The GNU Name System and the Future of Social Networking with GNUnet

Christian Grothoff

Technische Universität München

24.08.2013

"Never doubt your ability to change the world." –Glenn Greenwald

Cyberwar

Presidential Policy Directive 20, issued October 2012 and released by Edward Snowden, outlines U.S. cyberwar policy:

"Offensive Cyber Effect Operations (OCEO) can offer unique and unconventional capabilities to **advance U.S. national objectives** around the world with little or no warning to the adversary or target and with potential effects ranging from **subtle** to severely damaging. (...)
The United States Government shall identify potential targets of national importance where OCEO can offer a favorable **balance of effectiveness and risk** as compared with other instruments of national power, establish and maintain OCEO capabilities integrated as appropriate with other U.S. offensive capabilities, and execute those capabilities in a manner consistent with the provisions of this directive."

Not Just Monitoring

- US **controls** key Internet infrastructure:
  - Number resources (IANA)
  - Domain Name System (Root zone)
  - DNSSEC root certificate
  - X.509 CAs (HTTPS certificates)
  - Major browser vendors (CA root stores!)
- Encryption does not help if PKI is compromised!

Decentralize Everything

- Encrypt everything end-to-end
- Decentralized PKI
- Decentralized data storage
- No servers
- No authorities

Decentralize Everything

- Encrypt everything end-to-end
- Decentralized PKI
- Decentralized data storage
- No servers
- No authorities

$\Rightarrow$ No juicy targets for APTs

Decentralized vs. Centralized

Decentralized:	**Centralized:**
Slower	
No economics of scale	
More complex to use	
More complex to develop	
Hard to secure	
Hard to evolve	

Decentralized vs. Centralized

Decentralized:	**Centralized:**
Slower	Compromised
No economics of scale	
More complex to use	
More complex to develop	
Hard to secure	
Hard to evolve	

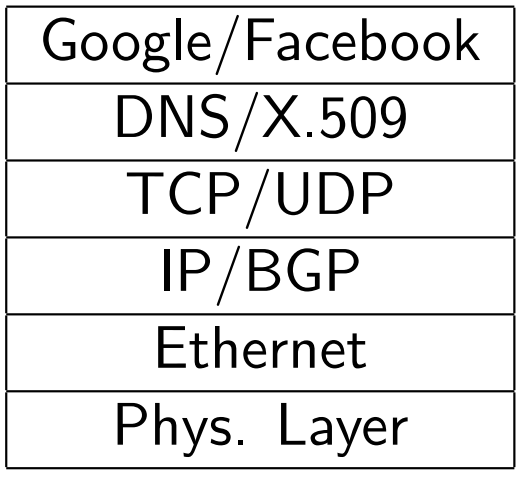
My Research and Development Agenda

Make decentralized systems:

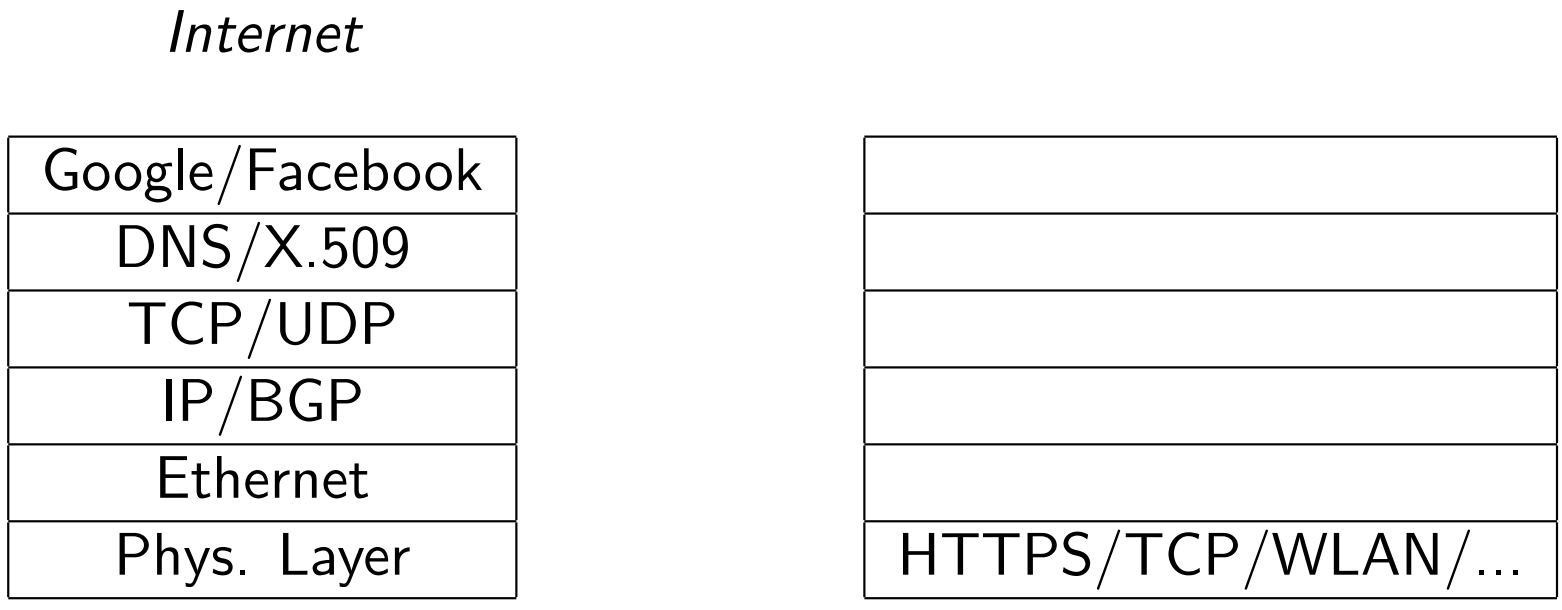
- Faster, more scalable
- Easier to develop, deploy and use
- Easier to evolve and extend
- Secure (privacy-preserving, censorship-resistant, available, ...)

