The GNU Name System: A Public Key Infrastructure for Social Movements in the Age of Universal Surveillance

Christian Grothoff

The GNUnet Project

28.04.2017

“Never doubt your ability to change the world.” —Glenn Greenwald
The Internet

Virtually all Internet protocols are broken:

- **Ethernet**: MAC spoofing, cleartext
- **IP**: IP spoofing, cleartext
- **BGP**: AS hijacking, cleartext
- **DNS**: cache poisoning, cleartext
- **DNSSEC**: cleartext, often no end-to-end authentication
- **TLS**: 100 CAs can certify anybody for anything
- **HTTP**: too chatty, complex, slow

...
The Internet

Virtually all Internet protocols are broken:

- **Ethernet**  MAC spoofing, cleartext
- **IP**  IP spoofing, cleartext
- **BGP**  AS hijacking, cleartext
- **DNS**  cache poisoning, cleartext
- **DNSSEC**  cleartext, often no end-to-end authentication
- **TLS**  100 CAs can certify anybody for anything
- **HTTP**  too chatty, complex, slow

... 

Rule 1 for the GNUnet: Encrypt everything.
Encryption to the Rescue?

- Existing Internet PKIs are easily controlled:
  - DNSSEC root certificate
  - X.509 CAs (HTTPS certificates)
  - Major browser vendors (CA root stores!)
Encryption to the Rescue?

- Existing Internet PKIs are easily controlled:
  - DNSSEC root certificate
  - X.509 CAs (HTTPS certificates)
  - Major browser vendors (CA root stores!)
- Encryption does not help if PKI is compromised!
Encryption to the Rescue?

- Existing Internet PKIs are easily controlled:
  - DNSSEC root certificate
  - X.509 CAs (HTTPS certificates)
  - Major browser vendors (CA root stores!)
- Encryption does not help if PKI is compromised!
- PGP Web-of-Trust leaks social graph
How bad is it?
A DNS Lookup in 2014...
What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
A DNS Lookup in 2014...
What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
A DNS Lookup in 2014...
What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
A DNS Lookup in 2014...
What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
- NS of de is n.de.net
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of **net** is a.gtld-servers.net
- NS of **taler.net** is dns1.name-services.com
- NS of **com** is a.gtld-servers.net
- CNAME of **taler.net** is **pixel.net.in.tum.de**
- NS of **de** is n.de.net
- NS of **net** was a.gtld-servers.net
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
- NS of de is n.de.net
- NS of net was a.gtld-servers.net
- NS of de.net is ns1.denic.de
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of **net** is a.gtld-servers.net
- NS of **taler.net** is dns1.name-services.com
- NS of **com** is a.gtld-servers.net
- CNAME of **taler.net** is **pixel.net.in.tum.de**
- NS of **de** is n.de.net
- NS of **net** was a.gtld-servers.net
- NS of **de.net** is ns1.denic.de
- NS of **tum.de** is dns1.lrz.de
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
- NS of de is n.de.net
- NS of net was a.gtld-servers.net
- NS of de.net is ns1.denic.de
- NS of tum.de is dns1.lrz.de
- NS of lrz.de is dns1.lrz.de
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of **net** is a.gtld-servers.net
- NS of **taler.net** is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of **taler.net** is **pixel.net.in.tum.de**
- NS of **de** is n.de.net
- NS of **net** was a.gtld-servers.net
- NS of **net**.net is ns1.denic.de
- NS of **tum.de** is dns1.lrz.de
- NS of **lrz.de** is dns1.lrz.de
- NS of **in.tum.de** is tuminfo1.informatik.tu-muenchen.de
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
- NS of de is n.de.net
- NS of net was a.gtld-servers.net
- NS of de.net is ns1.denic.de
- NS of tum.de is dns1.lrz.de
- NS of lrz.de is dns1.lrz.de
- NS of in.tum.de is tuminfo1.informatik.tu-muenchen.de
- NS of tu-muenchen.de is ws-han1.wip-ip.dfn.de
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
- NS of de is n.de.net
- NS of net was a.gtld-servers.net
- NS of de.net is ns1.denic.de
- NS of tum.de is dns1.lrz.de
- NS of lrz.de is dns1.lrz.de
- NS of in.tum.de is tuminfo1.informatik.tu-muenchen.de
- NS of tu-muenchen.de is ws-han1.wip-ip.dfn.de
- NS of dfn.de is ws-han1.wip-ip.dfn.de
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
- NS of de is n.de.net
- NS of net was a.gtld-servers.net
- NS of de.net is ns1.denic.de
- NS of tum.de is dns1.lrz.de
- NS of lruz.de is dns1.lrz.de
- NS of in.tum.de is tuminfo1.informatik.tu-muenchen.de
- NS of tu-muenchen.de is ws-han1.wip-ip.dfn.de
- NS of dfn.de is ws-han1.wip-ip.dfn.de
- NS of net.in.tum.de is dns1.lrz.de
A DNS Lookup in 2014...

