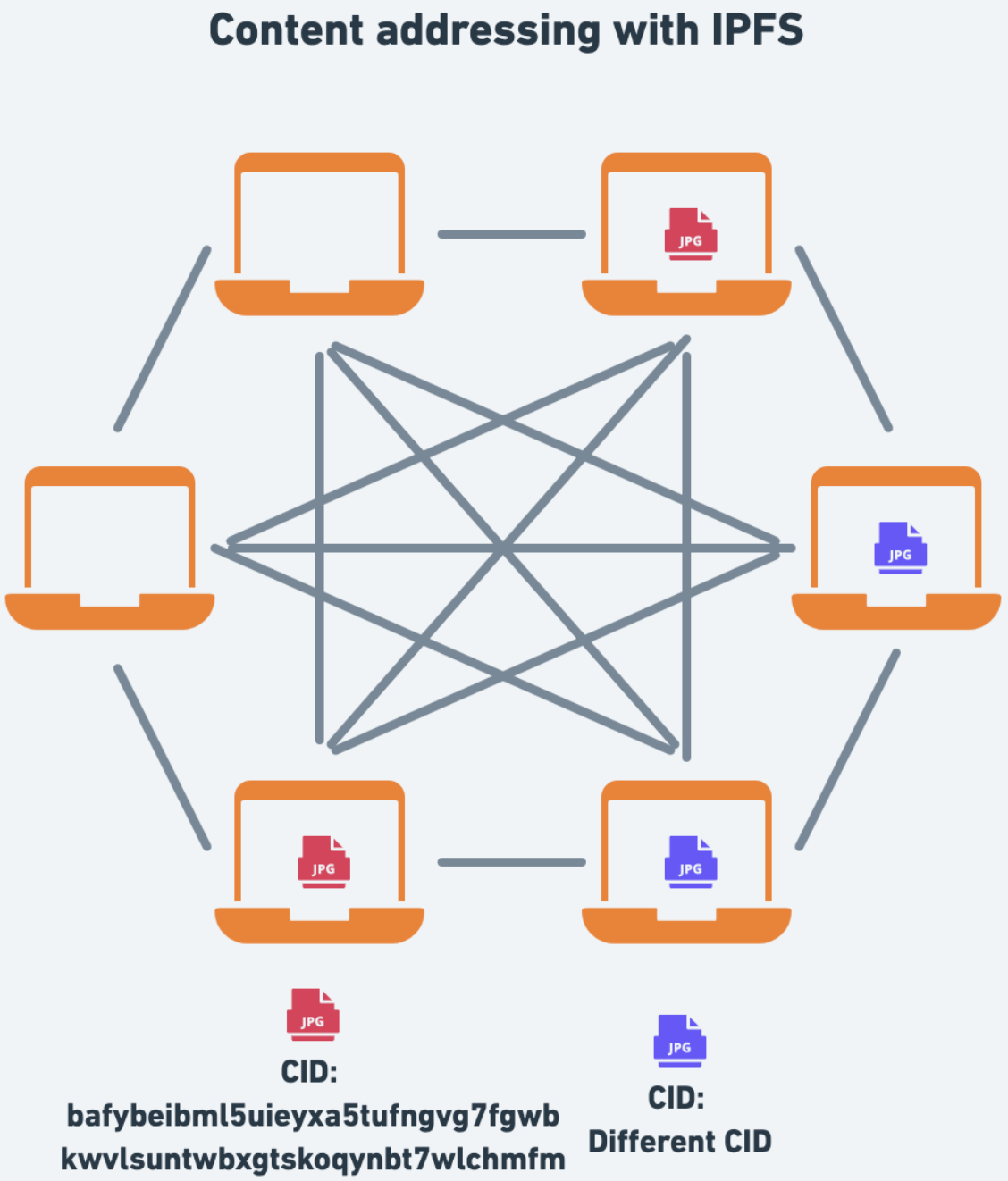
Built on IPFS

Decentralized Key Management

