SAFE Network

Evolving the Internet in Rust

Paige Peterson - Communications, MaidSafe

Rust Berlin, 20 August 2015

OUR RUST STORY

The SAFE Network is an open source, open development project researched and developed by MaidSafe.

Not quite “in production”... yet...

Python for initial research stages, C++ for production libraries...
...and then Rust happened.

From just the founder experimenting to going all in.

ALL IN ON RUST

10x smaller codebase (and that’s after the initial refactoring)

Quicker iteration on design and development.

...we cannot short circuit / alter the system of tests/build etc. so we need to work harder to test really. Otherwise we go back to the land of spaghetti and complexity where nobody knows what is happening. We are not going back there ;)

I still remember my first week with Rust, when [we] started coding the Crust library. We had just one test and I was very sure that we wouldn’t be able to get any library to work on the very first attempt. However, I was very happy to be proven wrong!

ALL IN ON RUST

Empowerment of front end developers

Those initially hired for application development - now core developers

Original core developers also welcoming the change

The problem is a very good C++ programmer would be able to write code which doesn't produce any conflicts when it's running on multithreaded systems but it requires being awake and you cannot program in the late hours of night... it requires a lot of mental effort and the help of Rust is exactly that: the computer is always there watching over you.

ALL IN ON RUST

Outside contributors and mixing communities

Expanding core development and understanding

MaidSafe code bounty program

Application ecosystem

Appreciation and alignment with Rust core development

THE NETWORK

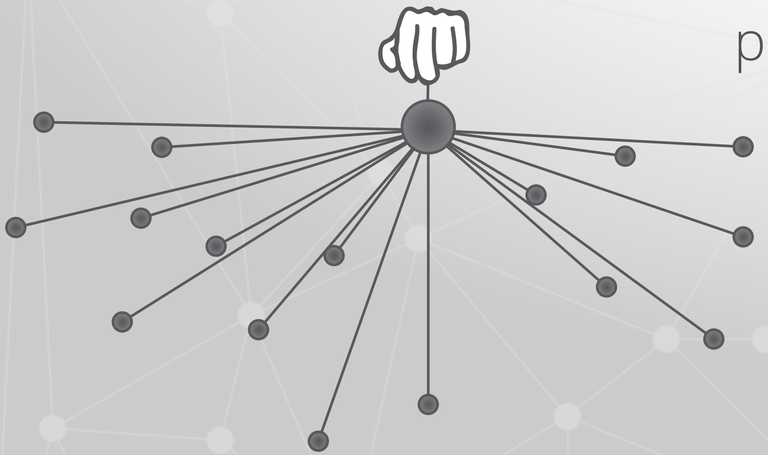
Massive array of internet disks, secure access for everyone

The SAFE Network is an alternative Internet infrastructure and development platform for decentralising storage and communication where privacy and security are priority.

