Components for Building Secure Decentralized Networks

Christian Grothoff

Technische Universität München

26.11.2013

“Never doubt your ability to change the world.” –Glenn Greenwald
Where We Are
Where We Are
My Research and Development Agenda

Make decentralized systems:

- Faster, more scalable
- Easier to develop, deploy and use
- Easier to evolve and extend
- Secure (privacy-preserving, censorship-resistant, available, ...)

by:

- designing secure network protocols
- implementing secure software following and evolving best practices
- creating tools to support developers
## Our Vision

### Internet

<table>
<thead>
<tr>
<th>Protocol/Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google/Facebook</td>
</tr>
<tr>
<td>DNS/X.509</td>
</tr>
<tr>
<td>TCP/UDP</td>
</tr>
<tr>
<td>IP/BGP</td>
</tr>
<tr>
<td>Ethernet</td>
</tr>
<tr>
<td>Phys. Layer</td>
</tr>
</tbody>
</table>
## Our Vision

**Internet**

<table>
<thead>
<tr>
<th>Protocols</th>
<th>Datapaths</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google/Facebook</td>
<td></td>
</tr>
<tr>
<td>DNS/X.509</td>
<td></td>
</tr>
<tr>
<td>TCP/UDP</td>
<td></td>
</tr>
<tr>
<td>IP/BGP</td>
<td></td>
</tr>
<tr>
<td>Ethernet</td>
<td></td>
</tr>
<tr>
<td>Phys. Layer</td>
<td></td>
</tr>
<tr>
<td>HTTPS/TCP/WLAN/...</td>
<td></td>
</tr>
</tbody>
</table>
### Our Vision

#### Internet

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google/Facebook</td>
<td>DNS/X.509</td>
</tr>
<tr>
<td></td>
<td>TCP/UDP</td>
</tr>
<tr>
<td></td>
<td>IP/BGP</td>
</tr>
<tr>
<td></td>
<td>Ethernet</td>
</tr>
<tr>
<td></td>
<td>Phys. Layer</td>
</tr>
</tbody>
</table>

...
Our Vision

### Internet

<table>
<thead>
<tr>
<th>Google/Facebook</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS/X.509</td>
<td></td>
</tr>
<tr>
<td>TCP/UDP</td>
<td></td>
</tr>
<tr>
<td>IP/BGP</td>
<td></td>
</tr>
<tr>
<td>Ethernet</td>
<td></td>
</tr>
<tr>
<td>Phys. Layer</td>
<td></td>
</tr>
</tbody>
</table>

|  | $R^5N$ DHT |
|  | CORE (ECDHE+AES) |
|  | HTTPS/TCP/WLAN/... |


### Our Vision

#### Internet

<table>
<thead>
<tr>
<th>Google/Facebook</th>
<th>Mesh (ECDHE+AES)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS/X.509</td>
<td>$R^5N$ DHT</td>
</tr>
<tr>
<td>TCP/UDP</td>
<td>CORE (ECDHE+AES)</td>
</tr>
<tr>
<td>IP/BGP</td>
<td>HTTPS/TCP/WLAN/...</td>
</tr>
<tr>
<td>Ethernet</td>
<td></td>
</tr>
<tr>
<td>Phys. Layer</td>
<td></td>
</tr>
</tbody>
</table>
# Our Vision

## Internet

<table>
<thead>
<tr>
<th>Google/Facebook</th>
<th>GNU Name System</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS/X.509</td>
<td>Mesh (ECDHE+AES)</td>
</tr>
<tr>
<td>TCP/UDP</td>
<td>$R^5N$ DHT</td>
</tr>
<tr>
<td>IP/BGP</td>
<td>CORE (ECDHE+AES)</td>
</tr>
<tr>
<td>Ethernet</td>
<td>HTTPS/TCP/WLAN/...</td>
</tr>
<tr>
<td>Phys. Layer</td>
<td></td>
</tr>
</tbody>
</table>
## Our Vision

