Distributing Authenticated Mappings

Keys, policies, binaries, and more

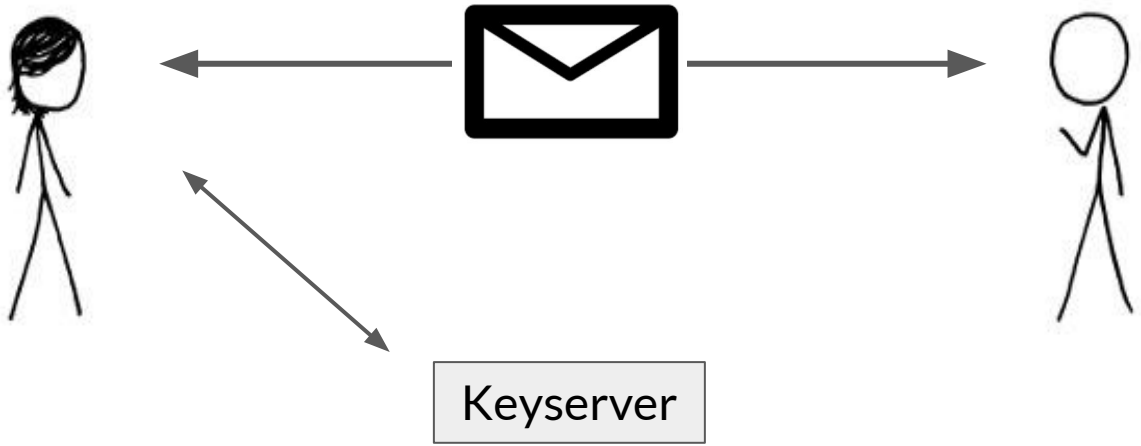
Sydney Li[1], Colin Man[2], Jean-Luc Watson[2]