serverless

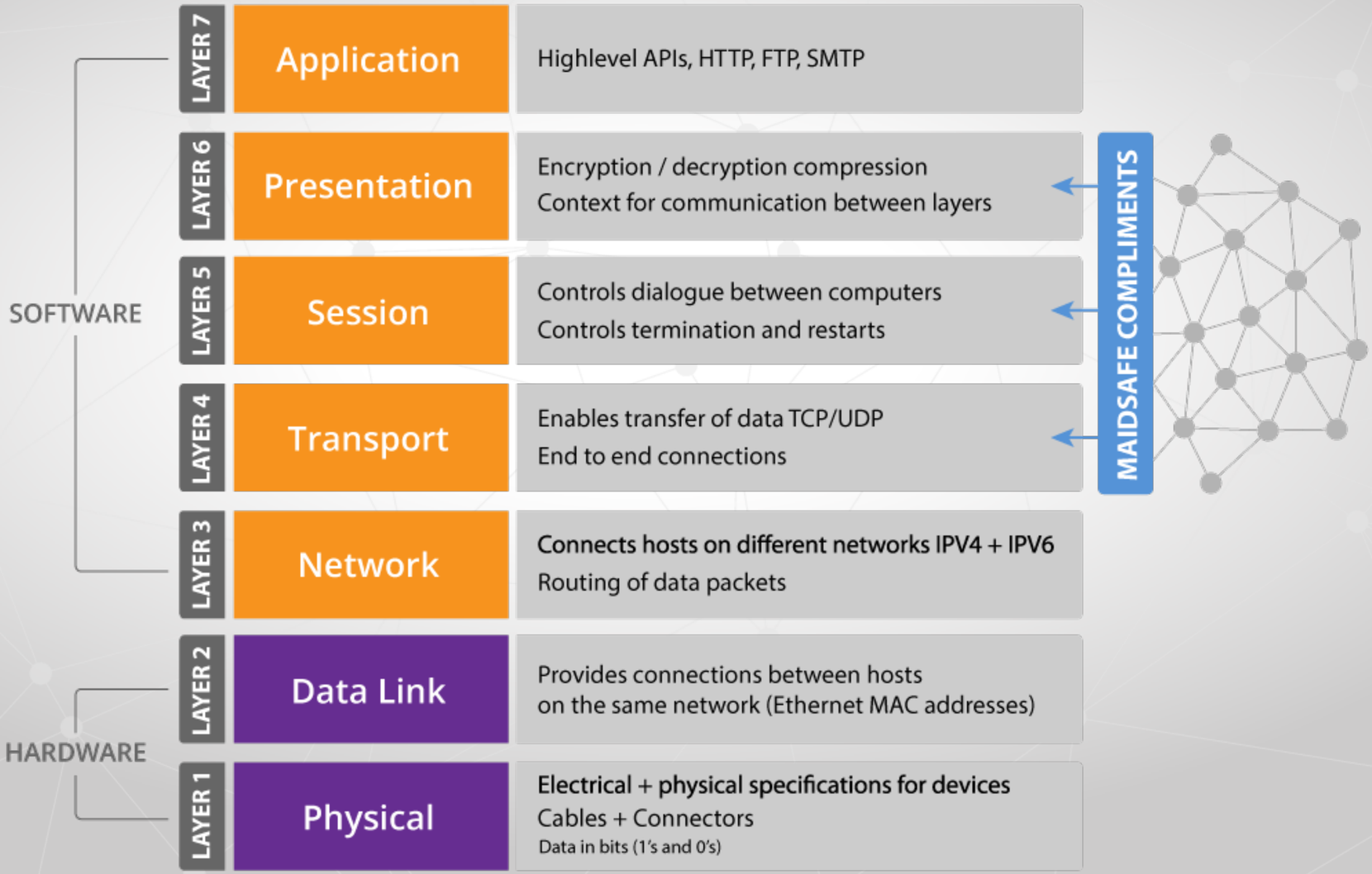
self-encryption, self-authentication secured