### Internet

<table>
<thead>
<tr>
<th>Google/Facebook</th>
<th>RegEx/PSYC</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS/X.509</td>
<td>GNU Name System</td>
</tr>
<tr>
<td>TCP/UDP</td>
<td>Mesh (ECDHE+AES)</td>
</tr>
<tr>
<td>IP/BGP</td>
<td>( R^5N ) DHT</td>
</tr>
<tr>
<td>Ethernet</td>
<td>CORE (ECDHE+AES)</td>
</tr>
<tr>
<td>Phys. Layer</td>
<td>HTTPS/TCP/WLAN/...</td>
</tr>
</tbody>
</table>
### Our Vision

#### Internet

<table>
<thead>
<tr>
<th>Google/Facebook</th>
<th>DNS/X.509</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP/UDP</td>
<td></td>
</tr>
<tr>
<td>IP/BGP</td>
<td></td>
</tr>
<tr>
<td>Ethernet</td>
<td></td>
</tr>
<tr>
<td>Phys. Layer</td>
<td></td>
</tr>
</tbody>
</table>

#### GNUnet

<table>
<thead>
<tr>
<th>RegEx/PSYC</th>
</tr>
</thead>
<tbody>
<tr>
<td>GNU Name System</td>
</tr>
<tr>
<td>Mesh (ECDHE+AES)</td>
</tr>
<tr>
<td>$R^5 N$ DHT</td>
</tr>
<tr>
<td>CORE (ECDHE+AES)</td>
</tr>
<tr>
<td>HTTPS/TCP/WLAN/...</td>
</tr>
</tbody>
</table>
The GNU Name System

Properties of GNS

▶ Decentralized name system with secure memorable names
▶ Delegation used to achieve transitivity
▶ Also supports globally unique, secure identifiers
▶ Achieves query and response privacy
▶ Provides alternative public key infrastructure
▶ Interoperable with DNS

Uses for GNS in GNUnet

▶ Identify IP services hosted in the P2P network
▶ Identities in social networking applications

1 Joint work with Martin Schanzenbach and Matthias Wachs
Zone Management: like in DNS

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Value</th>
<th>Expiration</th>
<th>Public</th>
</tr>
</thead>
<tbody>
<tr>
<td>* +</td>
<td>&lt;new record&gt;</td>
<td>5.mail. +</td>
<td>end of time</td>
<td>✔️</td>
</tr>
<tr>
<td>* prv</td>
<td>&lt;new record&gt;</td>
<td>3QT1G601GU8V05S0C0J870EFB8N3DBJ4L938I8FLR8UKCVGHG</td>
<td>end of time</td>
<td></td>
</tr>
<tr>
<td>* heise</td>
<td>&lt;new record&gt;</td>
<td>heise.de</td>
<td>end of time</td>
<td>✔️</td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>2a02:2a0:3fe:100::8</td>
<td>end of time</td>
<td>✔️</td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>193.99.144.80</td>
<td>end of time</td>
<td>✔️</td>
</tr>
<tr>
<td>* home</td>
<td>&lt;new record&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>* 大学</td>
<td>&lt;new record&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>* short</td>
<td>&lt;new record&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>* mail</td>
<td>&lt;new record&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>* homepage</td>
<td>&lt;new record&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>* fcfs</td>
<td>&lt;new record&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>* www</td>
<td>&lt;new record&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Welcome to gnunet-setup.
Bob can locally reach his webserver via www.gnu
Secure introduction

Bob Builder, Ph.D.

Address: Country, Street Name 23
Phone: 555-12345
Mobile: 666-54321
Mail: bob@H2R84L4JIL3G5C.zkey

Bob gives his public key to his friends, possibly via QR code
Delegation

- Alice learns Bob’s public key
- Alice creates delegation to zone **bob**
- Alice can reach Bob’s webserver via **www.bob.gnu**
Name Resolution

Bob

DHT

Alice

Bob

8FS7

www A 5.6.7.8

Alice

A47G

bob PKEY 8FS7
Name Resolution

Bob

PUT 8FS7-www: 5.6.7.8

DHT

Alice

1 www.bob.gnu ?

Bob

8FS7

www    A    5.6.7.8

Alice

A47G

bob    PKEY    8FS7
Name Resolution

Bob

PUT 8FS7-www: 5.6.7.8

DHT

Alice

www.bob.gnu ?

'bob'?
Name Resolution

0. Bob PUT 8FS7-www: 5.6.7.8

1. Alice

2. 'bob'? PKEY 8FS7!

3. PKEY 8FS7!

Bob

8FS7

www A 5.6.7.8

Alice

A47G

bob PKEY 8FS7
Name Resolution

Bob

PUT 8FS7-www: 5.6.7.8

DHT

Alice

8FS7

www      A      5.6.7.8

Bob

PKEY 8FS7!

Alice

A47G

bob     PKEY       8FS7   

1 www.bob.gnu ?

2 'bob'?

3 PKEY 8FS7!

4 8FS7-www?
Name Resolution

Bob PUT 8FS7-www: 5.6.7.8

DHT

Alice

PKEY 8FS7!

www.bob.gnu?

8FS7-www?

A 5.6.7.8!

PUT 8FS7-www: 5.6.7.8

www A 5.6.7.8

Bob

Alice

8FS7

A47G

bob PKEY 8FS7
GNS as PKI (via DANE/TLSA)

The GNU Operating System

What is GNU?

A Unix-like operating system is a software collection of applications, libraries, and developer tools, plus a program to allocate resources and talk to the hardware, known as a kernel.

The Hurd, GNU's own kernel, is some way from being ready for daily use. Thus, GNU is typically used today with a kernel called Linux. This combination is the GNU/Linux operating system. GNU/Linux is used by millions, though many call it "Linux" by mistake.

The GNU Project was launched in 1984 to develop the GNU system. The name "GNU" is a recursive acronym for "GNU's Not Unix!". "GNU" is pronounced "gno", as one syllable, like saying "grew" but replacing the r with n.

The GNU Project

Identity verified

Permissions Connection

The identity of this website has been verified by GNS CA.

Certificate Information

Your connection to freedom.gnu is encrypted with 256-bit encryption.

The connection uses TLS 1.2.

The connection is encrypted using AES_256_CBC, with SHA1 for message authentication and ECDSA_RSA as the key exchange mechanism.

Site Information

You have never visited this site before today.

What do these mean?

https://freedom.gnu
Query Privacy: Terminology

$G$ generator in ECC curve, a point

$n$ size of ECC group, $n := |G|$, $n$ prime

$x$ private ECC key of zone ($\in \mathbb{Z}_n$)

$P$ public key of zone, a point $P := xG$

$l$ label for record in a zone ($\in \mathbb{Z}_n$)

$R_{P,l}$ set of records for label $l$ in zone $P$

$q_{P,l}$ query hash (hash code for DHT lookup)

$B_{P,l}$ block with information for label $l$ in zone $P$ published in the DHT under $q_{P,l}$
Query Privacy: Cryptography

Publishing $B$ under $q_{P,l} := H(dG)$

\begin{align*}
h & := H(l, P) \quad \text{(1)} \\
d & := h \cdot x \mod n \quad \text{(2)} \\
B_{P,l} & := S_d(E_{HKDF(l,P)}(R_{P,l})), dG \quad \text{(3)}
\end{align*}
Publishing $B$ under $q_P,l := H(dG)$

\begin{align*}
    h &:= H(l, P) \\
    d &:= h \cdot x \mod n \\
    B_{P,l} &:= S_d(E_{HKDF(l,P)}(R_{P,l})), dG
\end{align*}

Searching for $l$ in zone $P$

\begin{align*}
    h &= H(l, P) \\
    q_{P,l} &= H(dG) = H(hxG) = H(hP) \Rightarrow \text{obtain } B_{P,l} \\
    R_{P,l} &= D_{HKDF(l,P)}(B_{P,l})
\end{align*}
Our Vision