Our Vision

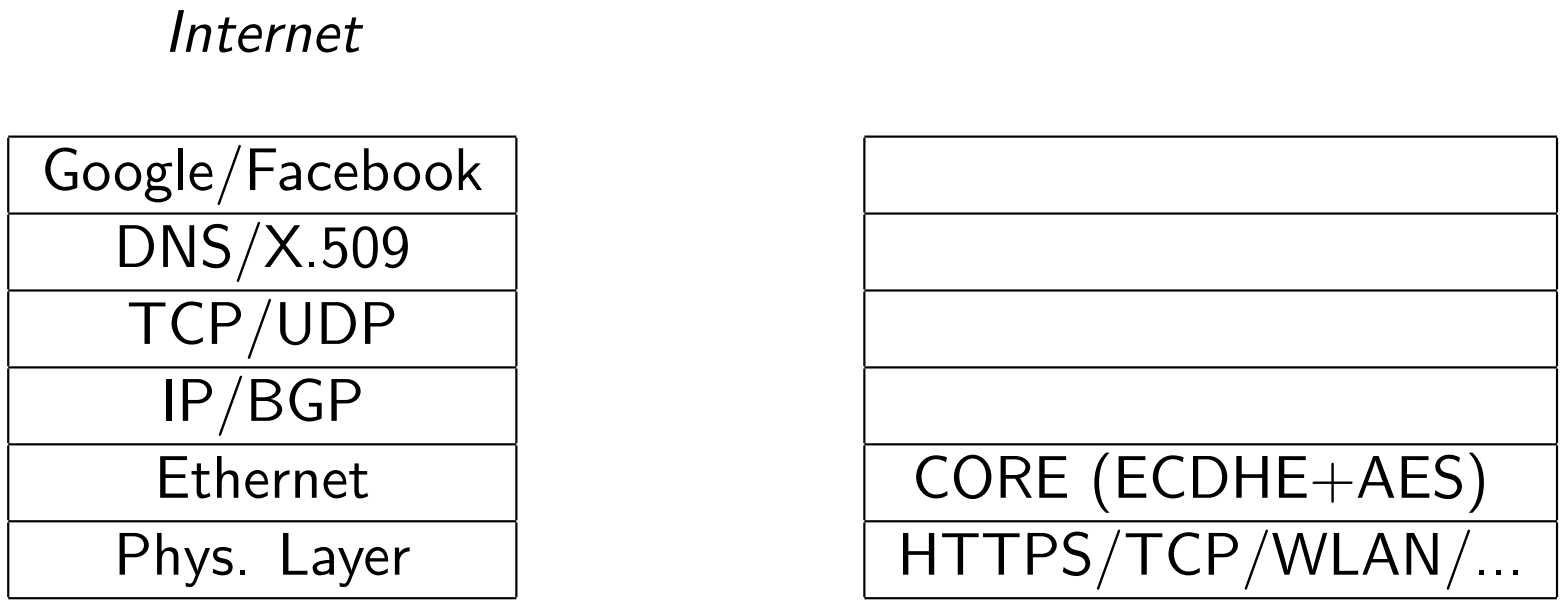
Internet



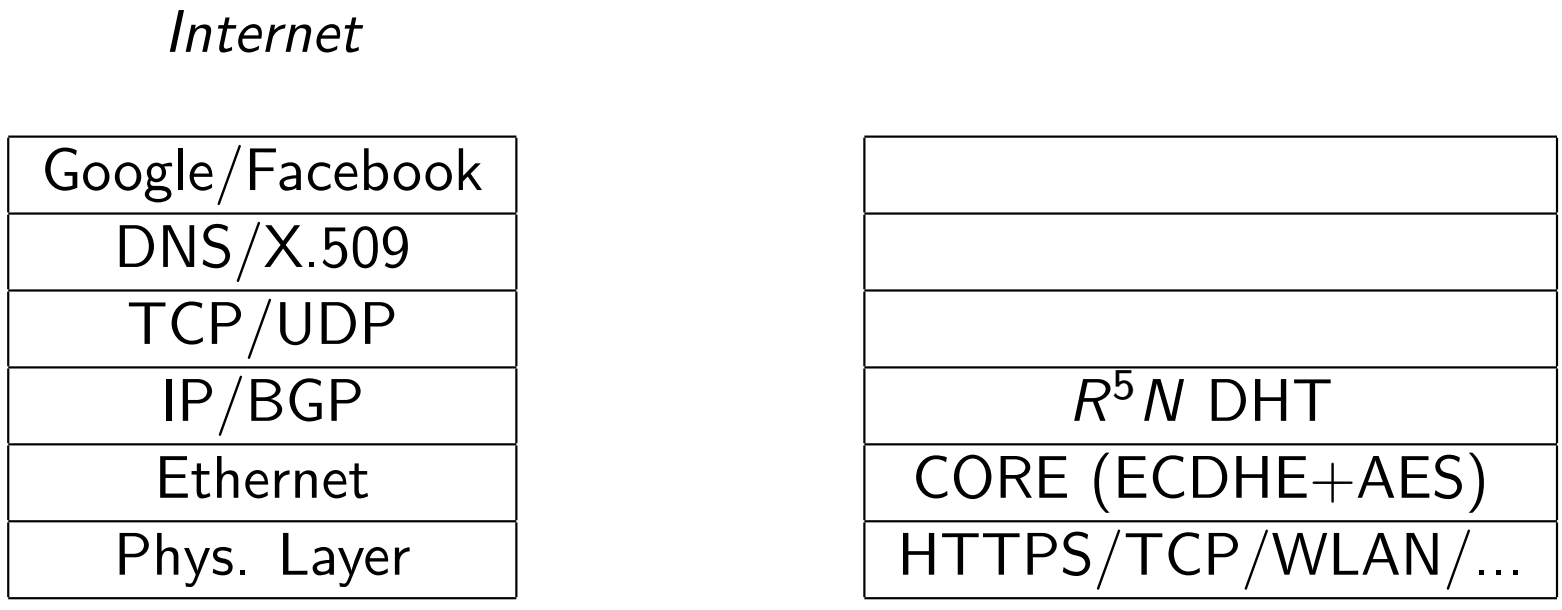
Our Vision


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
Ethernet
Phys. Layer
R^5N DHT
CORE (ECDHE+AES)
HTTPS/TCP/WLAN/...

Our Vision

Internet

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

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

GNS
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

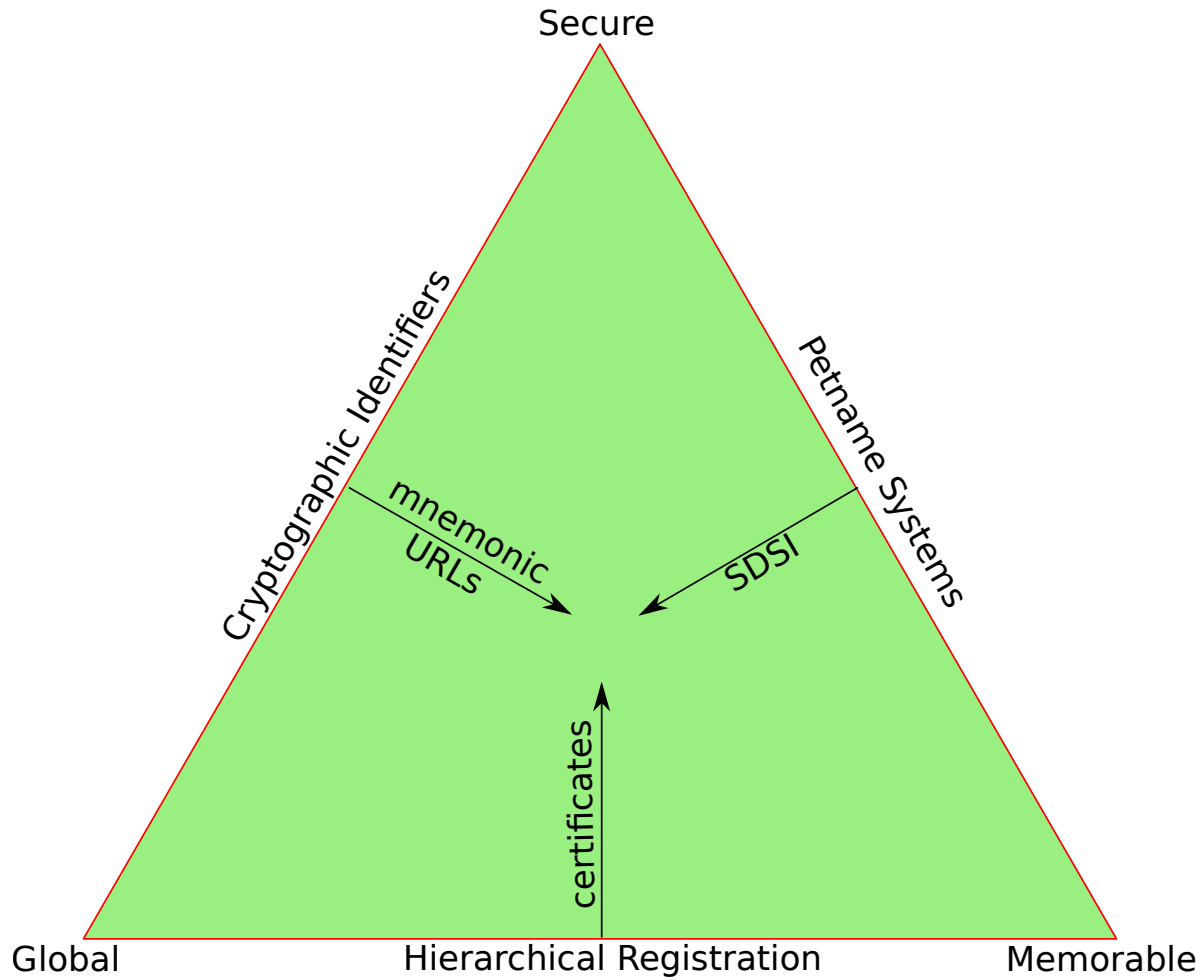
RegEx/PSYC
GNS
Mesh (ECDHE+AES)
R^5N DHT
CORE (ECDHE+AES)
HTTPS/TCP/WLAN/...