personal, private or public data

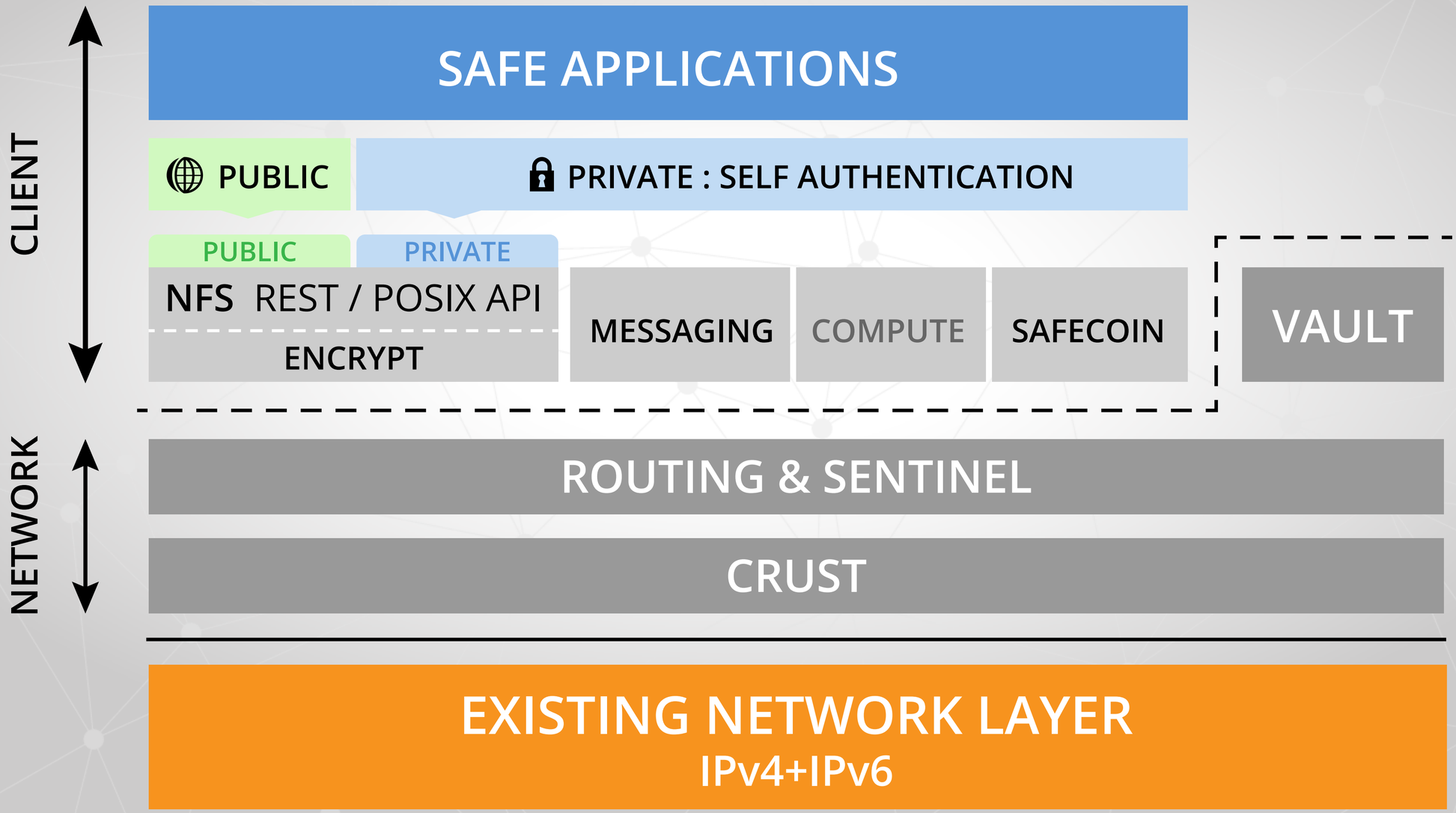


autonomous

STACK

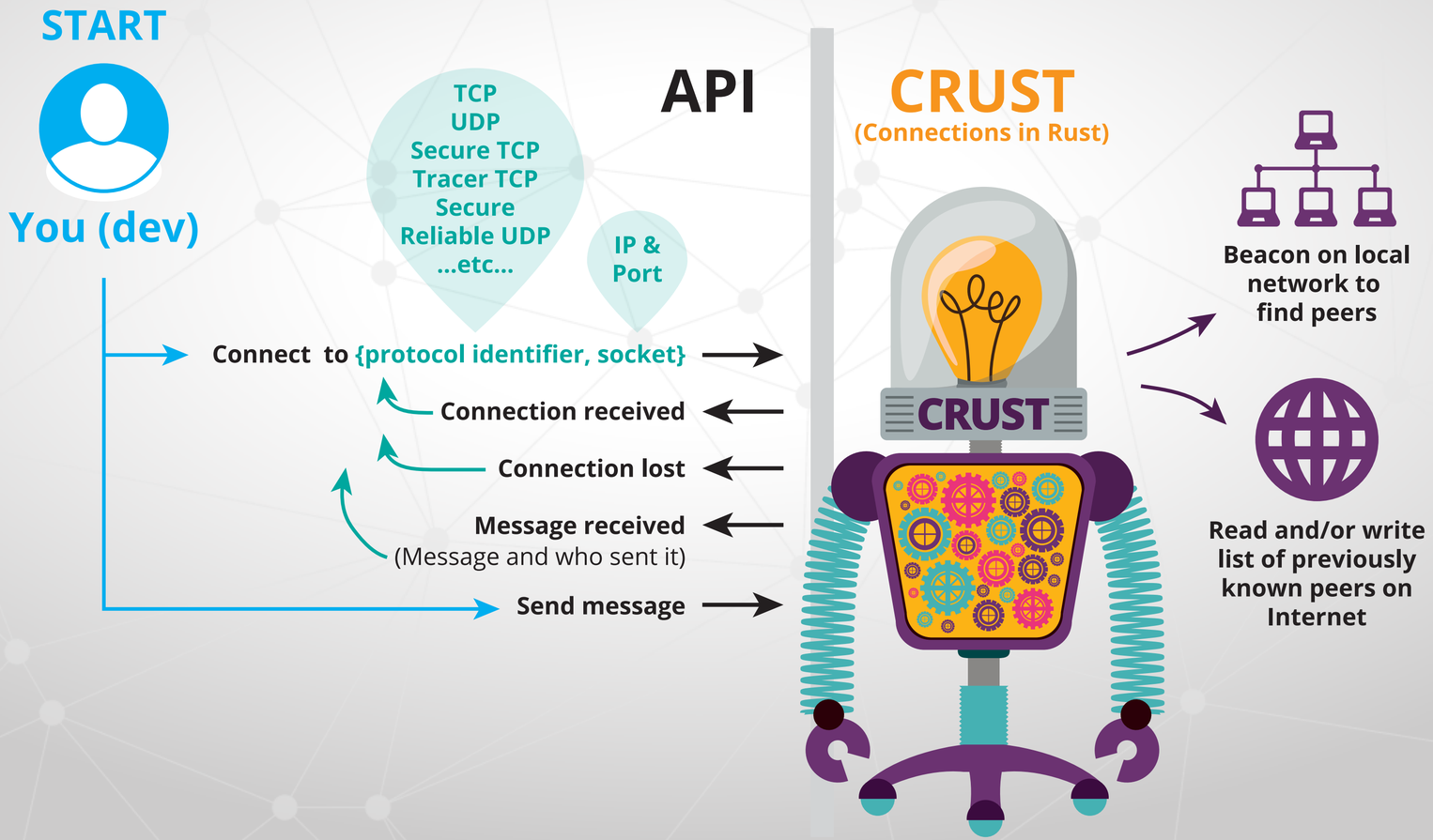

SOFTWARE
HARDWARE
LAYER 7
Application
Highlevel APIs, HTTP, FTP, SMTP
LAYER 6
Presentation
Encryption / decryption compression
Context for communication between layers
LAYER 5
Session
Controls dialogue between computers
Controls termination and restarts
LAYER 4
Transport
Enables transfer of data TCP/UDP
End to end connections
LAYER 3
Network
Connects hosts on different networks IPV4 + IPV6
Routing of data packets
LAYER 2
Data Link