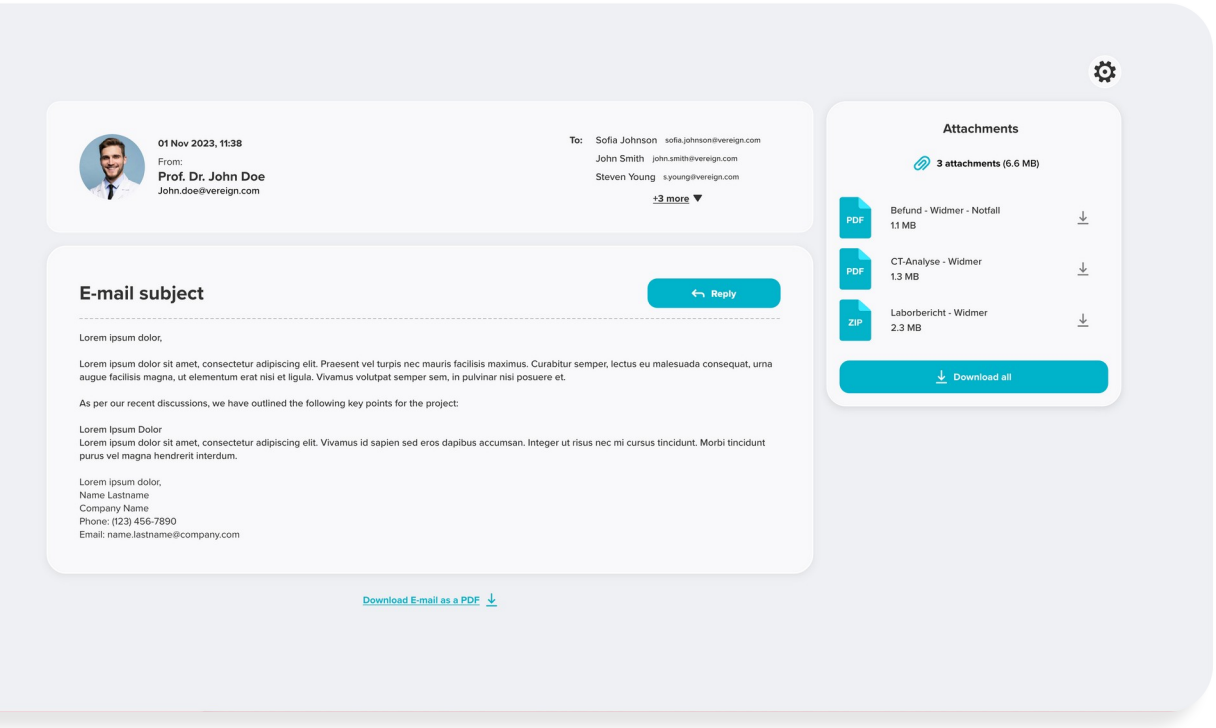
Meets the “paper” test

Just as easy as web, but better

Contributors welcome!