Our Vision

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

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

Decentralized Naming Systems[1]



Zooko's Triangle

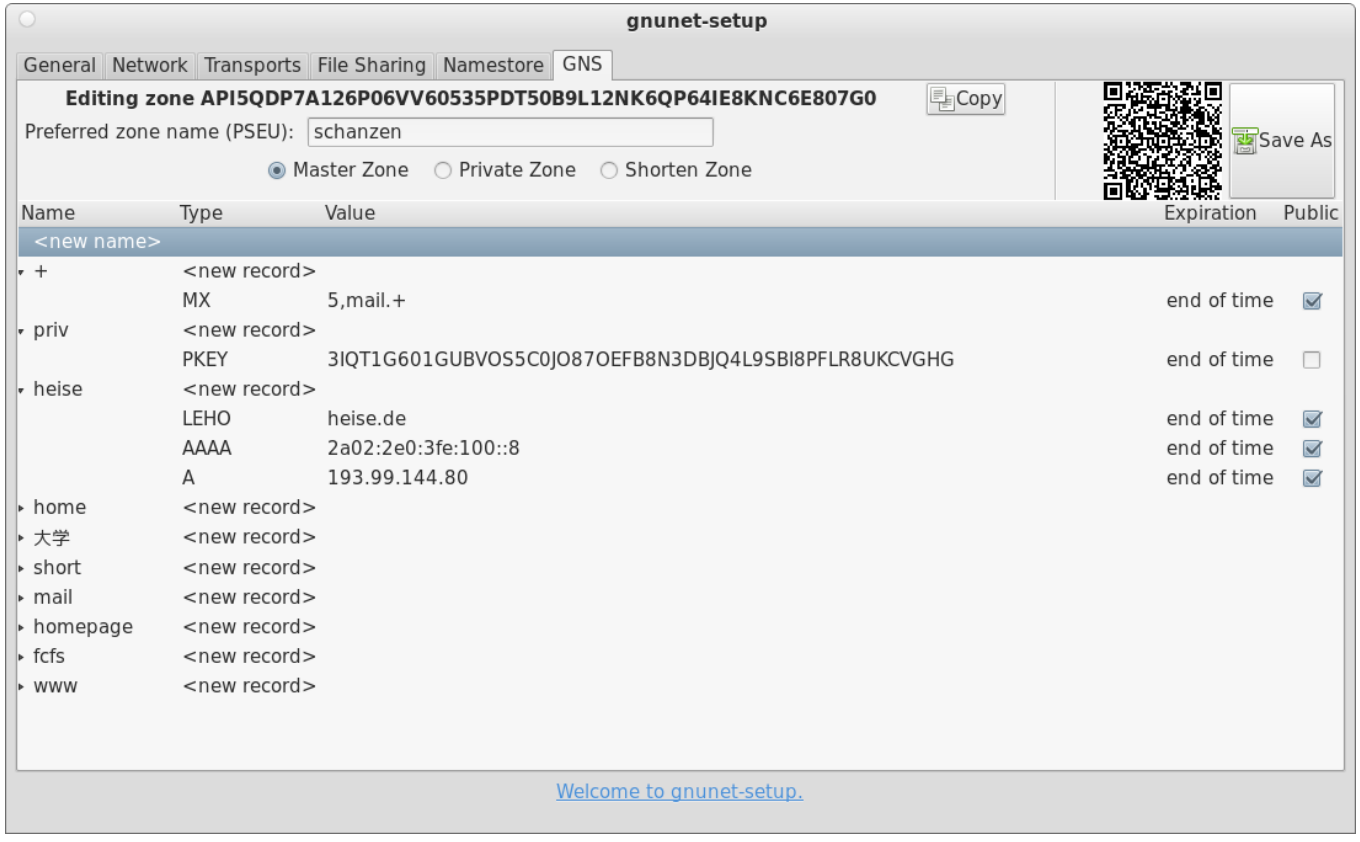
[1] Joint work with Martin Schanzenbach and Matthias Wachs

The GNU Name System (GNS)

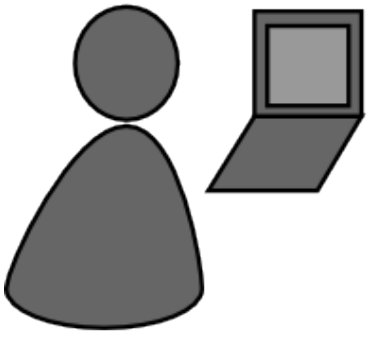
Decentralized PKI that can also replace DNS/DNSSEC:

- Signed Resource Records (RRs)
- Secure delegation provides **transitivity** (SDSI)
- Decentralized resolution (R^5N DHT)
- Every user manages his own zone

