The Evolving Architecture of the Web

Nick Sullivan
Head of Cryptography

CFSSL
Universal SSL
Keyless SSL
Privacy Pass
Geo Key Manager

Recently
Standards work
TLS 1.3
Competing Goals

make browsing more

private

performant
HTTP
DNS
Geographically Centralized
Administratively Diverse

One IP per Hostname
What a network observer can see

- Unique Client IP
- Unique Server IP
- Server URL
- Website content
Anonymity set

<table>
<thead>
<tr>
<th>Client IP</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server IP</td>
<td>1</td>
</tr>
</tbody>
</table>
IPv4
4.3 Billion Addresses
Not enough for every user
What a network observer can see

- Client Proxy IP
- Unique Server IP
- Server URL
- Website content

HTTP →

Clients
Hosts
Proxy
Latency Cost

Tor
3 round-the-world

VPN
1 round-the-world

Carrier NAT
Small

3 round-the-world

1 round-the-world
TLS 1.3: coming soon
What a network observer can see

HTTP → HTTPS

Unique Client IP
Unique Server IP
Server URL
Website content

Clients
Hosts
Client 1
Server 1

Anonymity set 1
IPv4
4.3 Billion Addresses
Not enough for every website
Geographically Centralized
Administratively Diverse

HTTP →

Clients
Hosts

One IP per Hostname
More Geographically Centralized
More Administratively Centralized

HTTP ➔ HTTPS

Clients
Hosts
Shared Hosts

Multiple Hostname per IPs
SNI Virtual Hosting

Send the hostname to the server so it can choose the certificate
What a network observer can see

HTTP → HTTPS

Client Unique IP
Shared Server IP
Hostname

Clients
Hosts
Shared Hosts
Client
Server
(Shared IP+Hostname)

Anonymity set
1
1
Internet Scans and IPv6
Privacy Evolves
Certificate Transparency
Wildcard certificates
Edge Services
Edge Services

- Websites and are delegating to globally distributed parties
- Authorized to terminate TLS
- Reduced Latency
- Improved DDoS resilience
- Anycast to reduce number of IPs needed
More Geographically Centralized
More Administratively Centralized

Multiple IPs per Hostname
Geographically Distributed
Administratively Centralized

Multiple IPs per Hostname
HTTPS

Client

ISP

Host

Browser

Operating System
Questions

Can we improve *privacy*?
Can we improve *latency*?
Can we improve *both*???
HTTP 1.1

Client

Operating System

Browser

Resolver

Edge

SNI: burrito.com

Q: burrito.com A: 1.2.3.4

SNI: beans.com

Q: beans.com A: 1.2.3.5

burrito.com

beans.com
Safety in Numbers
Client

Operating System

Browser

SNI: burrito.com

Q: burrito.com A: 1.2.3.4

Resolver

Edge

burrito.com

Host

Meek

Origin

beans.com

Host
Meek

Mismatch: SNI, Host, SAN
Connection Coalescing
Client

Operating System

Browser

HTTP/2

Resolver

Edge

SNI: burrito.com

GET https://burrito.com

ORIGIN: beans.com

GET https://beans.com

Q: burrito.com A: 1.2.3.4

ORIGIN Frame
What a network observer can see

HTTP → HTTPS

- Client Unique IP
- Shared Server IP
- First Hostname

Clients
Hosts
Anycast Hosts
Client
Server
(Shared IP+Certificate)

Anonymity set

1
~20
SNI: burrito.com

Q: burrito.com A: 1.2.3.4
What this changes

Having a certificate gives you routing authority
Anonymity set

Client
Server

(Shared IP+First Hostname)

\[ k \]

\[ 1 \]

\[ k \]

\[ 1 \]

\[ k \]

\[ 1 \]

\[ k \]

\[ 1 \]

k is the set of domains on certificates that can be obtained through “First Hostname”
Meek-like circumvention protection

Only send the CERTIFICATE frame on certain resources
QNAME Minimization
Latency

ISPs - Closer to user
Smaller cache
UDP

Edge DoH - Globally Distributed
TLS 1.3 0RTT
Challenges in the Enterprise
HTTP/2

Client

GET https://burrito.com

ORIGIN: beans.com
CERTIFICATE: beans.com

GET https://beans.com

SNI: resolver.com
Q: burrito.com A: 1.2.3.4

resolver.com

ORIGIN/CERT + DoH

DoH

Resolver

Edge

burrito.com
HTTP/2

SNI: burrito.com

SNI: resolver.com

ORIGIN/CERT + DoH
Encrypt SNI with client ephemeral key + server public key from DNS
TLS 1.3

eSNI: E(burrito.com)

SNI: resolver.com
Q: burrito.com A: 1.2.3.4, PubKey

O/C + DoH + eSNI
What a network observer can see

HTTP → HTTPS

Client Unique IP
Shared Server IP
First Hostname (SNI)

Clients
Hosts
Anycast Hosts
K is the set of domains that can be served on the IP.

**Caveat:** If Server IP is static, then this gives a hint about the first hostname.
HTTP/2

SNI: resolver.com
Q: beans.com A: 1.2.3.5
ORIGIN: beans.com CERTIFICATE: beans.com
GET https://beans.com

DOH “VPN”
HTTP/2

SNI: resolver.com

DOH “VPN”
Anonymity set

Client IP

Server IP

1

K

K is the set of domains that can be served on the IP

No dynamic IP requirement
Where are we now?

**ORIGIN** implemented in Firefox

**CERTIFICATE** being standardized by IETF

**DOH** supported by Google DNS, 1.1.1.1

**eSNI** about to be submitted to IETF
ORIGIN

Privacy improvement limited by shared certs

Latency skip both DNS and HTTPS

Security certificate compromise risk
CERTIFICATE

Privacy hide any bean in any burrito

Latency extends origin benefits to any cert

Security exchange DNS for CT or OCSP stapling
DOH

**Privacy** first hop improvement

**Latency** depends on provider, TLS 1.3

**Security** security against attacks, allows passive DNS
eSNI

**Privacy** first domain privacy given dynamic IPs

**Latency** depends on DoH for reliability

**Security** risk of more MiTM
Open Questions

How much privacy does this actually give people?

Does this incentivize further consolidation?

Does increased performance and privacy outweigh the legitimate need for external visibility?
Website Fingerprinting

Removing explicit signals does not protect you from passive ones
Consolidation

Better performance when using a popular provider
Is visibility necessary?

Safety vs. Security
The Evolving Architecture of the Web

Nick Sullivan