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


Source: csmoni


Source: gawand.org


Source: padresteve.com


Source: www.businessinsid


SECURITY
UNITED STATES OF AMERICA
line.de


الموقع محظور
نأسف إن الموقع الذي أردت تصفحه قد أحجب وذلك بسبب إحتوائه على نشاط مخالف للقيم الإجتماعية أو السياسية أو الثقافية أو الدينية لدولة الإمارات العربية المتحدة.
في حالة أردت فتح موقع قد أحجب. الرجاء قم بتعبئة استمارة الملاحظات الموجودة على موقعنا.
We apologize the site you are attempting to visit has been blocked due to its content being inconsistent with the religious, cultural, political and moral values of the United Arab Emirates.
If you think this site should not be blocked, please visit the Feedback Form available on our website.
SITE BLOCKED
Source: wikileaks.org


G.C.H.Q.
ALWAYS LISTENING TO OUR CUSTOMERS


Bundesnachrichtendiens
Source: wikicommons.org


Source: enturebeat.com

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

Google/Facebook
DNS/X.509
TCP/UDP
IP/BGP
Ethernet
Phys. Layer

Our Vision

Internet	
Google/Facebook	
DNS/X.509	
TCP/UDP	
IP/BGP	
Ethernet	
Phys. Layer	HTTPS/TCP/WLAN/...

Our Vision

Internet

Google/Facebook
DNS/X.509
TCP/UDP
IP/BGP
Ethernet
Phys. Layer

CORE (ECDHE+AES)
HTTPS/TCP/WLAN/...

Our Vision

Internet	
Google/Facebook	
DNS/X.509	
TCP/UDP	
IP/BGP	R^5N DHT
Ethernet	CORE (ECDHE+AES)
Phys. Layer	HTTPS/TCP/WLAN/...

Our Vision

Internet	
Google/Facebook	
DNS/X.509	
TCP/UDP	Mesh (ECDHE+AES)
IP/BGP	R^5N DHT
Ethernet	CORE (ECDHE+AES)
Phys. Layer	HTTPS/TCP/WLAN/...

Our Vision

Internet	
Google/Facebook	
DNS/X.509	**GNU Name System**
TCP/UDP	Mesh (ECDHE+AES)
IP/BGP	R^5N DHT
Ethernet	CORE (ECDHE+AES)
Phys. Layer	HTTPS/TCP/WLAN/...

Our Vision

Internet

Google/Facebook
DNS/X.509
TCP/UDP
IP/BGP
Ethernet
Phys. Layer

RegEx/PSYC
GNU Name System
Mesh (ECDHE+AES)
R^5N DHT
CORE (ECDHE+AES)
HTTPS/TCP/WLAN/...

Our Vision

Internet
Google/Facebook
DNS/X.509
TCP/UDP
IP/BGP
Ethernet
Phys. Layer

GNUnet
RegEx/PSYC
GNU Name System
Mesh (ECDHE+AES)
R^5N DHT
CORE (ECDHE+AES)
HTTPS/TCP/WLAN/...

The GNU Name System[1]

Properties of GNS

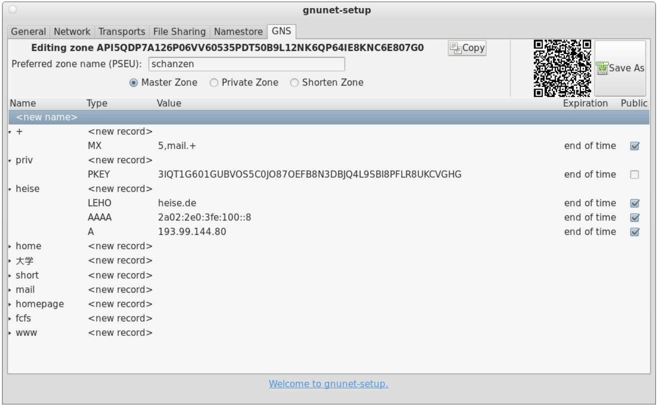
- Decentralized name system with secure memorable names
- Delegation used to achieve transitivity
- Also supports globally unique, secure identifiers
- Achieves query and response privacy
- Provides alternative public key infrastructure
- Interoperable with DNS