What would a simple DNS lookup do? Say for taler.net?

- NS of net is a.gtld-servers.net
- NS of taler.net is dns1.name-services.com
- NS of com is a.gtld-servers.net
- CNAME of taler.net is pixel.net.in.tum.de
- NS of de is n.de.net
- NS of net was a.gtld-servers.net
- NS of de.net is ns1.denic.de
- NS of tum.de is dns1.lrz.de
- NS of lrz.de is dns1.lrz.de
- NS of in.tum.de is tuminfo1.informatik.tu-muenchen.de
- NS of tu-muenchen.de is ws-han1.wip-ip.dfn.de
- NS of dfn.de is ws-han1.wip-ip.dfn.de
- NS of net.in.tum.de is dns1.lrz.de
- A of pixel.net.in.tum.de is 131.159.20.32
Exemplary Attacks: MORECOWBELL

(U) How Does it Work?

- (U) Consists of:
  - (U//FOUO) Central tasking system housed in V43 office Spaces
  - (S//REL) Several covertly rented web servers (referred to as bots) in: Malaysia, Germany, and Denmark
- (S//REL) The MCB bots utilize open DNS resolvers to perform thousands of DNS lookups every hour.
- (S//REL) MCB bots have the ability to perform HTTP GET requests (mimicking a user’s web browser)
- (S//REL) The data is pulled back to the NSA every 15-30 minutes
- (S//REL) Data Currently available on NSANet via web services

TOP SECRET//COMINT//REL FVEY
Exemplary Attacks: QUANTUMDNS

- (TS//SI//REL) QUANTUMBISCUIT
  - Redirection based on keyword
  - Mostly HTML Cookie Values

- (TS//SI//REL) QUANTUMDNS
  - DNS Hijacking
  - Caching Nameservers

- (TS//SI//REL) QUANTUMBOT2
  - Combination of Q-BOT/Q-BISCUIT for web based
    Command and controlled botnets
Query Name Minimization

Stub Resolver

Recursive Name Server

www.example.com? A 93.184.216.119

NS a.gtld-servers.net.

DNS Server .com a.gtld-servers.net

DNS Server Root Zone a.root-servers.net.

DNS Server example.com? NS example.com?

NS a.iana-servers.net.

DNS Server example.com

DNS Server example.com a.iana-servers.net

A 93.184.216.119

9 / 51
DNS over TLS

Stub Resolver

Recursive Name Server

DNS Server Root Zone
a.root-servers.net.

DNS Server .com
a.gtld-servers.net.

DNS Server example.com
a.iana-servers.net.

www.example.com?
A 93.184.216.119

www.example.com?
NS a.gtld-servers.net.

www.example.com?
NS a.iana-servers.net.

www.example.com?
A 93.184.216.119

www.example.com?
The Textbook Version of the Internet

Layering, $\approx$ 1990

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTPS</td>
<td></td>
</tr>
<tr>
<td>DNS</td>
<td>TLS</td>
</tr>
<tr>
<td>UDP</td>
<td>TCP</td>
</tr>
<tr>
<td>IPv4</td>
<td></td>
</tr>
<tr>
<td>Ethernet</td>
<td></td>
</tr>
<tr>
<td>Phys. Layer</td>
<td></td>
</tr>
</tbody>
</table>
The Textbook Version of the Internet