First application: SEAL



Secure, encrypted, private data transmission without installation

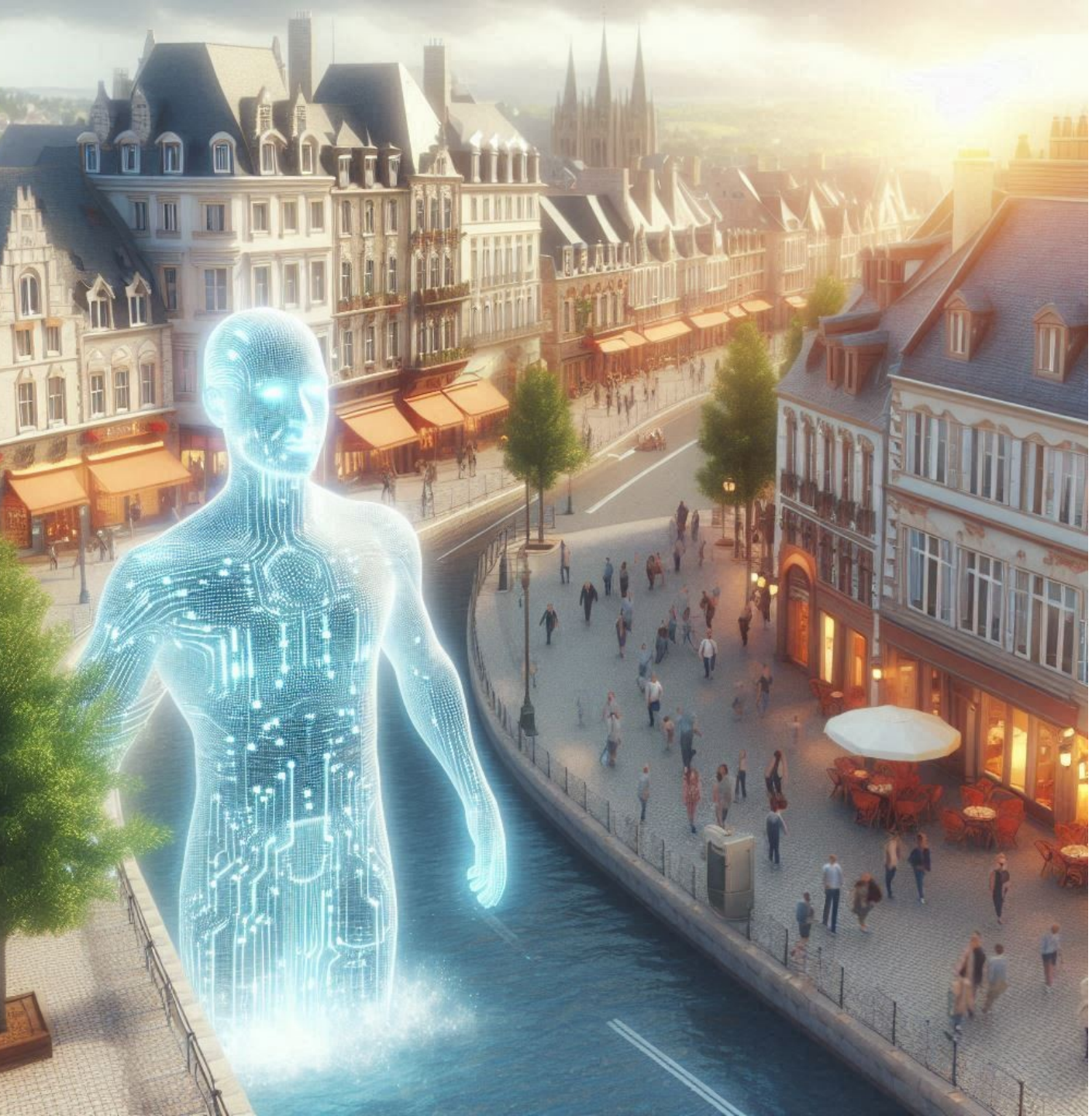
Still “looks” a lot like the web, by design