Zone Management: like in DNS

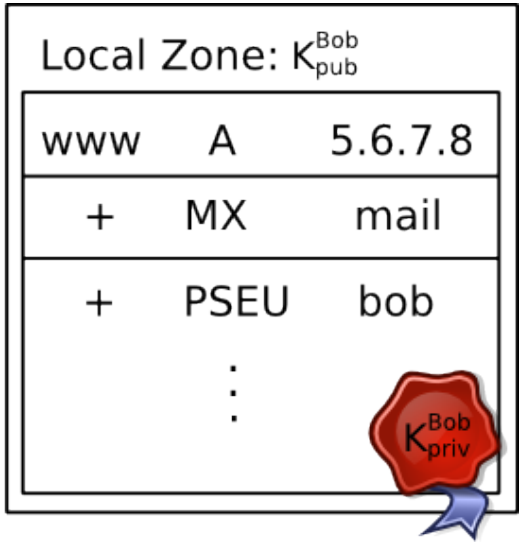


Name resolution in GNS



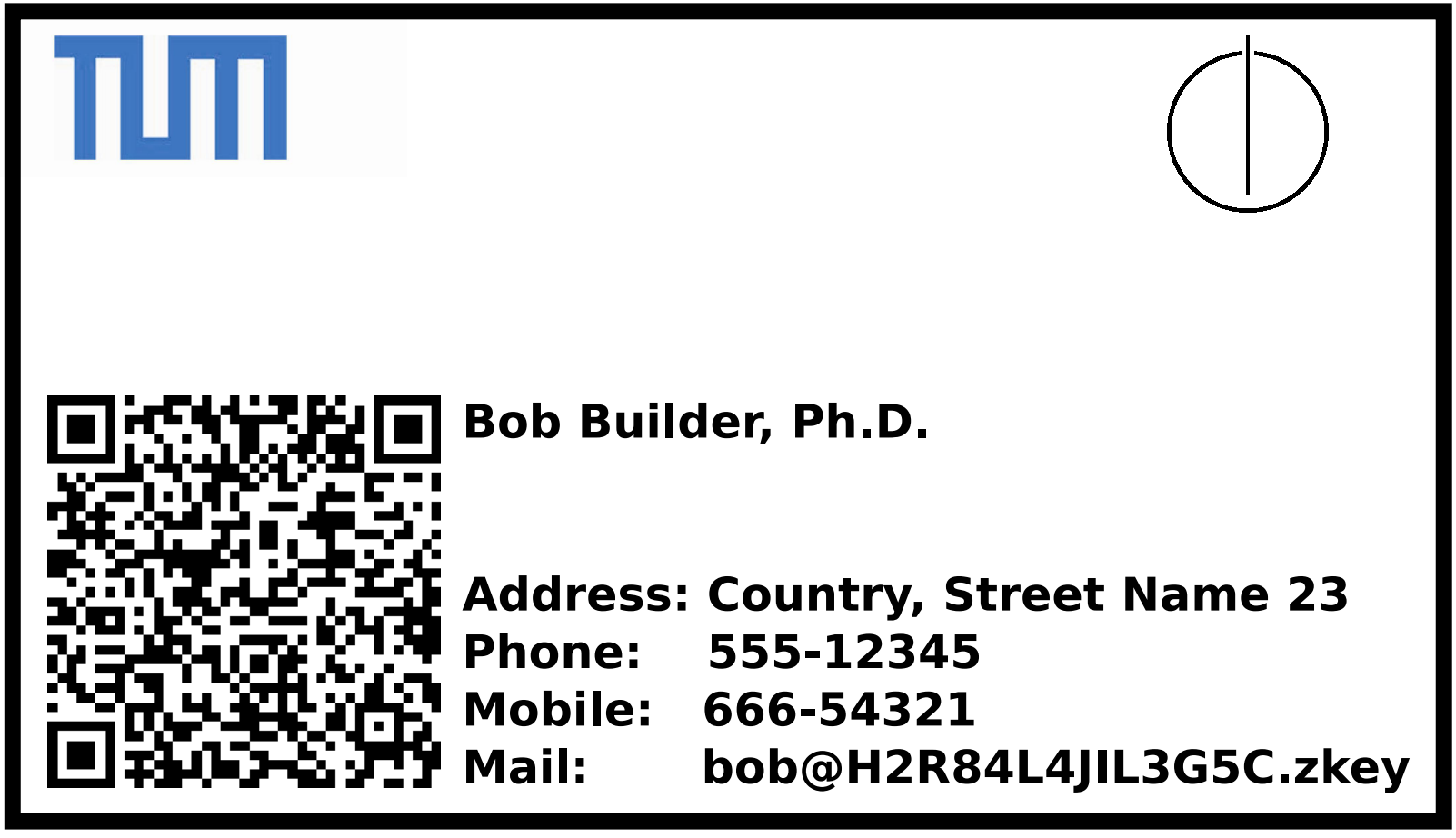
Bob

Bob's webserver



- Bob wants to be called **bob**
- Bob can reach his webserver via **www.gnu**

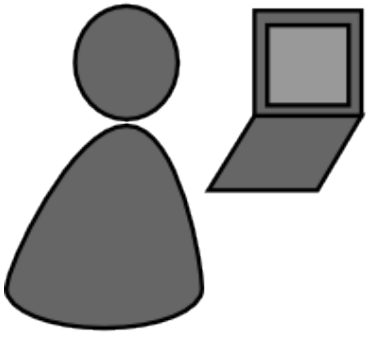
Secure introduction

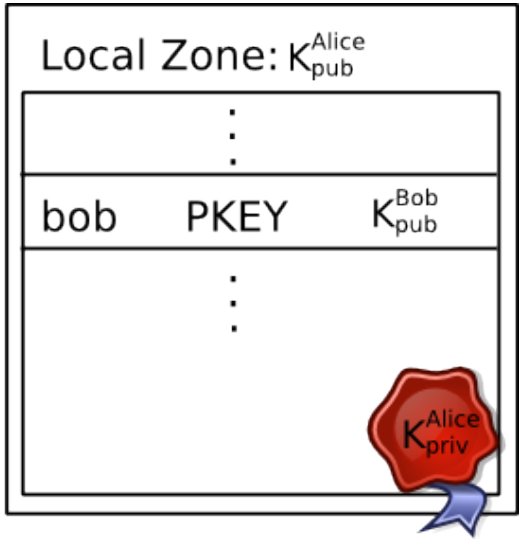