### Layering, ≈ 1990

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocol 1</th>
<th>Protocol 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phys. Layer</td>
<td>HTTPS</td>
<td></td>
</tr>
<tr>
<td>Ethernet</td>
<td>DNS</td>
<td>TLS</td>
</tr>
<tr>
<td>IPv4</td>
<td>TCP</td>
<td></td>
</tr>
</tbody>
</table>

### “Layering”, ≈ 2020

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocol 1</th>
<th>Protocol 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phys. Layer</td>
<td>HTTPS</td>
<td>libmicrohttpd</td>
</tr>
<tr>
<td>Ethernet</td>
<td>TLS-with-DANE</td>
<td>libgnutls</td>
</tr>
<tr>
<td>IPv6</td>
<td>DNS-over-TLS</td>
<td>libunbound</td>
</tr>
<tr>
<td>TCP</td>
<td>TLS*</td>
<td>libnss</td>
</tr>
<tr>
<td>IPv4</td>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Ethernet</td>
<td>Linux</td>
<td></td>
</tr>
</tbody>
</table>

* = castrated version without RFC 6125 or RFC 6394, possibly NULL cipher, see TLS profiles draft.
DNSCurve

DNS Curve Cache
Public Key $P_c$
Private Key $S_c$

DNS Server
Root Zone a.root-servers.net.

DNS Server .com a.gtld-servers.net.

DNSCurve Server example.com uz5...hyw.iana-servers.net.

www.example.com?
NS a.gtld-servers.net.

www.example.com?
NS uz5...hyw.iana-servers.net.

$P_c$, $N$, $E$ (www.example.com?)

$N$, $E$ (A 93.184.216.119)
Namecoin

Diagram:
- Namecoin Client
- Local Copy of Block Chain
- P2P Network
- Block Chain

Actions:
- Append registration to block chain
- Get copy of block chain
A name system can only fulfill two!
Zooko’s Triangle

DNS, “.onion” IDs and /etc/hosts/ are representative designs.
Zooko’s Triangle

DNSSEC security is broken by design (adversary model!)
Namecoin
Namecoin

- Memorable:
Namecoin

- Memorable: Check
- Global:
Namecoin

- Memorable: Check
- Global: Check
- Secure:
Namecoin

- Memorable: Check
- Global: Check
- Secure: different adversary model!
Namecoin

- Memorable: Check
- Global: Check
- Secure: different adversary model!

⇒ Availability of names (registration rate) is restricted
Namecoin

- Memorable: Check
- Global: Check
- Secure: different adversary model!

⇒ Availability of names (registration rate) is restricted
⇒ Adversary must not have 51% compute power
The GNU Name System

Properties of GNS

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

\[ \text{Joint work with Martin Schanzenbach and Matthias Wachs} \]
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></td>
<td>5, mail. +</td>
<td>end of time</td>
<td></td>
</tr>
<tr>
<td>priv</td>
<td></td>
<td>3/QT1G601GUBVOSSC0JO87QEFB8N3DBJQ4L95BI8PFLR8UKCVGHG</td>
<td>end of time</td>
<td></td>
</tr>
<tr>
<td>heise</td>
<td></td>
<td>leise.de</td>
<td>end of time</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2a02:2e0:3fe:100::8</td>
<td>end of time</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>193.99.144.80</td>
<td>end of time</td>
<td></td>
</tr>
<tr>
<td>home</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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 $K^{Bob}_{pub}$ under label `bob`
- Alice can reach Bob’s webserver via `www.bob.gnu`
Name Resolution

Bob

DHT

Alice

Bob

<table>
<thead>
<tr>
<th>8FS7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>www  A  5.6.7.8</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Alice

<table>
<thead>
<tr>
<th>A47G</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>bob  PKEY  8FS7</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Name Resolution

Bob

DHT

Alice

PUT 8FS7-www: 5.6.7.8

Bob

Alice

8FS7

A

www  A  5.6.7.8

A47G

bob  PKEY  8FS7
Name Resolution

1. www.bob.gnu?

Bob Alice
DHTPUT 8FS7-www: 5.6.7.8
0
...
...
www A 5.6.7.8
...