Uses for GNS in GNUnet

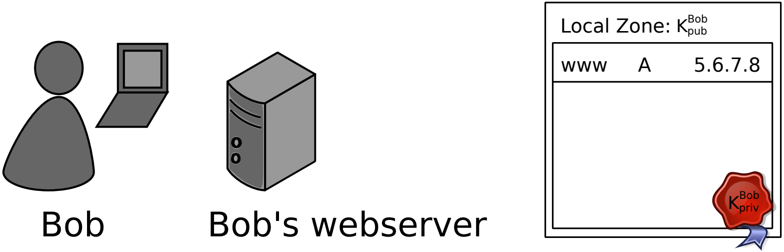
- Identify IP services hosted in the P2P network
- Identities in social networking applications

[1] Joint work with Martin Schanzenbach and Matthias Wachs

Zone Management: like in DNS



Name resolution in GNS

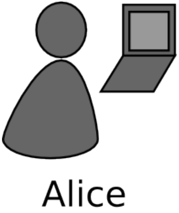


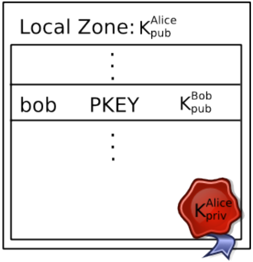
- Bob can locally reach his webserver via **www.gnu**

Secure introduction

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

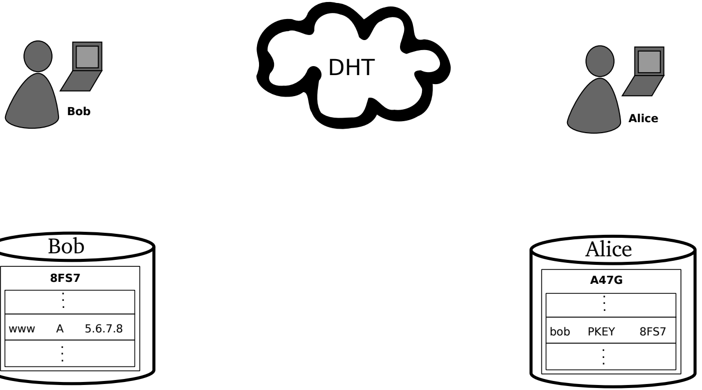
Delegation



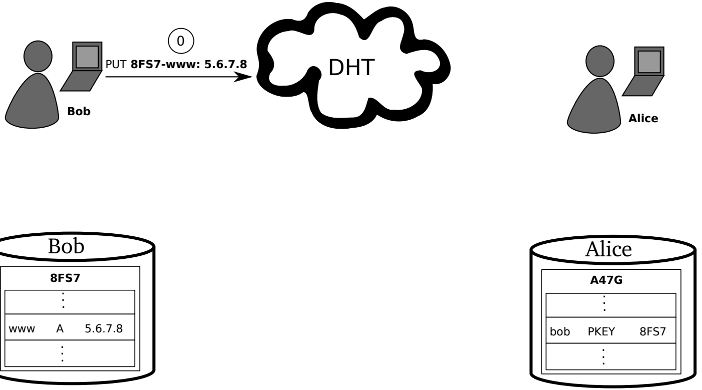


- Alice learns Bob's public key
- Alice creates delegation to zone **bob**
- Alice can reach Bob's webserver via **www.bob.gnu**

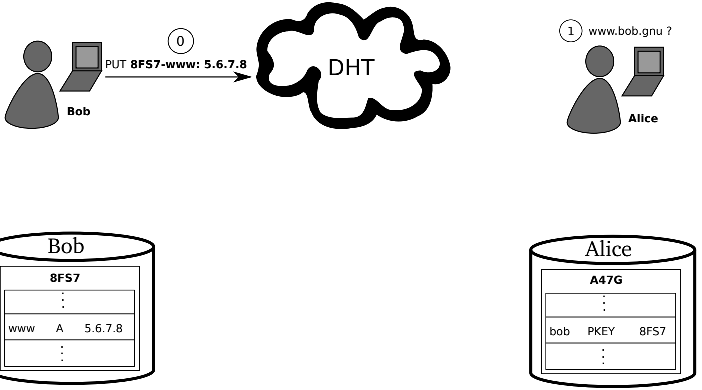
Name Resolution



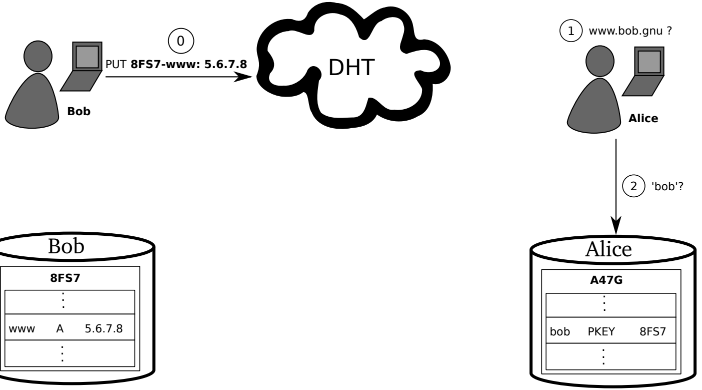
Name Resolution



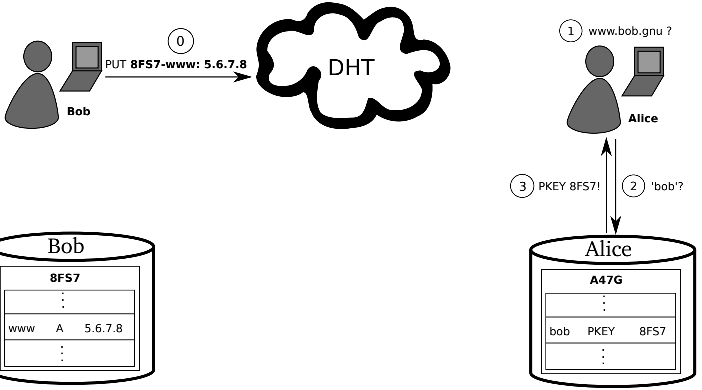
Name Resolution



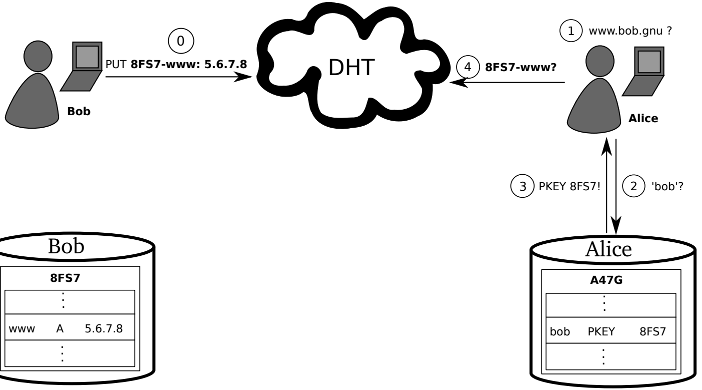
Name Resolution



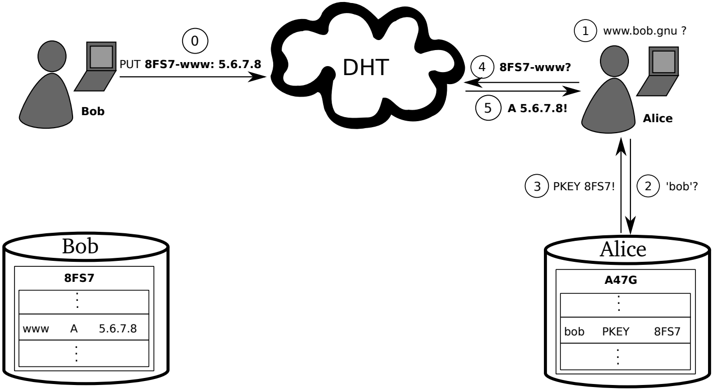
Name Resolution



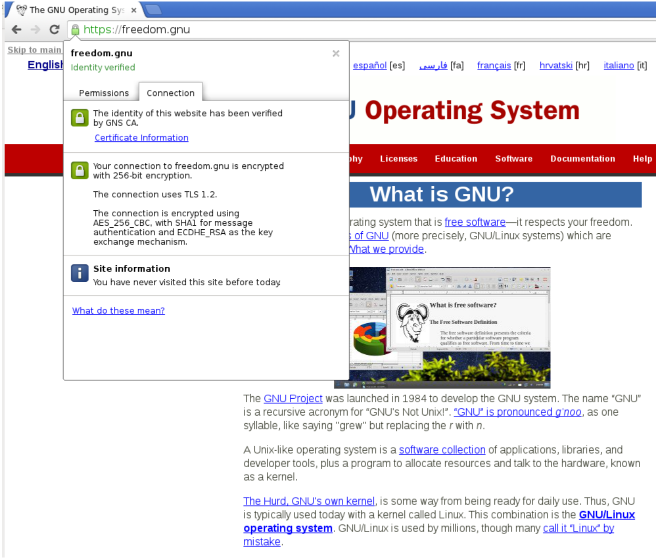
Name Resolution



Name Resolution



GNS as PKI (via DANE/TLSA)



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

$$
\begin{aligned}
h &:= H(l, P) && (1)\\
d &:= h \cdot x \mod n && (2)\\
B_{P,l} &:= S_d(E_{HKDF(l,P)}(R_{P,l})), dG && (3)
\end{aligned}
$$

Query Privacy: Cryptography

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

$$
\begin{aligned}
h &:= H(l, P) && (1)\\
d &:= h \cdot x \mod n && (2)\\
B_{P,l} &:= S_d(E_{HKDF(l,P)}(R_{P,l})), dG && (3)
\end{aligned}
$$

Searching for l in zone P

$$
\begin{aligned}
h &= H(l, P) && (4)\\
q_{P,l} &= H(dG) = H(hxG) = H(hP) \Rightarrow \texttt{obtain } B_{P,l} && (5)\\
R_{P,l} &= D_{HKDF(l,P)}(B_{P,l}) && (6)
\end{aligned}
$$

Our Vision

Internet	*GNUnet*
Google/Facebook	RegEx/**PSYC**
DNS/X.509	GNS
TCP/UDP	Mesh (ECDHE+AES)
IP/BGP	R^5N DHT
Ethernet	CORE (ECDHE+AES)
Phys. Layer	HTTPS/TCP/WLAN/...

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**

The Evolution Challenge[2]

- 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*

[2] Joint work with Carlo v. Loesch and Gabor Toth

Related Work: XML

- Extensible Markup Language
- Syntax is *extensible*
- Extensions have no **semantics**

PSYC

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

PSYC State: Example

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.

PSYC Methods: Example

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

```
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	*GNUnet*
Google/Facebook	**RegEx**/PSYC
DNS/X.509	GNS
TCP/UDP	Mesh (ECDHE+AES)
IP/BGP	R^5N DHT
Ethernet	CORE (ECDHE+AES)
Phys. Layer	HTTPS/TCP/WLAN/...

RegEx Search for GNUnet

Properties of RegEx Search

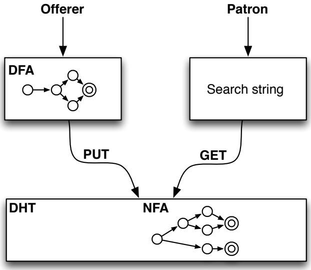
- Capability discovery in DHT-based P2P networks using regular expressions
- Linear latency in the length of the search string
- Suitable for applications that can tolerate moderate latency

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[3]

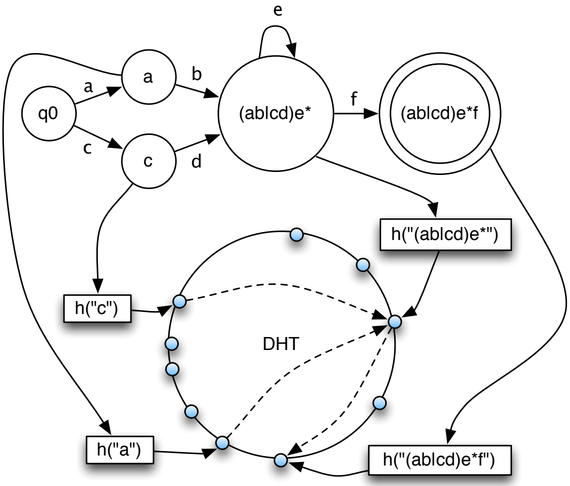
1. Offerer creates regular expression describing service
2. Regular expression is compiled to a DFA
3. DFA is stored in the DHT
4. Patron matches using a string



[3] Joint work with Max Szengel, Ralph Holz, Bart Polot and Heiko Niedermayer

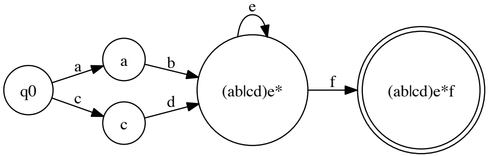
Problem: Mapping of States to Keys

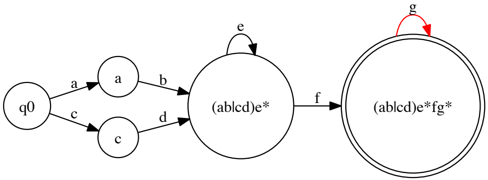
Regular expression $(ab|cd)e^*f$ and corresponding **DFA**



Problem: Merging of DFAs

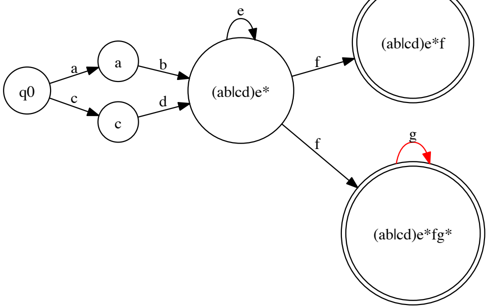
Regular expressions $(ab|cd)e^*f$ and $(ab|cd)e^*fg^*$ with corresponding **DFAs**





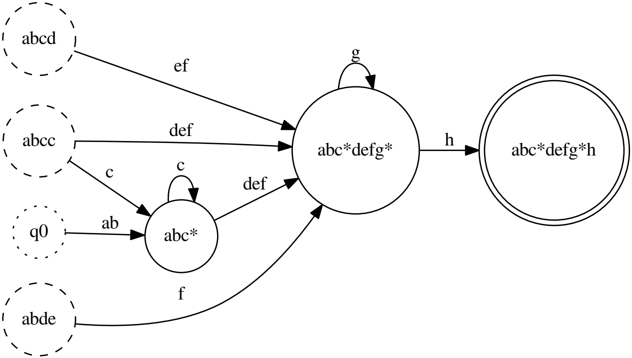
Problem: Merging of DFAs

Merged **NFA** for regular expressions $(ab|cd)e^*fg^*$ and $(ab|cd)e^*f$



Problem: Decentralizing the Start State

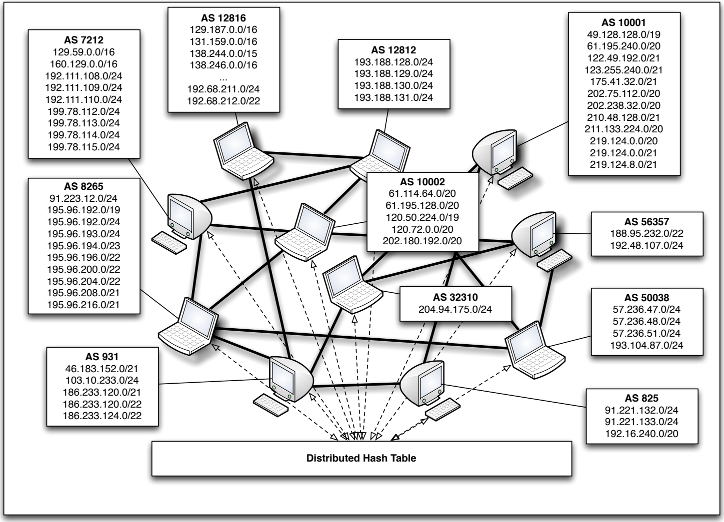
Regular expression: abc^*defg^*h and $k = 4$.



Evaluation

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

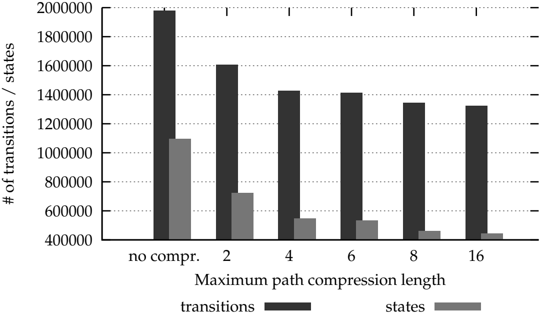
Evaluation



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
219.124.0.0/21
219.124.8.0/21
AS 8265
91.223.12.0/24