<table>
<thead>
<tr>
<th>Internet</th>
<th>GNUnet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google/Facebook</td>
<td>RegEx/PSYC</td>
</tr>
<tr>
<td>DNS/X.509</td>
<td>GNS</td>
</tr>
<tr>
<td>TCP/UDP</td>
<td>Mesh (ECDHE+AES)</td>
</tr>
<tr>
<td>IP/BGP</td>
<td>$R^5N$ DHT</td>
</tr>
<tr>
<td>Ethernet</td>
<td>CORE (ECDHE+AES)</td>
</tr>
<tr>
<td>Phys. Layer</td>
<td>HTTPS/TCP/WLAN/...</td>
</tr>
</tbody>
</table>
PSYC2 for GNUnet

Properties of PSYC2

- Extensible syntax and semantics: try-and-slice pattern
- Supports stateful multicast

Uses for PSYC2 in GNUnet

- P2P social networking foundation
- Push social profiles (state) to all recipients
- Replay from local database used as primary access method
- My data is stored on my machine
Features are frequently added to social applications
Some require changes ("extensions") to data formats and messages
Centralized, browser-based networks can easily update to new version
Decentralized systems must transition *gracefully*

---

Joint work with Carlo v. Loesch and Gabor Toth
Related Work: XML

- Extensible Markup Language
- Syntax is *extensible*
- Extensions have no *semantics*
We are working on PSYC2, the successor to PSYC:

- More compact, mostly human-readable, faster-to-parse relative of XML/JSON/XMPP
- PSYC messages consist of a state update and a method invocation
- PSYC includes interesting ideas for social networking:
  - Stateful multicast
  - History
  - Difference-based updates
- PSYC addresses extensibility problem using try-and-slice pattern
The PSYC state is a set of key-value pairs where the names of keys use underscores to create an inheritance relationship:

- _name
- _name_first
- _name_first_chinese
- _address
- _address_street
- _address_country

The data format for each state is fixed for each top-level label.
A PSYC method has a name which follows the same structure as keys:

- `_message`
- `_message_private`
- `_message_public`
- `_message_public_whisper`
- `_message_announcement`
- `_message_announcement_anonymous`

Methods have access to the current state and a per-message byte-stream.
The Try-and-Slice Pattern

```c
int msg (string method) {
    while (1) {
        switch (method) {
            case "_notice_update_news": // handle news update
                return 1;
            case "_notice": // handle generic notice
                return 1;
            case "_message": // handle generic message
                return 1;
            // ...
        }
    }
    int glyph = strrpos (method, '_');
    if (glyph <= 1) break;
    truncate (method, glyph);
}
```
Advantages of Try-and-Slice

- Extensible, can support many applications
- Can be applied to state and methods
- Defines what backwards-compatible extensibility means:
  - Can incrementally expand implementations by deepening coverage
  - Incompatible updates = introduce new top-level methods
## Our Vision

### Internet

<table>
<thead>
<tr>
<th>Google/Facebook</th>
<th>DNS/X.509</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP/UDP</td>
<td></td>
</tr>
<tr>
<td>IP/BGP</td>
<td></td>
</tr>
<tr>
<td>Ethernet</td>
<td></td>
</tr>
<tr>
<td>Phys. Layer</td>
<td></td>
</tr>
</tbody>
</table>

### GNUnet

<table>
<thead>
<tr>
<th>RegEx/PSYC</th>
<th>GNS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh (ECDHE+AES)</td>
<td></td>
</tr>
<tr>
<td>$R^5N$ DHT</td>
<td></td>
</tr>
<tr>
<td>CORE (ECDHE+AES)</td>
<td></td>
</tr>
<tr>
<td>HTTPS/TCP/WLAN/</td>
<td></td>
</tr>
</tbody>
</table>
RegEx Search for GNUnet

Properties of RegEx Search

- Capability discovery in DHT-based P2P networks using regular expressions
- Linear latency in the length of the search string
- Suitable for applications that can tolerate moderate latency

Uses for RegEx in GNUnet

- Discovery of matching services, such as VPN exit nodes
- Topic-based subscriptions in messaging (decentralized MQTT)
Distributed Search via Regular Expressions: Idea