But moves all data & control to the edge

Bridge between Web & DKMS

Contains Verifiable Presentations

Allows for agent & wallet upgrade path



GOAL

Sovereign Data Exchange

For any kind of data

Peer to peer

Authenticity, security, privacy, control

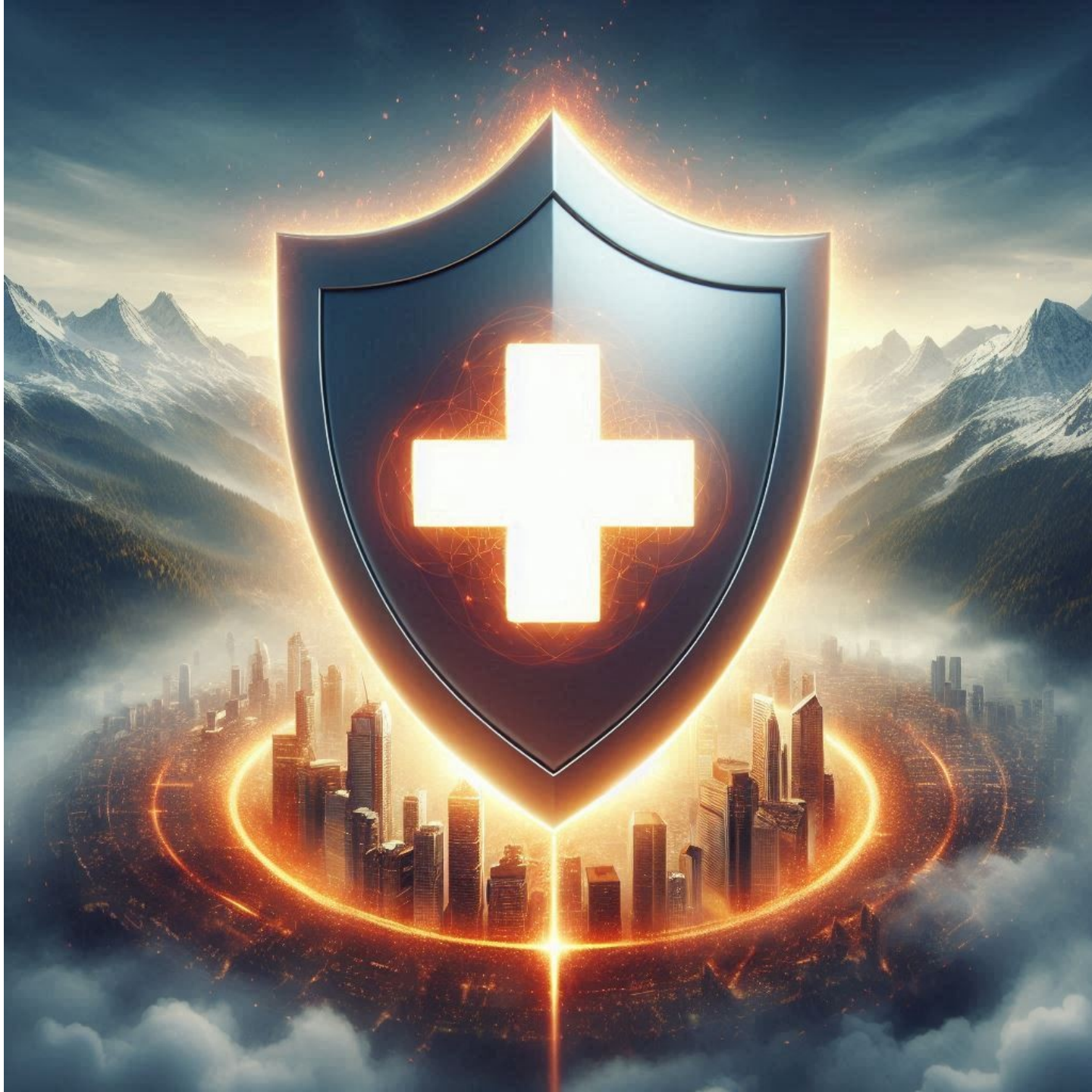
Mutual verification, as required

Integrated into Swiss eID

Perspective

Swiss CERN was the birth place for the Web...

... Swiss Healthcare is working to become the birth place for a new privacy & security first network of decentralized applications of authenticity.



THANK YOU!

contact @ Vereign .com

verifiable and self-sovereign

Vereign AG | Dammstrasse 16 | 6300 Zug
+41 41 541 50 63

images made using Microsoft Copilot

How does it work?

- The senders MTA routes messages for delivery via SEAL to the **SEAL engine**.
- The engine compresses, encrypts, and fragments the message. MIME parts are encoded recursively in the same way.
- The **first fragment** is encoded into a URL for the **Web Verification App**, all other fragments are stored into the Interplanetary File System (IPFS).
- The **first fragment URL** can be transmitted to the recipient via text message, messenger, or email in form of a link or QR code.
- The **recipient** can open this URL on any device to access the message, loading the Web Verification App and all fragments from IPFS.
- If authentication is used, the Web Verification App handles authentication of the recipient against the **Key & Auth Service** to decode the encryption key to render the message.