[1]Electronic Frontier Foundation, [2]Stanford University

Things that are broken on the internet

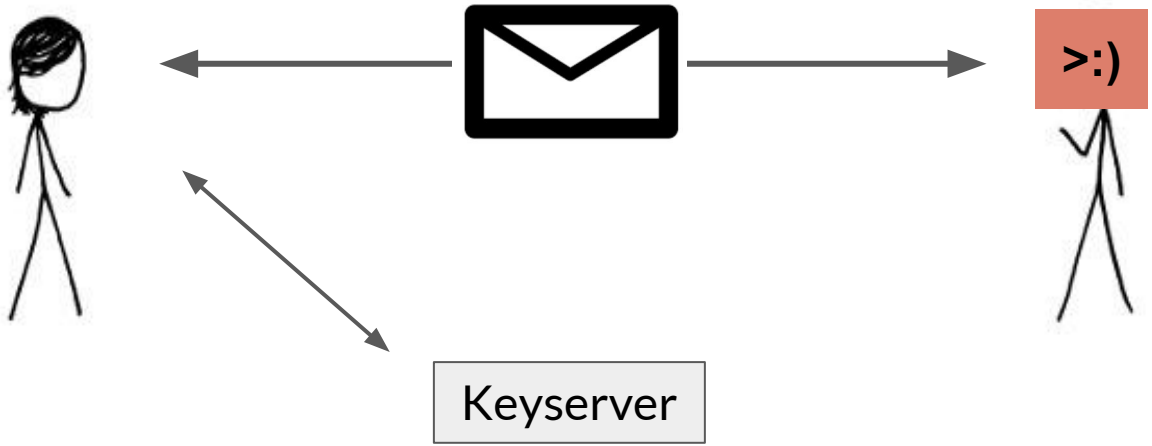
Things that are broken on the internet

- Two people having a private conversation over encrypted email



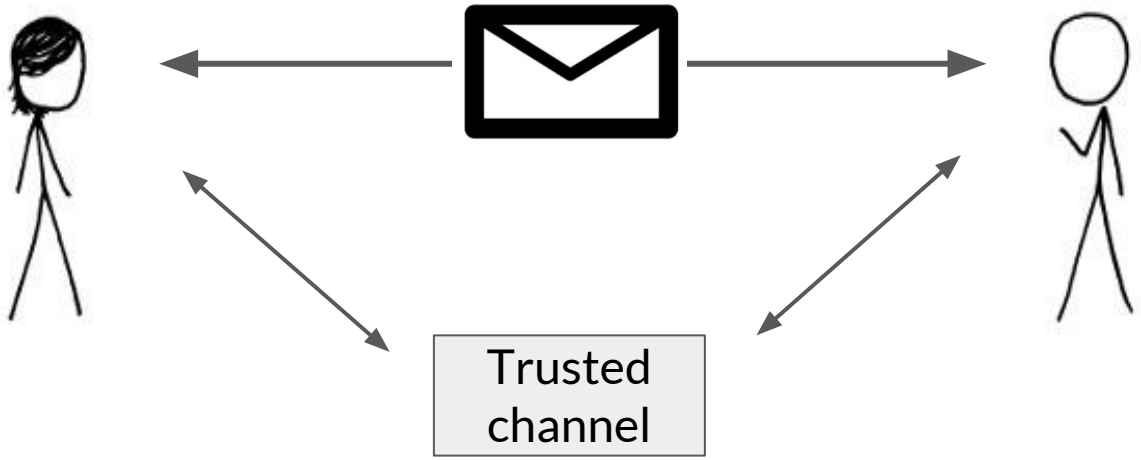
Things that are broken on the internet

- Two people having a private conversation over encrypted email
  Susceptible to MITM the key distribution source is untrusted



Things that are broken on the internet

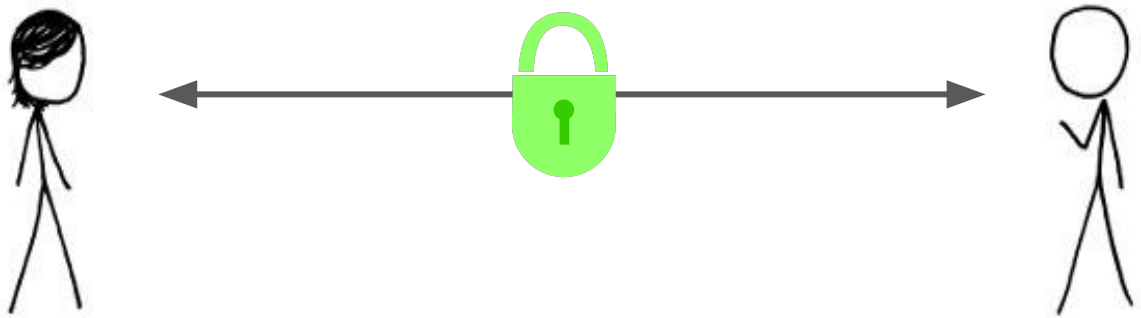
- Two people having a private conversation over encrypted email
  Susceptible to MITM the key distribution source is untrusted



Things that are broken on the internet

- Two people having a private conversation over encrypted email
  Susceptible to MITM the key distribution source is untrusted

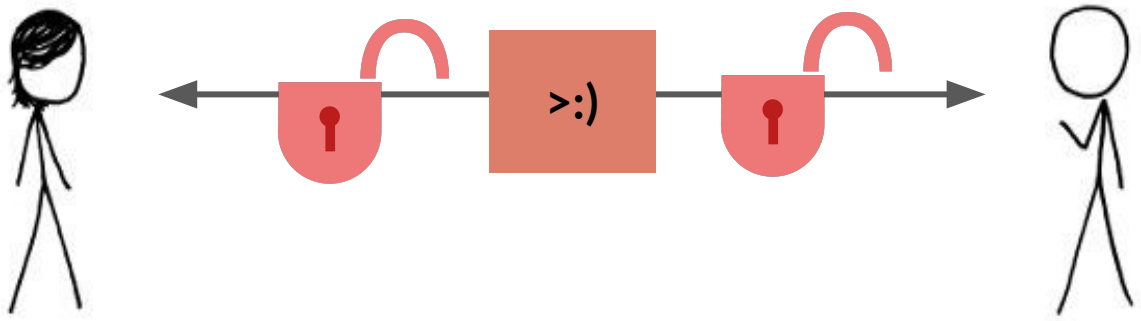
- Small website operator trying to provide secure internet service



Things that are broken on the internet

- Two people having a private conversation over encrypted email
  Susceptible to MITM the key distribution source is untrusted

- Small website operator trying to provide secure internet service
  Susceptible to downgrade attacks until they get onto HSTS-preload



Things that are broken on the internet

- Two people having a private conversation over encrypted email
  Susceptible to MITM the key distribution source is untrusted

- Small website operator trying to provide secure internet service
  Susceptible to downgrade attacks until they get onto HSTS-preload

- DNS
  DNSSEC yet to see widespread adoption

What the Internet needs

Authenticated mappings!

Problem

Name mappings

Policy mappings

Certificate mappings

Binary distribution

Public key mappings

What the Internet needs

Authenticated mappings!

Problem	**Solution**
Name mappings	DNS (+ DNSSEC)
Policy mappings	HSTS preload lists
Certificate mappings	CA trust chains + CT
Binary distribution	package lists / bin. transparency
Public key mappings	Trusted keyservers

What the Internet needs

Authenticated mappings!

Problem	**Solution**
Name mappings	DNS (+ DNSSEC)
Policy mappings	HSTS preload lists
Certificate mappings	CA trust chains + CT
Binary distribution	package lists / bin. transparency
Public key mappings	Trusted keyservers

Many solutions based on *incorrect assumptions of trust*, *aren't scalable*, or *aren't generalizable*.

Generalized Mappings

Instead, can we derive a scalable solution that will work for any mapping?

Generalized Mappings

Instead, can we derive a scalable solution that will work for any mapping?

Solution: infrastructure for a global state database

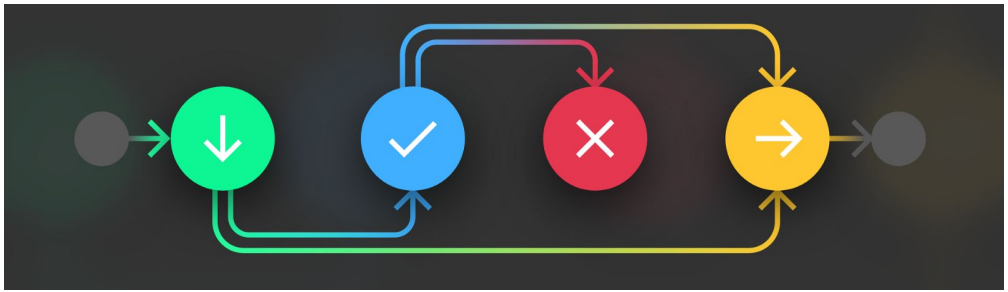
- Append-only

Generalized Mappings

Instead, can we derive a scalable solution that will work for any mapping?

Solution: infrastructure for a global state database

- Append-only
- Well-formed transitions

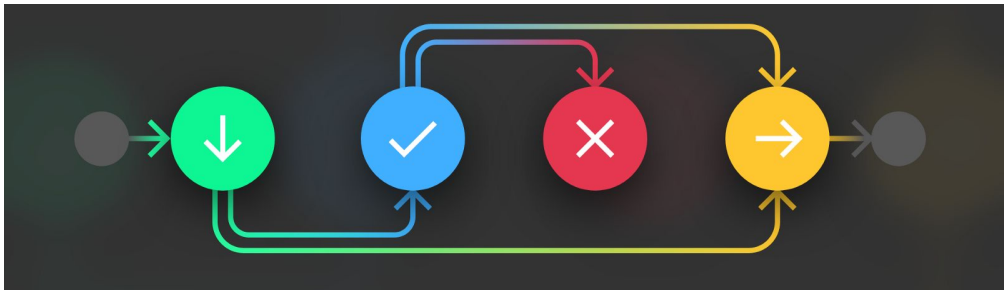


Generalized Mappings

Instead, can we derive a scalable solution that will work for any mapping?

Solution: infrastructure for a global state database

- Append-only
- Well-formed transitions
- Transparent



Option 1: Bootstrapping Certificate Transparency

CT works well -- CAs cooperate!

- Let’s bootstrap binary transparency?

Option 1: Bootstrapping Certificate Transparency

CT works well -- CAs cooperate!

- Let's bootstrap binary transparency?
  - Sure! Log binary hash into the CT log

Option 1: Bootstrapping Certificate Transparency

CT works well -- CAs cooperate!

- Let's bootstrap binary transparency?
  - Sure! Log binary hash into the CT log

Problems

- Why should CAs care about your binaries?

Option 1: Bootstrapping Certificate Transparency

CT works well -- CAs cooperate!

- Let's bootstrap binary transparency?
  - Sure! Log binary hash into the CT log

Problems

- Why should CAs care about your binaries?
- How do CAs know how to enforce semantics for binaries?

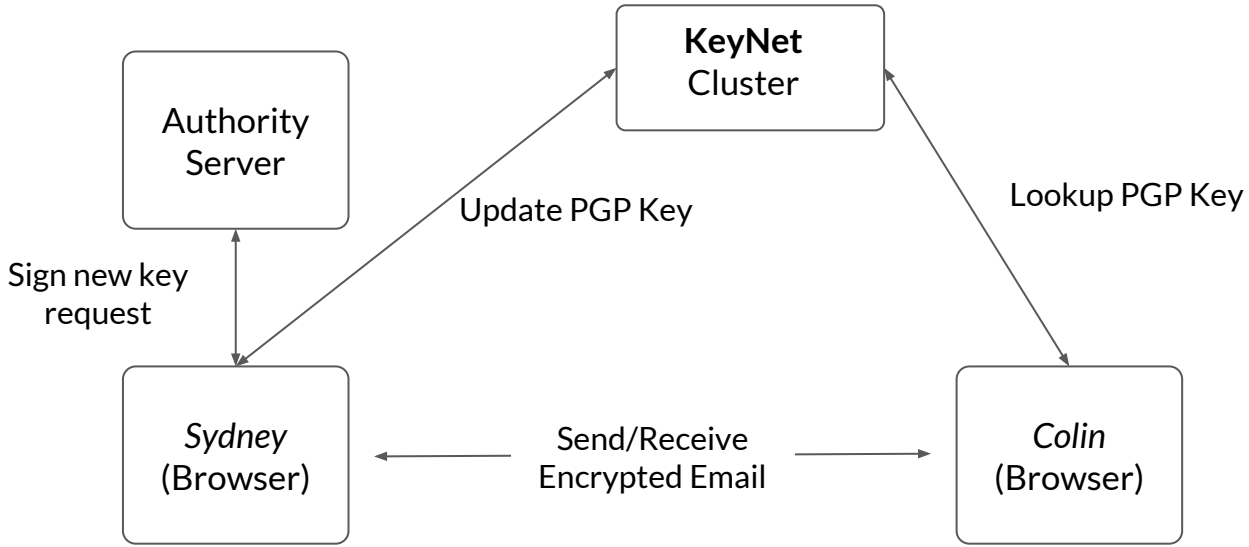
Option 2: Byzantine Fault Tolerant Cluster

Set up a number of PBFT nodes and distribute mapping database.

- Enforce append-only and transition semantics via traditional consensus
- **KeyNet**

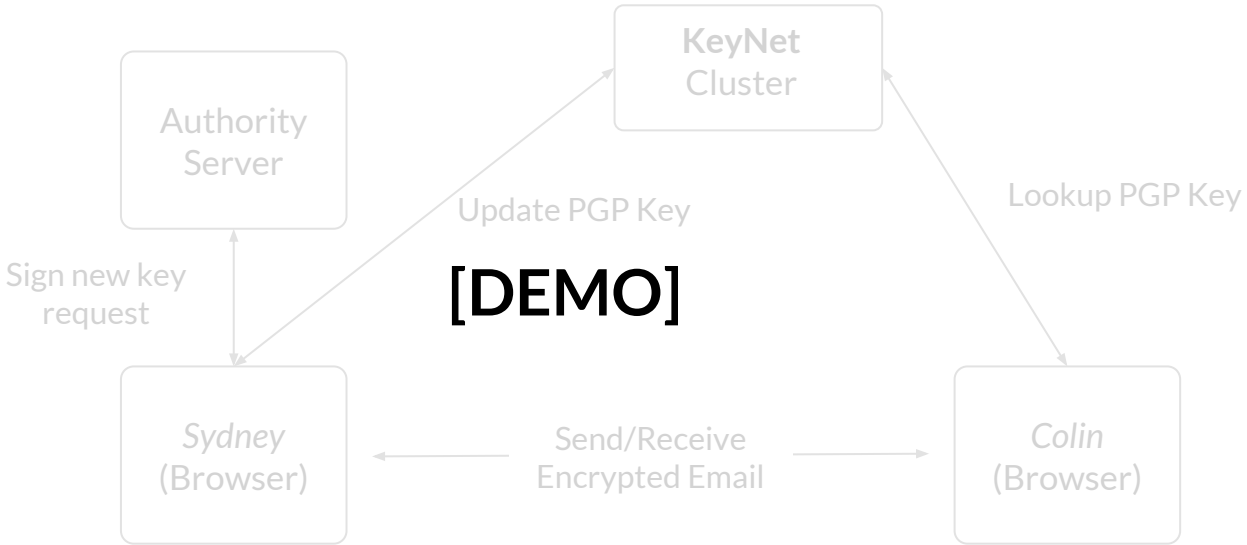
KeyNet

- Distributed key-value store for OpenPGP-standard keys
- Rerouted Mailvelope on the front end to sign and send emails



KeyNet

- Distributed key-value store for OpenPGP-standard keys
- Rerouted Mailvelope on the front end to sign and send emails



Option 2: Byzantine Fault Tolerant Cluster

Set up a number of PBFT nodes and distribute mapping database.

- Enforce append-only and transition semantics via traditional consensus
- No difference to the end user!

Option 2: Byzantine Fault Tolerant Cluster

Set up a number of PBFT nodes and distribute mapping database.

- Enforce append-only and transition semantics via traditional consensus
- No difference to the end user!

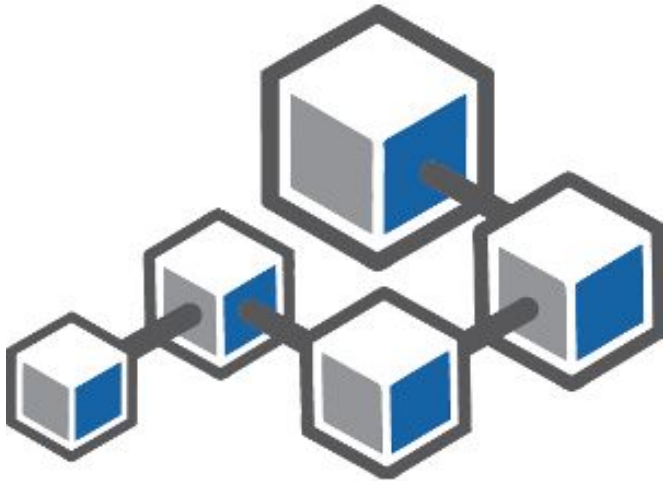
Problem: limited participation

- Uniform set of incentives undermines security

Option 3a: Proof-of-Work

Gets us almost there!

- We can create an append-only log
- Anyone can participate and enforce transition semantics
- Maturing technology



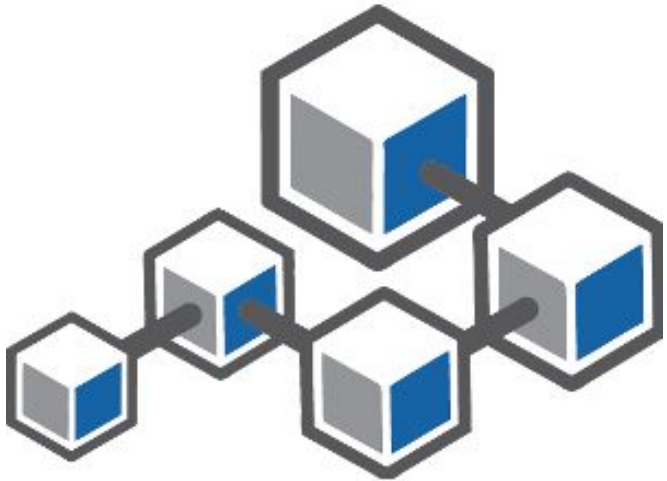
Option 3a: Proof-of-Work

Gets us almost there!

- We can create an append-only log
- Anyone can participate and enforce transition semantics
- Maturing technology

Problems

- No accountability
- Trust is tied to hash power
- Environmental cost



Option 3b: Proof-of-Stake

Even better!

- We can create append-only logs
- Anyone can participate and enforce transition semantics
- Environmentally-friendly

Option 3b: Proof-of-Stake

Even better!

- We can create append-only logs
- Anyone can participate and enforce transition semantics
- Environmentally-friendly

Problem: Yet another incentive mismatch: trust is tied to money

Option 4: Federated Byzantine Agreement

Combines safety guarantees of BFT with open membership of PoW/S schemes

- Allows actors with different interests to participate and enforce transition semantics
- Accountability

Option 4: Federated Byzantine Agreement

Combines safety guarantees of BFT with open membership of PoW/S schemes

- Allows actors with different interests to participate and enforce transition semantics
- Accountability

Trust in the network is tied to real-world relationships

- Rely on interdependence to ensure security
- Malicious behavior risks reputation

Open Problems

Bootstrapping and interoperability

Privacy

Scalable data structures

Defining well-formed updates (contract language)

Next Steps

How can DIN help?

- Infrastructure for authenticated mappings is moving forward independently, in parallel
- Generalize solution
    - diversity of incentives = everyone securing each other’s services

Next Steps

How can DIN help?

- Infrastructure for authenticated mappings is moving forward independently, in parallel
- Generalize solution
    - diversity of incentives = everyone securing each other's services

Let's standardize the way we distribute trust at scale:

1. Specs for describing transition semantics
2. A distributed protocol for enforcing these rules

Questions?