195.96.192.0/19
195.96.192.0/24
195.96.193.0/24
195.96.194.0/23
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 56357
188.95.232.0/22
192.48.107.0/24
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 931
46.183.152.0/21
103.10.233.0/24
186.233.120.0/21
186.233.120.0/22
186.233.124.0/22
AS 825
91.221.132.0/24
91.221.133.0/24
192.16.240.0/20
Distributed Hash Table

Evaluation: Results of Simulation (1)

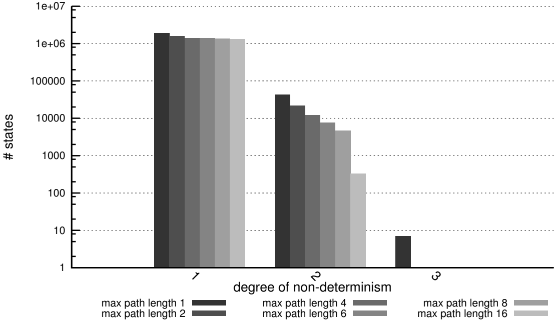
Number of transitions and states in the merged NFA



Dataset: All $40,696$ ASes

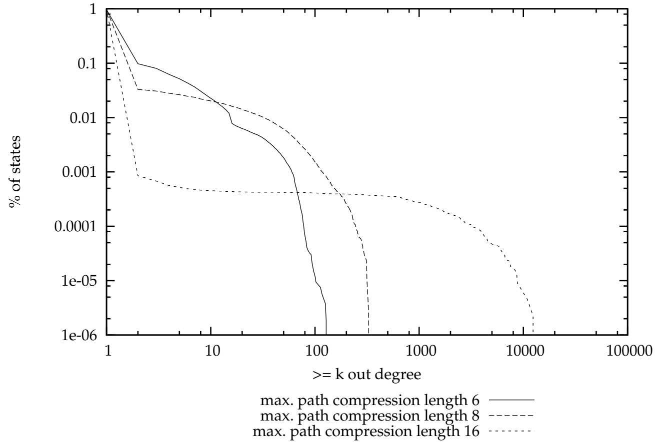
Evaluation: Results of Simulation

Degree of non-determinism at states in the merged NFA



Dataset: All $40,696$ ASes

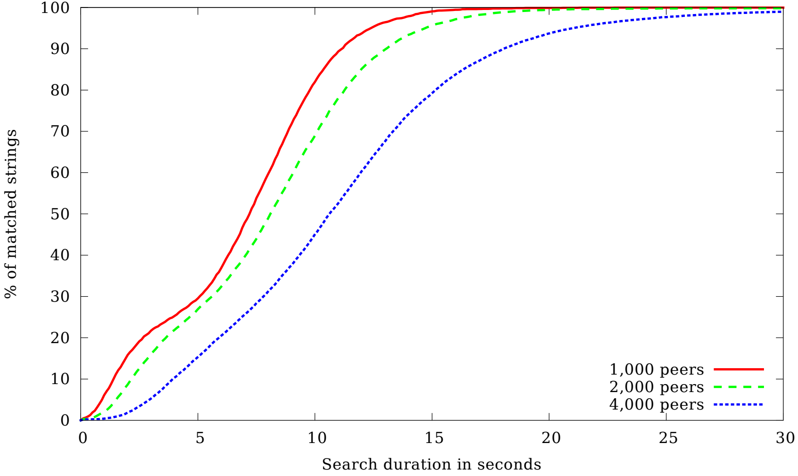
Evaluation: Results of Simulation (3)



Dataset: All $40,696$ ASes

Evaluation: Results of Emulation

Search duration averaged over five runs with randomly connected peers.



Our Vision

Internet	*GNUnet*
Google/Facebook	News/Timeline
DNS/X.509	**Scalarproduct**
TCP/UDP	Mesh (ECDHE+AES)
IP/BGP	R^5N DHT
Ethernet	CORE (ECDHE+AES)
Phys. Layer	HTTPS/TCP/WLAN/...

Scalarproduct for GNUnet

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 GNUnet

- Collaborative filtering
- Maybe: collaborative attack detection

Background: Paillier

We use the Paillier cryptosystem:

$$\mathsf{E}_K(m) := g^m \cdot r^n \mod n^2, \tag{7}$$

$$\mathsf{D}_K(c) := \frac{(c^\lambda \mod n^2) - 1}{n} \cdot \mu \mod n \tag{8}$$

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 (λ, μ), which is computed from p and q as follows:

$$\lambda := \mathrm{lcm}(p-1, q-1), \tag{9}$$

$$\mu := \left(\frac{(g^\lambda \mod n^2) - 1}{n}\right)^{-1} \mod n. \tag{10}$$

Paillier offers **additive** homomorphism

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

$$\mathsf{E}_K(a) \otimes \mathsf{E}_K(b) \equiv \mathsf{E}_K(a + b) \tag{11}$$

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).

Preliminaries

- Alice has public key A and input map $m_A : M_A \to \mathbb{Z}$.
- Bob has public key B and input map $m_B : M_B \to \mathbb{Z}$.
- We want to calculate

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

- 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 $\mathsf{E}_A(s + a_i)$ for $i \in M$ to Bob.
- Bob creates two random permutations π and π' over the elements in M, and a random vector r_i for $i \in M$ and sends

$$\begin{aligned}
R :&= \mathsf{E}_A(s + a_{\pi(i)}) \otimes \mathsf{E}_A(s - r_{\pi(i)} - b_{\pi(i)}) && (13)\\
&= \mathsf{E}_A(2 \cdot s + a_{\pi(i)} - r_{\pi(i)} - b_{\pi(i)}), && (14)\\
R' :&= \mathsf{E}_A(s + a_{\pi'(i)}) \otimes \mathsf{E}_A(s - r_{\pi'(i)}) && (15)\\
&= \mathsf{E}_A(2 \cdot s + a_{\pi'(i)} - r_{\pi'(i)}), && (16)\\
S :&= \sum (r_i + b_i)^2, && (17)\\
S' :&= \sum r_i^2 && (18)
\end{aligned}$$

Decryption (1/3)

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

$$a_{\pi(i)} - b_{\pi(i)} - r_{\pi(i)} = \mathrm{D}_A\left(R\right) - 2 \cdot s, \tag{19}$$

$$a_{\pi'(i)} - r_{\pi'(i)} = \mathrm{D}_A\left(R'\right) - 2 \cdot s, \tag{20}$$

which is used to calculate

$$T := \sum_{i \in M} a_i^2 \tag{21}$$

$$U := -\sum_{i \in M} (a_{\pi(i)} - b_{\pi(i)} - r_{\pi(i)})^2 \tag{22}$$

$$U' := -\sum_{i \in M} (a_{\pi'(i)} - r_{\pi'(i)})^2 \tag{23}$$

Decryption (2/3)

She then computes

$$
\begin{aligned}
P :&= S + T + U \\
&= \sum_{i \in M} (b_i + r_i)^2 + \sum_{i \in M} a_i^2 + \left(- \sum_{i \in M} (a_i - b_i - r_i)^2 \right) \\
&= \sum_{i \in M} \left((b_i + r_i)^2 + a_i^2 - (a_i - b_i - r_i)^2 \right) \\
&= 2 \cdot \sum_{i \in M} a_i (b_i + r_i). \\
P' :&= S' + T + U' \\
&= \sum_{i \in M} r_i^2 + \sum_{i \in M} a_i^2 + \left(- \sum_{i \in M} (a_i - r_i)^2 \right) \\
&= \sum_{i \in M} \left(r_i^2 + a_i^2 - (a_i - r_i)^2 \right) = 2 \cdot \sum_{i \in M} a_i r_i.
\end{aligned}
$$

Decryption (3/3)

Finally, Alice computes the scalar product using:

$$\frac{P - P'}{2} = \sum_{i \in M} a_i (b_i + r_i) - \sum_{i \in M} a_i r_i = \sum_{i \in M} a_i b_i. \qquad (24)$$

Our Vision

Internet
Google/Facebook
DNS/X.509
TCP/UDP
IP/BGP
Ethernet
Phys. Layer

GNUnet
News/Timeline
Scalarproduct
Mesh (ECDHE+AES)
R^5N DHT
CORE (ECDHE+AES)
HTTPS/TCP/WLAN/...

Future Work: Privacy-enhanced “Gossple”

1. Alice selects peers $\Rightarrow$ Bob
2. Alice and Bob compute scalar product $\Rightarrow$ 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 $\Rightarrow$ 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

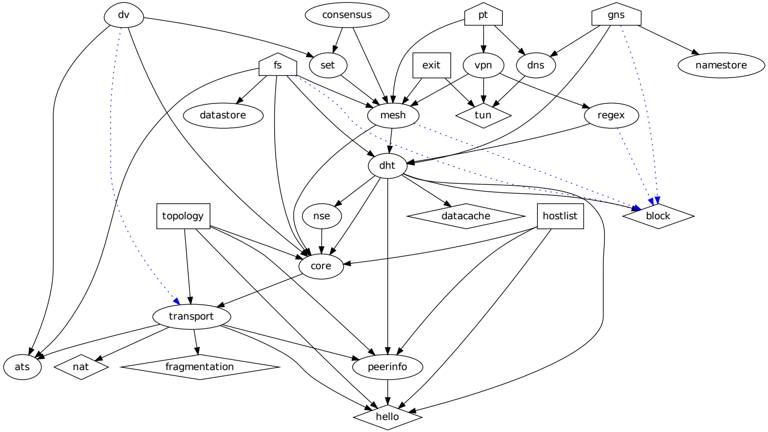
We need to build systems that address both!

Do you have any questions?

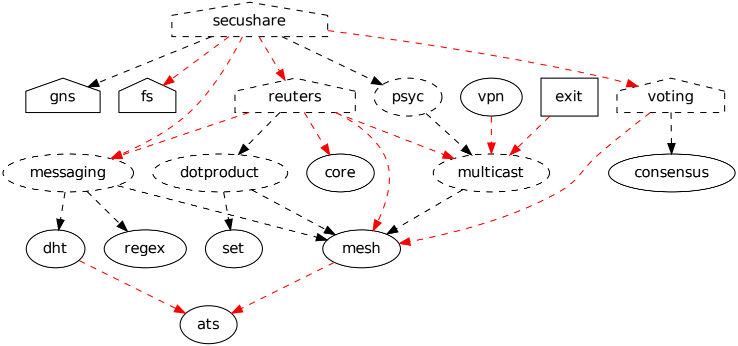
References:

- Glenn Greenwald and Ewen MacAskill. *NSA Prism program taps in to user data of Apple, Google and others*. In **The Guardian**, June 7 2013.
- Nathan Evans and Christian Grothoff. *R5N. Randomized Recursive Routing for Restricted-Route Networks*. **5th International Conference on Network and System Security**, 2011.
- M. Schanzenbach *Design and Implementation of a Censorship Resistant and Fully Decentralized Name System*. **Master's Thesis (TUM)**, 2012.
- M. Szengel. *Decentralized Evaluation of Regular Expressions for Capability Discovery in Peer-to-Peer Networks*. **Master's Thesis (TUM)**, 2012.

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