1. Offerer creates regular expression describing service
2. Regular expression is compiled to a DFA
3. DFA is stored in the DHT
4. Patron matches using a string

---

3 Joint work with Max Szengel, Ralph Holz, Bart Polot and Heiko Niedermayer
Problem: Mapping of States to Keys

Regular expression \((ab|cd)e^*f\) and corresponding DFA
Problem: Merging of DFAs

Regular expressions \((ab|cd)e^*f\) and \((ab|cd)e^*fg^*\) with corresponding DFAs
Problem: Merging of DFAs

Merged NFA for regular expressions \((ab|cd)e^*fg^*\) and \((ab|cd)e^*f\)
Problem: Decentralizing the Start State

Regular expression: $abc^*defg^*h$ and $k = 4$. 
Evaluation

- Implementation in GNUnet
- Profiling of Internet-scale routing using regular expressions to describe AS address ranges
- CAIDA AS data set: Real AS data
Evaluation

Distributed Hash Table

AS 7212
129.59.0.0/16
160.129.0.0/16
192.111.108.0/24
192.111.109.0/24
192.111.110.0/24
199.78.112.0/24
199.78.113.0/24
199.78.114.0/24
199.78.115.0/24

AS 12816
129.187.0.0/16
131.159.0.0/16
138.244.0.0/15
138.246.0.0/16
... 
192.68.211.0/24
192.68.212.0/22

AS 12812
193.188.128.0/24
193.188.129.0/24
193.188.130.0/24
193.188.131.0/24

AS 10001
49.128.128.0/19
61.195.240.0/20
122.49.192.0/21
123.255.240.0/21
175.41.32.0/21
202.75.112.0/20
202.238.32.0/20
210.48.128.0/21
211.133.224.0/20
219.124.0.0/20
219.124.0.21
219.124.8.0/21

AS 8265
91.223.12.0/24
195.96.192.0/24
195.96.193.0/24
195.96.194.0/24
195.96.196.0/22
195.96.200.0/22
195.96.204.0/22
195.96.208.0/21
195.96.216.0/21

AS 10002
61.114.64.0/20
61.195.128.0/20
120.50.224.0/19
120.72.0.0/20
202.180.192.0/20

AS 10003
61.114.64.0/20
61.195.128.0/20
120.50.224.0/19
120.72.0.0/20
202.180.192.0/20

AS 56357
188.95.232.0/22
192.48.107.0/24

AS 825
91.221.132.0/24
91.221.133.0/24
192.16.240.0/20

AS 931
46.183.182.0/21
103.10.233.0/24
186.233.120.0/21
186.233.120.0/22
186.233.124.0/22

AS 32310
204.94.175.0/24

AS 50038
57.236.47.0/24
57.236.48.0/24
57.236.51.0/24
193.104.87.0/24

AS 7212

AS 12812

AS 10001

AS 8265

AS 931

AS 32310

AS 50038

AS 825

Distributed Hash Table
Evaluation: Results of Simulation (1)

Number of transitions and states in the merged NFA

Dataset: All 40,696 ASes
Evaluation: Results of Simulation

Degree of non-determinism at states in the merged NFA

Dataset: All 40,696 ASes
Evaluation: Results of Simulation (3)

Dataset: All 40,696 ASes
Evaluation: Results of Emulation

Search duration averaged over five runs with randomly connected peers.

![Graph showing search duration and percentage of matched strings for 1,000, 2,000, and 4,000 peers.](image)
<table>
<thead>
<tr>
<th>Internet</th>
<th>GNUnet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google/Facebook</td>
<td>News/Timeline</td>
</tr>
<tr>
<td>DNS/X.509</td>
<td>Scalarproduct</td>
</tr>
<tr>
<td>TCP/UDP</td>
<td>Mesh (ECDHE + AES)</td>
</tr>
<tr>
<td>IP/BGP</td>
<td>$R^5 N$ DHT</td>
</tr>
<tr>
<td>Ethernet</td>
<td>CORE (ECDHE + AES)</td>
</tr>
<tr>
<td>Phys. Layer</td>
<td>HTTPS/TCP/WLAN/...</td>
</tr>
</tbody>
</table>
Scalarproduct for GNUUnet