- Bob gives his public key to his **friends** via QR code

→ Bob's friends can resolve his records via *.*petname*.gnu

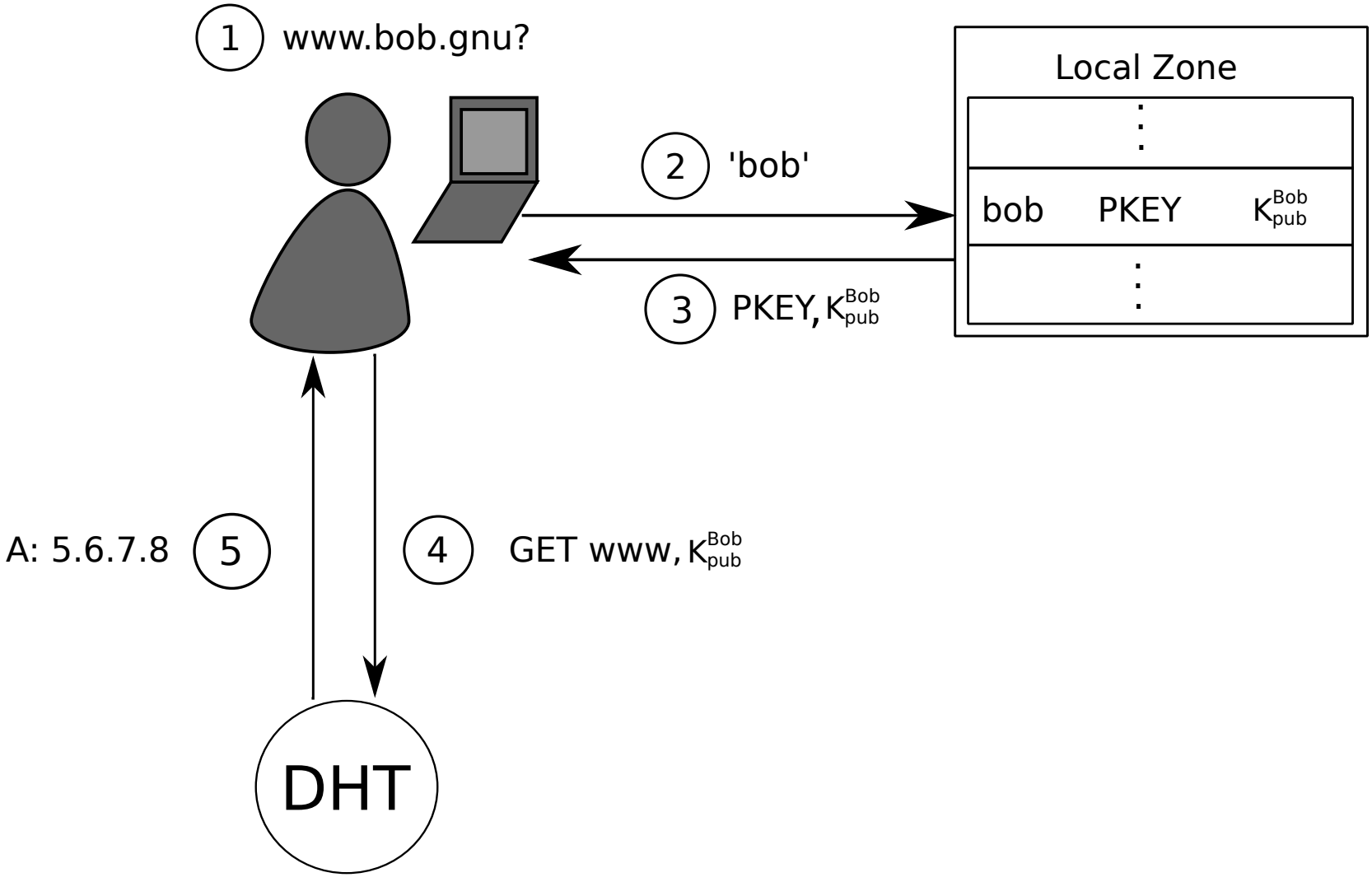
Delegation



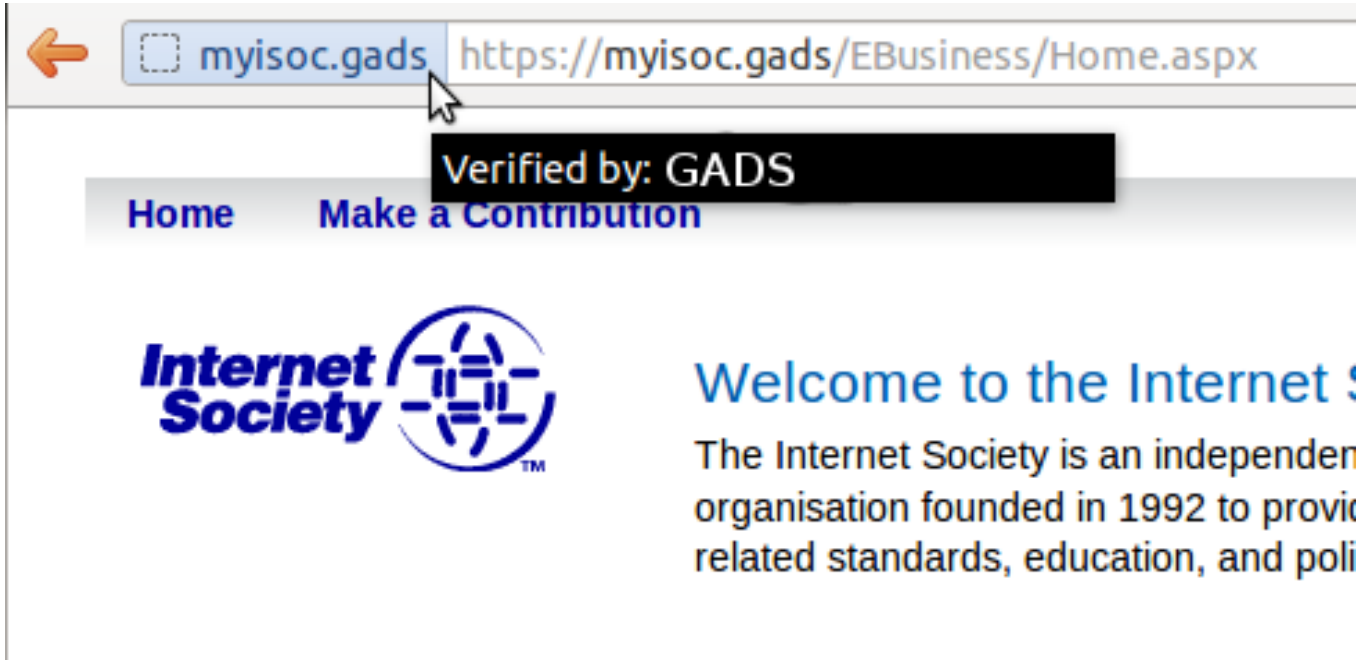


- Alice learns Bob's public key
- Alice creates delegation to zone **bob**
- Alice can reach Bob's webserver via **www.bob.gnu**