Provides connections between hosts
on the same network (Ethernet MAC addresses)
LAYER 1
Physical
Electrical + physical specifications for devices
Cables + Connectors
Data in bits (1's and 0's)
MAIDSAFE COMPLIMENTS

STACK


CLIENT
SAFE APPLICATIONS
PUBLIC
PRIVATE : SELF AUTHENTICATION
PUBLIC
PRIVATE
NFS REST / POSIX API
ENCRYPT
MESSAGING
COMPUTE
SAFECOIN
VAULT
NETWORK
ROUTING & SENTINEL
CRUST
EXISTING NETWORK LAYER
IPv4+IPv6

CRUST

PEER 2 PEER NETWORKING MADE EASY



ROUTING + SENTINEL + VAULTS

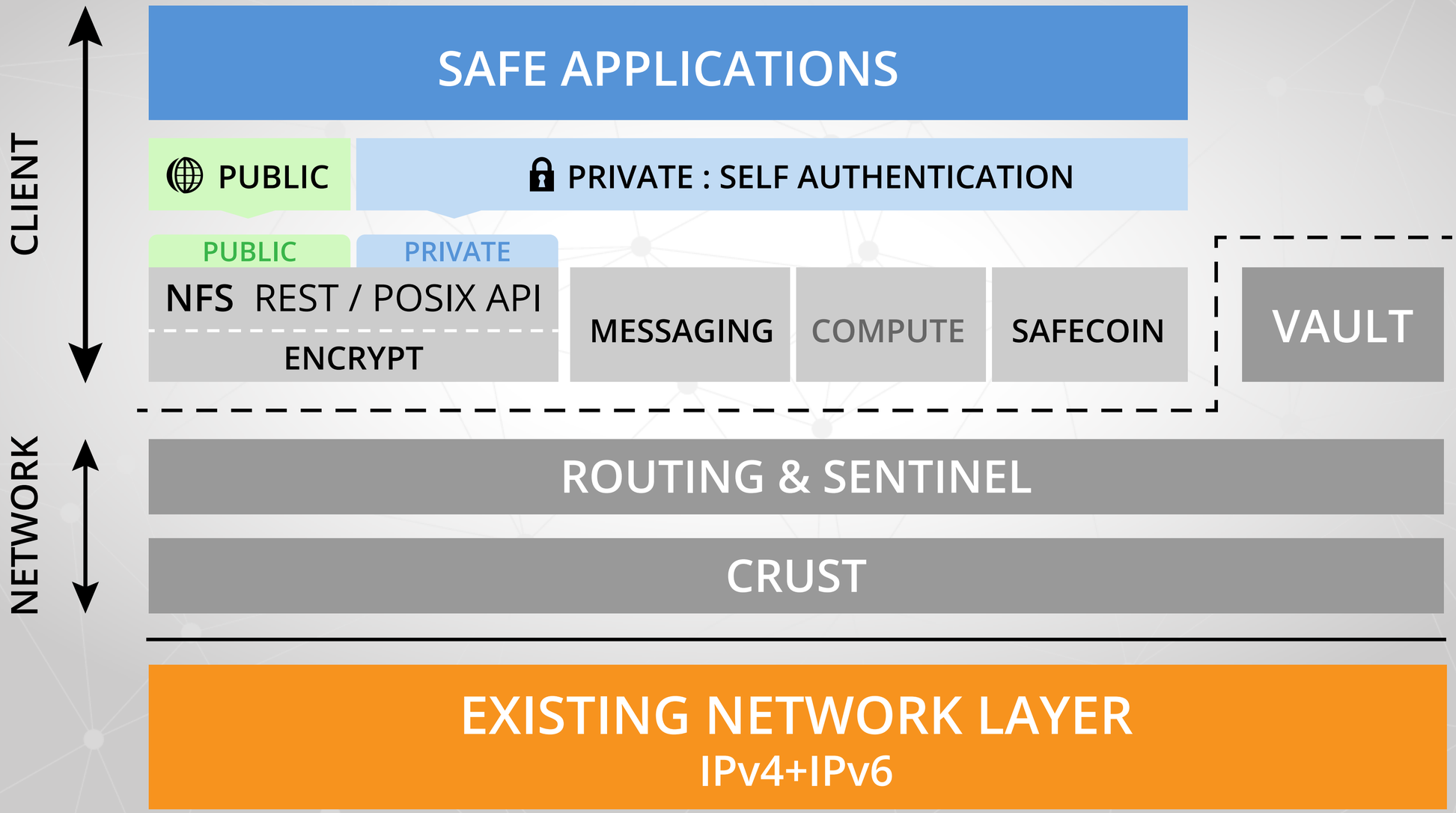
Modified Kademlia DHT with public key infrastructure

In addition to storing data, vaults take on various roles for routing data within the network

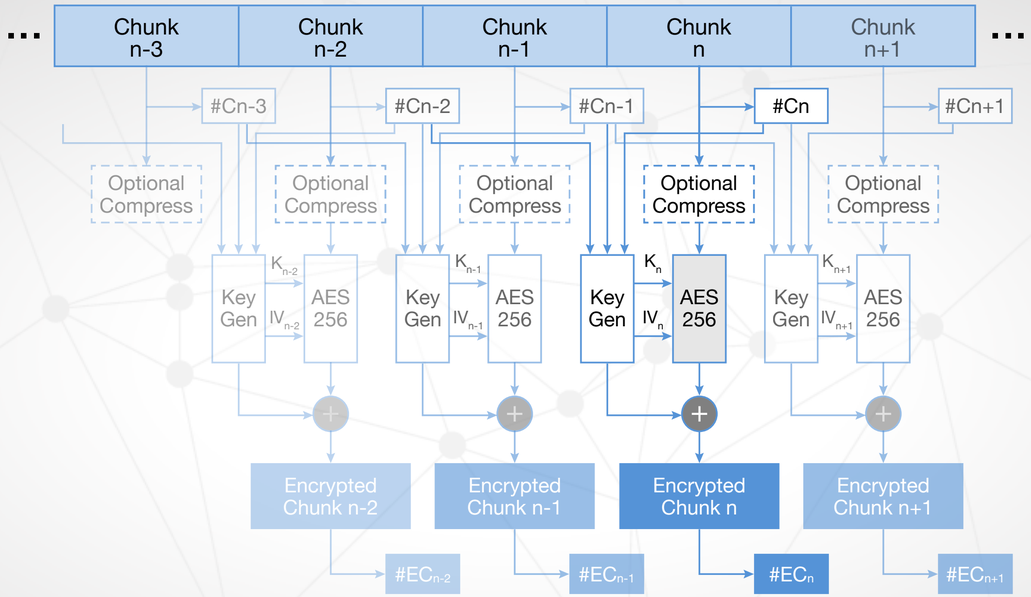
Language of the network
-consensus based authority to perform actions for data integrity
-role assignment via messages (data type id, persona id)

Network-layer ranking system to protect against uncooperative nodes

STACK


CLIENT
SAFE APPLICATIONS
PUBLIC
PRIVATE : SELF AUTHENTICATION
PUBLIC
PRIVATE
NFS REST / POSIX API
ENCRYPT
MESSAGING
COMPUTE
SAFECOIN
VAULT
NETWORK
ROUTING & SENTINEL
CRUST
EXISTING NETWORK LAYER
IPv4+IPv6

SELF-ENCRYPTION



Outputs encrypted data chunks and a data map

Identity-less, secured data

SELF-AUTHENTICATION

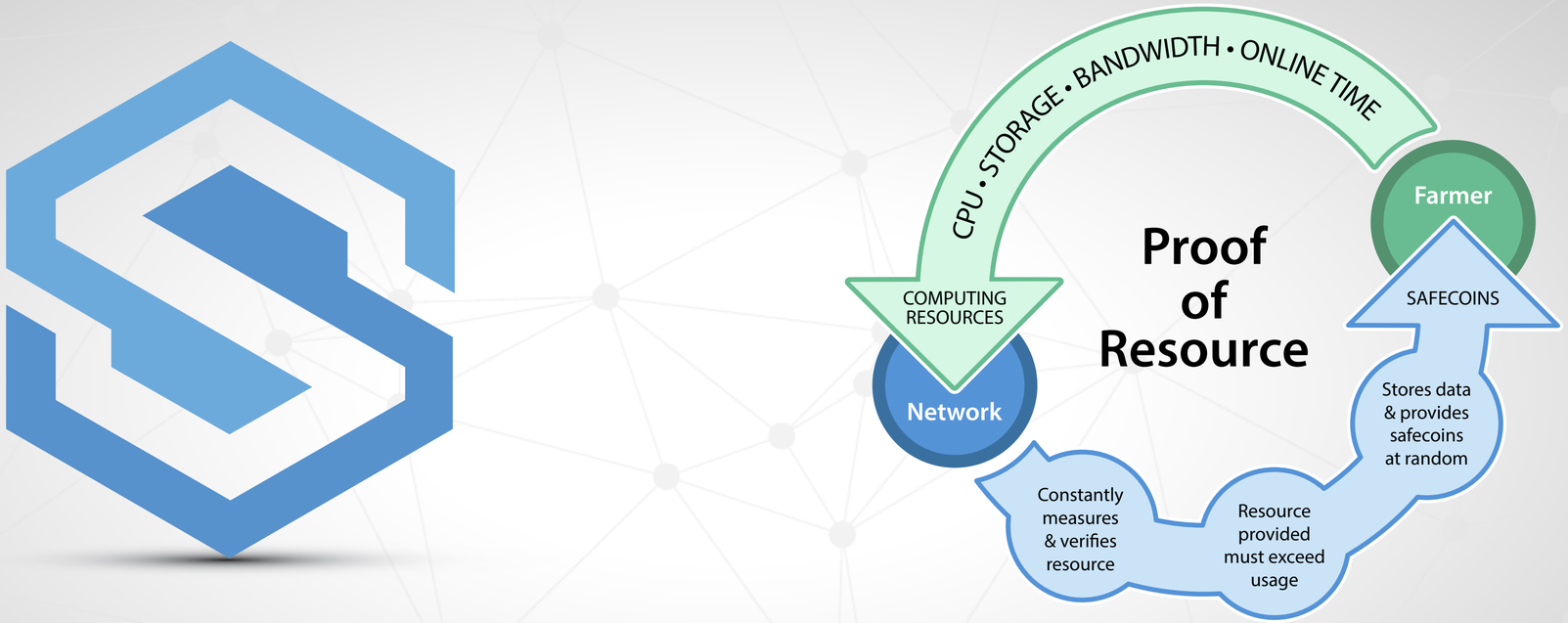
Account creation and verification from any computer connected to the Internet without third-party authority

Generate unique access ID to be stored on the network associated with a passport containing all uploaded data map keys

Network generates unique login token for each request (whether valid or not)

Unified login to all third-party applications existing on the SAFE Network

ECOSYSTEM



Proof of resource to balance consumption and availability

Safecoin is a built in asset which is acquired by providing resources to the network and required for putting new data to the network

DEVELOPER LINKS

github.com/maidsafe

maidsafe.atlassian.net

crates.io/search?q=maidsafe

forum.safenetwork.io

maidsafe.net/documents

paige.peterson@maidsafe.net
@ioptio
@MaidSafe

TECHNICAL OVERVIEWS

CRUST - https://www.youtube.com/watch?v=Zdt_8c1bWDk

Routing - https://www.youtube.com/watch?v=ao7ODM7Ndd8

Sentinel - https://www.youtube.com/watch?v=VwV1yar6PMw

Vault - https://www.youtube.com/watch?v=sKI8i4P66IA