Bob

Alice

A47G

bob PKEY 8FS7

25 / 51
Name Resolution

1. Bob performs a PUT operation to store the mapping of "8FS7-www" to IP address 5.6.7.8 in the DHT.

2. Alice queries the DHT for the mapping of "bob".

3. The DHT returns the mapping details to Alice, which she stores in her local database.

The diagram illustrates the interaction between Bob and Alice in a distributed hash table (DHT) network, where Bob stores a resource and Alice retrieves it using a query.
Name Resolution

Bob

PUT 8FS7-www: 5.6.7.8

DHT

www      A      5.6.7.8
8FS7
Bob

Alice

PKEY 8FS7!

'bob'?
Name Resolution

Bob

8FS7

www A 5.6.7.8

Alice

A47G

bob PKEY 8FS7

0 PUT 8FS7-www: 5.6.7.8

DHT

1 www.bob.gnu ?

2 'bob'?

3 PKEY 8FS7!

4 8FS7-www?
Name Resolution

Bob

8FS7

www    A    5.6.7.8

Alice

A47G

bob    PKEY    8FS7

1. www.bob.gnu?
2. 'bob'?
3. PKEY 8FS7!
4. 8FS7-www?
5. A 5.6.7.8!

PUT 8FS7-www: 5.6.7.8
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 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 "gnew", as one syllable, like saying "grew" but replacing the r with n.

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.
Privacy Issue: DHT

1. Bob PUT 8FS7-www: 5.6.7.8
2. Alice 'bob'?
3. Alice PKEY 8FS7!
4. Bob 8FS7-www?
5. Alice A 5.6.7.8!

Bob

<table>
<thead>
<tr>
<th>8FS7</th>
</tr>
</thead>
<tbody>
<tr>
<td>www</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>

Alice

<table>
<thead>
<tr>
<th>A47G</th>
</tr>
</thead>
<tbody>
<tr>
<td>bob</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>
Query Privacy: Terminology

\( G \) generator in ECC curve, a point

\( n \) size of ECC group, \( n := |G|, n \text{ prime} \)

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

\( P \) public key of zone, a point \( P := xG \)

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

\( R_{P,l} \) set of records for label \( l \) in zone \( P \)

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

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

Publishing records $R_{P,l}$ as $B_{P,l}$ under key $q_{P,l}$

$$h := H(l, P) \quad \text{(1)}$$
$$d := h \cdot x \mod n \quad \text{(2)}$$
$$B_{P,l} := S_d(E_{HKDF(l,P)}(R_{P,l})), dG \quad \text{(3)}$$
$$q_{P,l} := H(dG) \quad \text{(4)}$$
Query Privacy: Cryptography

Publishing records $R_{P,l}$ as $B_{P,l}$ under key $q_{P,l}$

$$h := H(l, P)$$ (1)
$$d := h \cdot x \mod n$$ (2)
$$B_{P,l} := S_d(E_{HKDF(l,P)}(R_{P,l})), dG$$ (3)
$$q_{P,l} := H(dG)$$ (4)

Searching for records under label $l$ in zone $P$

$$h := H(l, P)$$ (5)
$$q_{P,l} := H(hP) = H(hxG) = H(dG) \Rightarrow \text{obtain } B_{P,l}$$ (6)
$$R_{P,l} = D_{HKDF(l,P)}(B_{P,l})$$ (7)
The GNU Name System (GNS)

Bob's NSS
\[\text{.gnu} = P_{\text{bob}}\]

Prune:  \[\text{www}.P_{\text{bob}}?\]
[0.000, 0.000]  \[\text{P} 203.0.113.54\]

Bob's GNS Service

Carols's NSS
\[\text{.gnu} = P_{\text{bob}}\]

Prune:  \[H(\text{www}, P_{\text{bob}}), E(A 203.0.113.54)]\n
Carols's GNS Service

\[\text{P}_{\text{bob}}\text{ zone database}\]
[0.000, 0.000]  \[
\begin{array}{c}
\text{carol} \quad \text{PKEY} \\
\text{www} \quad A \\
203.0.113.54
\end{array}
\]