Name Resolution



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

$$h := H(I, P) \tag{1}$$
$$d := h \cdot x \mod n \tag{2}$$
$$B_{P,I} := S_d(E_{HKDF(I,P)}(R_{P,I})), dG \tag{3}$$

Query Privacy: Cryptography

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

$$h := H(I, P) \tag{1}$$
$$d := h \cdot x \mod n \tag{2}$$
$$B_{P,I} := S_d(E_{HKDF(I,P)}(R_{P,I})), dG \tag{3}$$

Searching for I in zone P

$$h = H(I, P) \tag{4}$$
$$q_{P,I} = H(dG) = H(hxG) = H(hP) \Rightarrow \texttt{obtain } B_{P,I} \tag{5}$$
$$R_{P,I} = D_{HKDF(I,P)}(B_{P,I}) \tag{6}$$

GNS for GNUnet

Properties of GNS

- Decentralized name system with secure memorable names
- Decentralized name system with globally unique, secure identifiers
- QR codes for introduction, delegation used to achieve transitivity
- Achieves query and response privacy except against confirmation attack
- Can provide alternative PKI, validate TLS via TLSA records

Uses for GNS in GNUnet

- Pseudonymous file-sharing
- IP services in the P2P network (P2P-VPN) via “VPN” records
- Identities in social networking applications

Our Vision

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

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

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: GNU libtool

Here are a set of rules to help you update your library version information:

1. Start with version information of 0:0:0 for each libtool library.
2. Update the version information only immediately before a public release of your software. More frequent updates are unnecessary, and only guarantee that the current interface number gets larger faster.
3. If the **library source code has changed** at all since the last update, then increment revision (c:r:a becomes c:r+1:a).
4. If any interfaces have been added, removed, or changed since the last update, increment current, and set revision to 0.
5. If any **interfaces have been added** since the last public release, then increment age.
6. If any **interfaces have been removed or changed** since the last public release, then set age to 0.

—taken from the GNU libtool manual.

Related Work: GNU libtool

There are three possible kinds of reactions from users of your library to changes in a shared library:

- Programs using the previous version may use the new version as **drop-in replacement**, and programs using the new version **can also work with the previous one**. In other words, no recompiling nor relinking is needed. In this case, bump revision only, don't touch current nor age.
- Programs using the previous version may use the new version as **drop-in replacement**, but programs using the new version may use **APIs not present in the previous one**. In other words, a program linking against the new version may fail with unresolved symbols if linking against the old version at runtime: set revision to 0, bump current and age.
- Programs may **need to be changed, recompiled, relinked** in order to use the new version. Bump current, set revision and age to 0.

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

PSYC2 for GNUnet

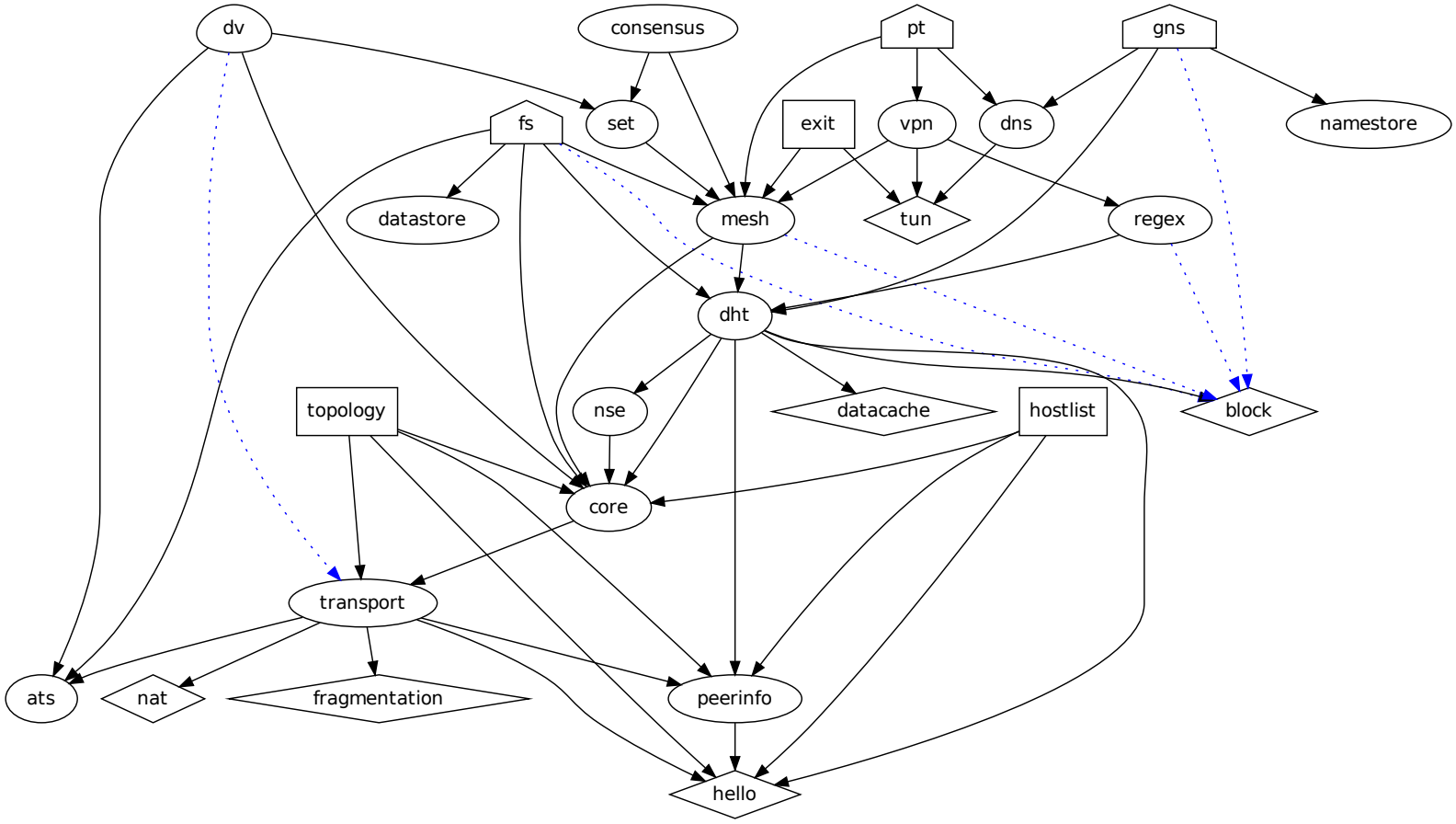
Properties of PSYC

- Compact encoding (much smaller than XML/JSON/XMPP)
- Supports stateful multicast
- Supports message history (replay, see latest news, etc.)
- Extensible syntax and semantics

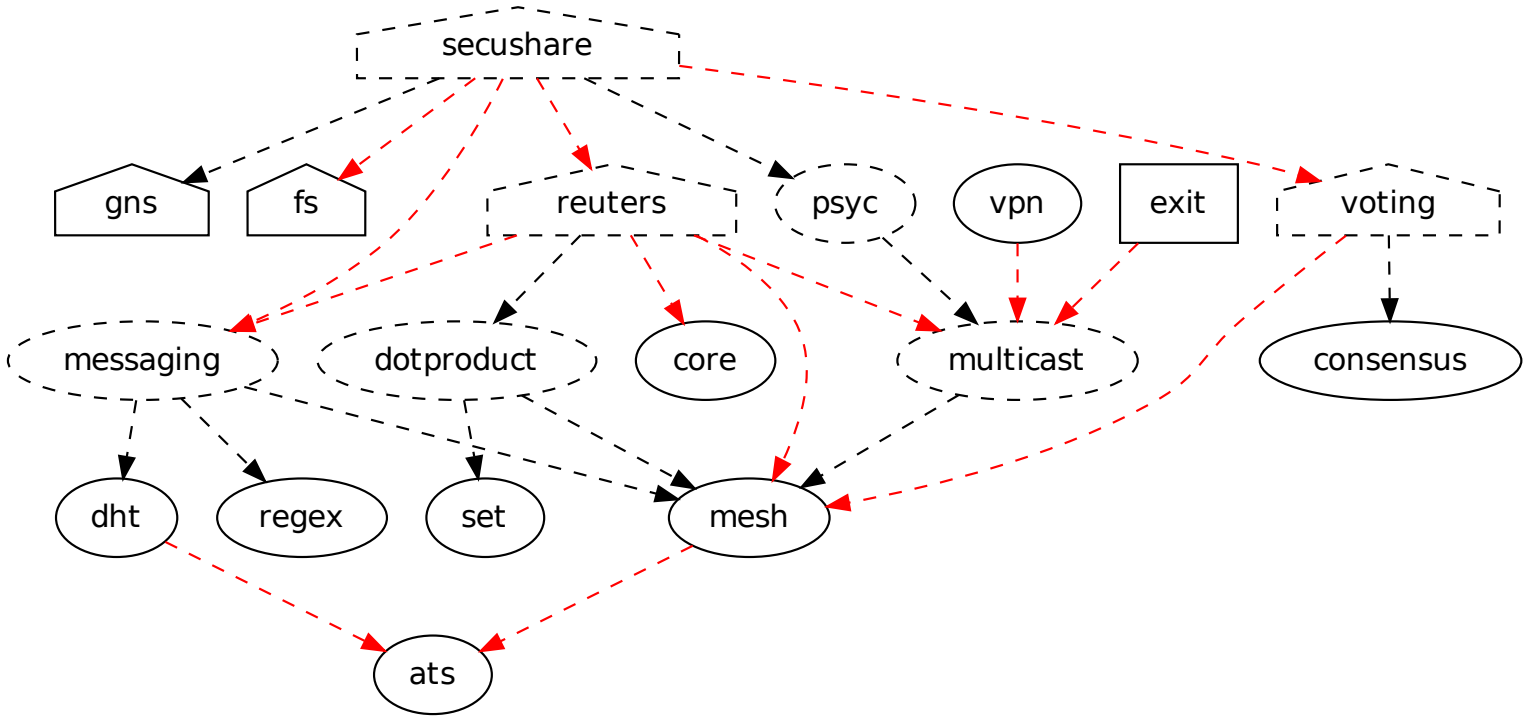
Uses for PSYC2 in GNUnet

- P2P social networking foundation (combine with GNS!)
- Pushes social profiles (state) to all recipients, no federation
- Replay from local database used as primary access method
- **My data is stored on my machine**
- Use secure multicast to support very large groups

GNUnet: Framework Architecture



GNUnet: Envisioned Applications



Conclusion

Snowden's disclosures show that we need GNUnet finished yesterday.

Conclusion

Snowden's disclosures show that we need GNUnet finished yesterday.

- Theoretical foundations: quite far
- System implementation: making progress
- User interfaces: could use help
- Packaging: coud use help (GUIX?)
- Documentation: could use help

We must decentralize or risk to loose control over our lives.

Do you have any questions?

References:

- 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.