Properties of SMC Scalarproduct

- Scalarproduct over map on intersecting sets, not just vectors
- Privacy-preserving (but need to limit number of interactions)
- Relatively efficient in bandwidth and CPU usage

Uses for Scalarproduct in GNUUnet

- Collaborative filtering
- Maybe: collaborative attack detection
We use the Paillier cryptosystem:

\[ E_K(m) : = g^m \cdot r^n \mod n^2, \]  
\[ D_K(c) : = \frac{(c^\lambda \mod n^2) - 1}{n} \cdot \mu \mod n \]  

where the public key \( K = (n, g) \), \( m \) is the plaintext, \( c \) the ciphertext, \( n \) the product of \( p, q \in \mathbb{P} \) of equal length, and \( g \in \mathbb{Z}_{n^2}^* \). The private key is \( (\lambda, \mu) \), which is computed from \( p \) and \( q \) as follows:

\[ \lambda : = \text{lcm}(p - 1, q - 1), \]  
\[ \mu : = \left( \frac{(g^\lambda \mod n^2) - 1}{n} \right)^{-1} \mod n. \]
Paillier offers **additive** homomorphism

Paillier offers additive homomorphic public-key encryption, that is:

$$E_K(a) \otimes E_K(b) \equiv E_K(a + b)$$  \hspace{1cm} (11)

for some public key $K$. 
Background: Secure Multiparty Computation

- Alice and Bob have private inputs $a_i$ and $b_i$.
- Alice and Bob run a protocol to jointly calculate $f(a_i, b_i)$.
- One of them learns the result.
- Adversary model: honest but curious
Secure Scalar Product

- Original idea by Ioannids et al. in 2002 (use: \((a - b)^2 = a^2 - 2ab + b^2\))
- Refined by Amirbekyan et al. in 2007 (corrected math)
- Implemented with practical extensions in GNUnet (negative numbers, small numbers, concrete protocol, set intersection, implementation).
Alice has public key $A$ and input map $m_A : M_A \rightarrow \mathbb{Z}$.

Bob has public key $B$ and input map $m_B : M_B \rightarrow \mathbb{Z}$.

We want to calculate

$$ \sum_{i \in M_A \cap M_B} m_A(i)m_B(i) \quad (12) $$

We first calculate $M = M_A \cap M_B$.

Define $a_i := m_A(i)$ and $b_i := m_B(i)$ for $i \in M$.

Let $s$ denote a shared static offset.
Network Protocol

- Alice transmits $E_A(s + a_i)$ for $i \in M$ to Bob.
- Bob creates two random permutations $\pi$ and $\pi'$ over the elements in $M$, and a random vector $r_i$ for $i \in M$ and sends

\[
R := E_A(s + a_\pi(i)) \otimes E_A(s - r_\pi(i) - b_\pi(i)) \quad (13)
\]
\[
= E_A(2 \cdot s + a_\pi(i) - r_\pi(i) - b_\pi(i)), \quad (14)
\]

\[
R' := E_A(s + a_{\pi'}(i)) \otimes E_A(s - r_{\pi'}(i)) \quad (15)
\]
\[
= E_A(2 \cdot s + a_{\pi'}(i) - r_{\pi'}(i)), \quad (16)
\]

\[
S := \sum (r_i + b_i)^2, \quad (17)
\]

\[
S' := \sum r_i^2 \quad (18)
\]
Decryption (1/3)

Alice decrypts $R$ and $R'$ and computes for $i \in M$:

\[
\begin{align*}
    a_{\pi(i)} - b_{\pi(i)} - r_{\pi(i)} &= D_A(R) - 2 \cdot s, \\
    a_{\pi'(i)} - r_{\pi'(i)} &= D_A(R') - 2 \cdot s,
\end{align*}
\]

which is used to calculate

\[
\begin{align*}
    T &= \sum_{i \in M} a_i^2, \\
    U &= -\sum_{i \in M} (a_{\pi(i)} - b_{\pi(i)} - r_{\pi(i)})^2, \\
    U' &= -\sum_{i \in \pi'(i)} (a_{\pi'(i)} - r_{\pi'(i)})^2
\end{align*}
\]
Decryption (2/3)

She then computes

\[ P := S + T + U \]

\[ = \sum_{i \in M} (b_i + r_i)^2 + \sum_{i \in M} a_i^2 + \left( - \sum_{i \in M} (a_i - b_i - r_i)^2 \right) \]

\[ = \sum_{i \in M} ( (b_i + r_i)^2 + a_i^2 - (a_i - b_i - r_i)^2 ) \]

\[ = 2 \cdot \sum_{i \in M} a_i (b_i + r_i). \]

\[ P' := S' + T + U' \]

\[ = \sum_{i \in M} r_i^2 + \sum_{i \in M} a_i^2 + \left( - \sum_{i \in M} (a_i - r_i)^2 \right) \]

\[ = \sum_{i \in M} (r_i^2 + a_i^2 - (a_i - r_i)^2) = 2 \cdot \sum_{i \in M} a_i r_i. \]
Finally, Alice computes the scalar product using:

\[ \frac{P - P'}{2} = \sum_{i \in M} a_i (b_i + r_i) - \sum_{i \in M} a_i r_i = \sum_{i \in M} a_i b_i. \]  

(24)
## Our Vision

<table>
<thead>
<tr>
<th>Internet</th>
<th>GNUnet</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Google/Facebook</strong></td>
<td><strong>News/Timeline</strong></td>
</tr>
<tr>
<td><strong>DNS/X.509</strong></td>
<td><strong>Scalarproduct</strong></td>
</tr>
<tr>
<td><strong>TCP/UDP</strong></td>
<td><strong>Mesh (ECDHE+AES)</strong></td>
</tr>
<tr>
<td><strong>IP/BGP</strong></td>
<td><strong>$R^5N$ DHT</strong></td>
</tr>
<tr>
<td><strong>Ethernet</strong></td>
<td><strong>CORE (ECDHE+AES)</strong></td>
</tr>
<tr>
<td><strong>Phys. Layer</strong></td>
<td><strong>HTTPS/TCP/WLAN/...</strong></td>
</tr>
</tbody>
</table>
Future Work: Privacy-enhanced “Gossple”

1. Alice selects peers ⇒ Bob
2. Alice and Bob compute scalar product ⇒ similarity
3. Bob forwards news to Alice with ranking based on similarity
4. Alice constructs timeline, ranks news, and
5. adapts her forwarding (2) and peer selection (1)
Dimensions for ranking news quality

- Agreement (on opinion, highly subjective)
Dimensions for ranking news quality

- Agreement (on opinion, highly subjective)
- Presentation (use of language, formatting, graphics)
Dimensions for ranking news quality

- Agreement (on opinion, highly subjective)
- Presentation (use of language, formatting, graphics)
- Accuracy (use of scientific method, well-sourced)
Dimensions for ranking news quality

- Agreement (on opinion, highly subjective)
- Presentation (use of language, formatting, graphics)
- Accuracy (use of scientific method, well-sourced)
- Relevance (by topic ⇒ need tags)
Components for Future Work

- Efficient set intersection
  (current design: $O(n \log n)$ with $O(\log n)$ rounds)
- **Secure** decentralized random peer selection
- Tagging system
- Reputation system for authors
More Open Issues

- Information leakage over time!
- Evaluation scenarios?
- Usability
- Social effects
Conclusion

▶ Decentralization is necessary
▶ Security and scalability are hard issues

We need to build systems that address both!
References:


GNUnet: Framework Architecture
GNUnet: Envisioned Applications
Research Agenda

- Secure, scaleable multicast
- Practical secure multiparty computation
- Tool support for building distributed systems
- Secure routing, censorship circumvention