P2P Network

\[\text{DHT}\]

\[\text{P}_{\text{bob}}\text{ zone database}\]
[0.000, 0.000]  \[
\begin{array}{c}
\text{carol} \quad \text{PKEY} \\
\text{www} \quad A \\
203.0.113.54
\end{array}
\]

\[\text{P}_{\text{bob}}\text{ zone database}\]
[0.000, 0.000]  \[
\begin{array}{c}
\text{carol} \quad \text{PKEY} \\
\text{www} \quad A \\
203.0.113.54
\end{array}
\]

Alice's NSS
\[\text{.gnu} = P_{\text{alice}}\]

Prune:  \[\text{www}.P_{\text{alice}}?\]
[0.000, 0.000]  \[\text{A} 203.0.113.34\]

Alice's GNS Service

\[\text{P}_{\text{alice}}\text{ zone database}\]
[0.000, 0.000]  \[
\begin{array}{c}
\text{bob} \quad \text{PKEY} \\
\text{www} \quad A \\
203.0.113.13
\end{array}
\]

P2P Network

\[\text{DHT}\]

\[\text{P}_{\text{alice}}\text{ zone database}\]
[0.000, 0.000]  \[
\begin{array}{c}
\text{bob} \quad \text{PKEY} \\
\text{www} \quad A \\
203.0.113.13
\end{array}
\]

\[\text{P}_{\text{alice}}\text{ zone database}\]
[0.000, 0.000]  \[
\begin{array}{c}
\text{bob} \quad \text{PKEY} \\
\text{www} \quad A \\
203.0.113.13
\end{array}
\]
Revocation

Revocation Basics

- Revocation certificate (RC): message signed with private key
- Peer receives new valid RC, floods to all neighbours
- All peers store all valid RCs forever

⇒ Expensive operation ⇒ proof-of-work
Revocation

Revocation Basics

- Revocation certificate (RC): message signed with private key
- Peer receives new valid RC, floods to all neighbours
- All peers store all valid RCs forever
  ⇒ Expensive operation ⇒ proof-of-work

Revocation Magic

- Peers maybe offline during initial flood
- Network might be temporarily partitioned
  ⇒ Need to reconcile revocation sets on connect

Whenever two peers establish a P2P connection, they must compute the set union of their RC sets!
The “.zkey” pTLD

- “LABELS.PKEY.zkey” format
- PKEY is the public key of the zone
- Works a bit like “.onion”
  ⇒ Globally unique identifiers!
“alice.bob.carol.dave.gnu” is a bit long for Edward (“.gnu”)
Also, we need to trust Bob, Carol and Dave (for each lookup)
Finally, Alice would have liked to be called Krista (just Bob calls her Alice)
NICKnames

▶ “alice.bob.carol.dave.gnu” is a bit long for Edward (“.gnu”)
▶ Also, we need to trust Bob, Carol and Dave (for each lookup)
▶ Finally, Alice would have liked to be called Krista (just Bob calls her Alice)
▶ “NICK” records allow Krista to specify her preferred NICKname
▶ GNS adds a “NICK” record to each record set automatically
▶ Eve learns the “NICK”, and GNS creates “krista.short.gnu”
NICKnames

- “alice.bob.carol.dave.gnu” is a bit long for Edward (“.gnu”)
- Also, we need to trust Bob, Carol and Dave (for each lookup)
- Finally, Alice would have liked to be called Krista (just Bob calls her Alice)
- “NICK” records allow Krista to specify her preferred NICKname
- GNS adds a “NICK” record to each record set automatically
- Eve learns the “NICK”, and GNS creates “krisa.short.gnu”
- Memorable, short trust path in the future! TOFU!
- Krista better pick a reasonably unique NICK.
Shadow Records

- Records change
- Expiration time controls validity, like in DNS
- DHT propagation has higher delays, compared to DNS
Shadow Records

- Records change
- Expiration time controls validity, like in DNS
- DHT propagation has higher delays, compared to DNS
- SHADOW is a flag in a record
- Shadow records are only valid if no other, non-expired record of the same type exists
Practical Concerns

- Name registration
- Support for browsing
- New record types
- Integration with applications
- State of the implementation
Registering a name in GNS

▶ Bob gives his PKEY to his friends via QR code

▶ or registers it at the GNUnet fcfs authority *pin.gnu as ”bob”

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

▶ → or *.bob.pin.gnu
From DNS to GNS

Names are not globally unique, but ...

... we need support for Virtual Hosting!
... we need support for SSL!
From DNS to GNS

Names are not globally unique, but ...  
... we need support for Virtual Hosting!  
... we need support for SSL!

Solution: Client Side SOCKS Proxy
Legacy Hostname (LEHO) Records

LEHO records give a hint about the DNS name the server expects.
Legacy Hostname (LEHO) Records

LEHO records give a hint about the DNS name the server expects.

![Diagram showing HTTP GET interactions between Dave, Alice, and the Server]
Long-Term Vision

- Integration with browser and HTTP server
- HTTP server receives “GNS-Zone: PKEY” instead of “Hostname”
- HTTP client uses “TLSA” record of GNS, instead of “LEHO”
Relative Names

- GNS records can contain “.+”
- CNAME: “server1.+”
- MX: “mail.+”
- “.+” stands for “relative to current zone”

Supporting this for links in browsers would be nice, too.
New Record Types

- PKEY: delegate to another GNS zone
- NICK: preferred names for shortening
- LEHO: legacy hostname
New Record Types

- PKEY: delegate to another GNS zone
- NICK: preferred names for shortening
- LEHO: legacy hostname
- GNS2DNS: delegate to DNS
- VPN: peers hosting TCP/IP services
- PHONE: call users using gnunet-conversation
DNS Delegation

- Delegate to DNS using GNS2DNS records
- GNS2DNS record specifies:
  - Name of DNS resolver (i.e. “ns1.example.com” or “piratedns.+”)
  - DNS domain to continue resolution in (i.e. “example.com” or “piratebay.org”)
- GNS will first resolve DNS resolver name to A/AAAA record
- GNS will then resolve “left.of.gns2dns.example.com” using DNS
VPN Delegation

- Delegates to GNUnet VPN
- VPN record specifies:
  - Identity of hosting peer (no anonymity!)
  - Service identifier (hash code)
- GNS can map VPN record to A/AAAA record of gnunet-vpn tunnel
PHONE service

- PHONE record specifies:
  - Identity of hosting peer (no anonymity yet!)
  - Line number (to support multiple phones per peer)
Application Integration

- SOCKS proxy (gnunet-gns-proxy)
- NSS plugin
- DNS packet interception (gnunet-dns-service)
- GNS (C) API
- GNS (IPC) protocol
- GNS command-line tool
Current State

- GNS part of GNUnet since 0.9.3
- Crypto changed to Curve25519 in 0.10.0
- Internationalized Domain Names are supported
Current State

- GNS part of GNUnet since 0.9.3
- Crypto changed to Curve25519 in 0.10.0
- Internationalized Domain Names are supported
- Installation is “non-trivial” (for your parents)
- Needs more work on reverse lookup
# Privacy summary

<table>
<thead>
<tr>
<th>Method</th>
<th>Defense against MiTM</th>
<th>Zone privacy</th>
<th>Privacy vs. network</th>
<th>Traffic amplification resistance</th>
<th>Censorship resistance</th>
<th>Ease of migration</th>
<th>*EDNS0</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>DNSSEC</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>DNSCurve</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>DNS-over-TLS</td>
<td>✓</td>
<td>n/a</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Namecoin</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>GNS</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
</tbody>
</table>
## Key management summary

<table>
<thead>
<tr>
<th></th>
<th>Suitable for personal use</th>
<th>Memorable</th>
<th>Decentralised</th>
<th>Modern cryptography</th>
<th>Understandable</th>
<th>Exposes metadata</th>
<th>Transitive</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>DNSSEC</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>DNSCurve</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>DNS-over-TLS</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>TLS-X.509</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Web of Trust</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>TOFU</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>SMP/PANDA</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Namecoin</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GNS</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Conclusion

- We have decentralized the PKI
- Privacy and security are preserved
Conclusion

- We have decentralized the PKI
- Privacy and security are preserved
Do you have any questions?

References:

