The Importance of Being an Earnest stub

Challenges and solution for the versatile stub

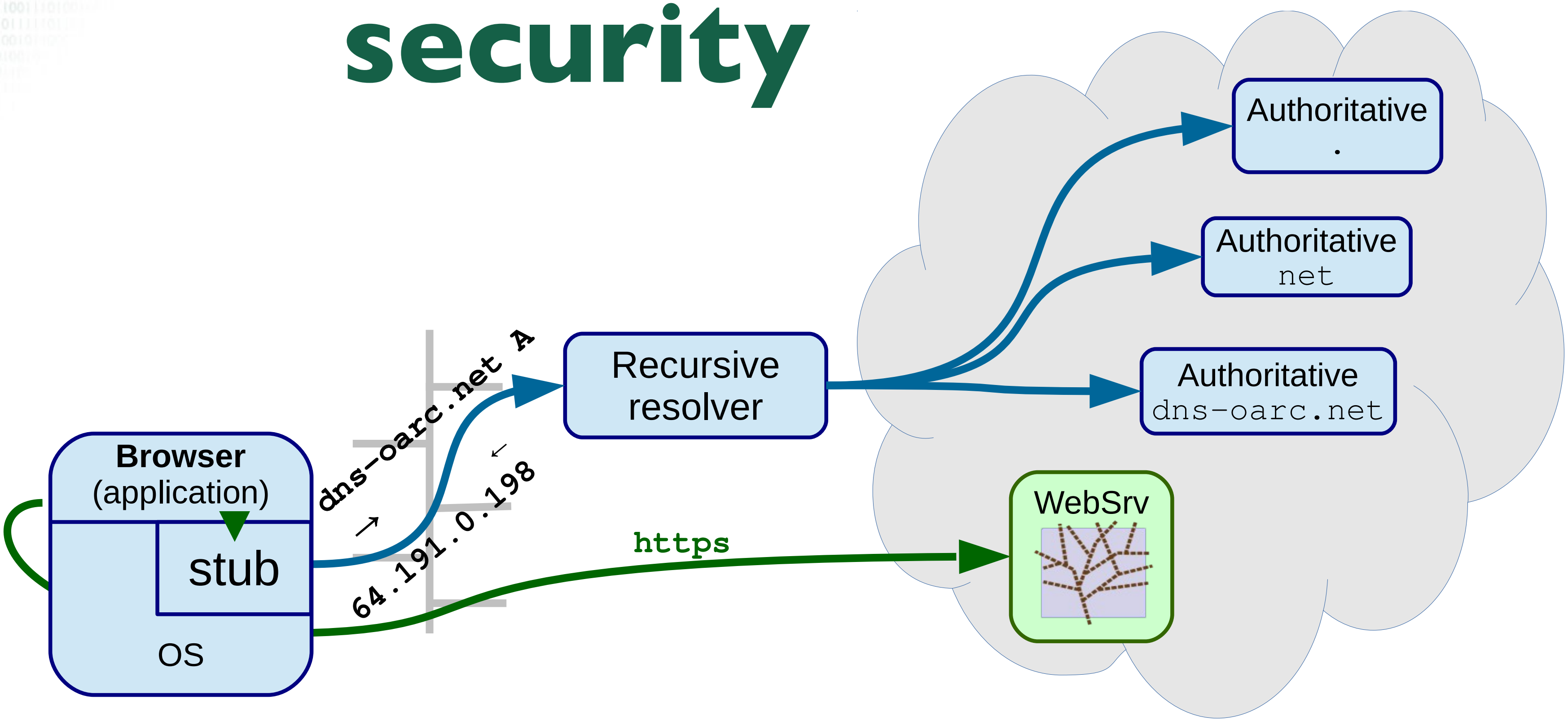
Willem Toorop

13 May 2017

OARC 26 (Madrid)

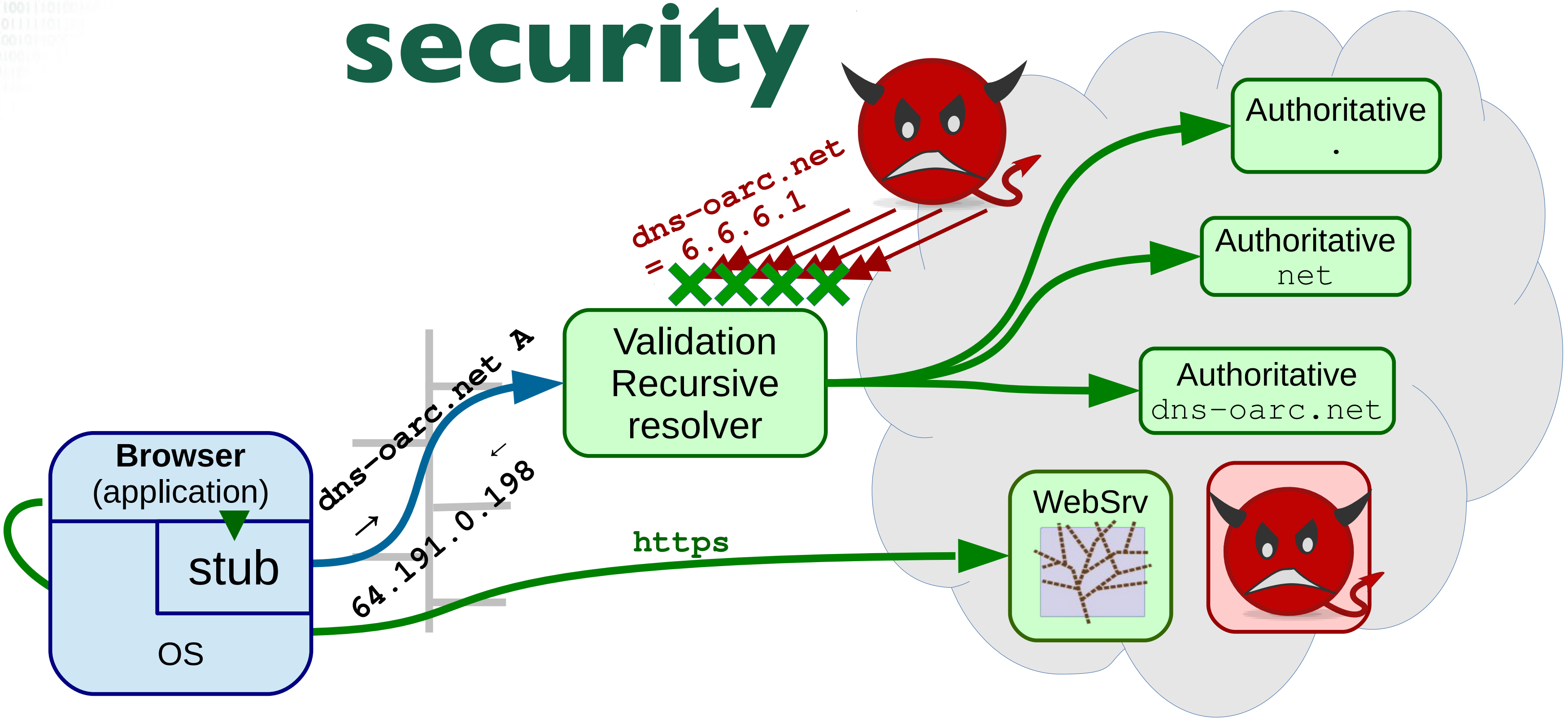
NLnet Labs

From the ground-up security



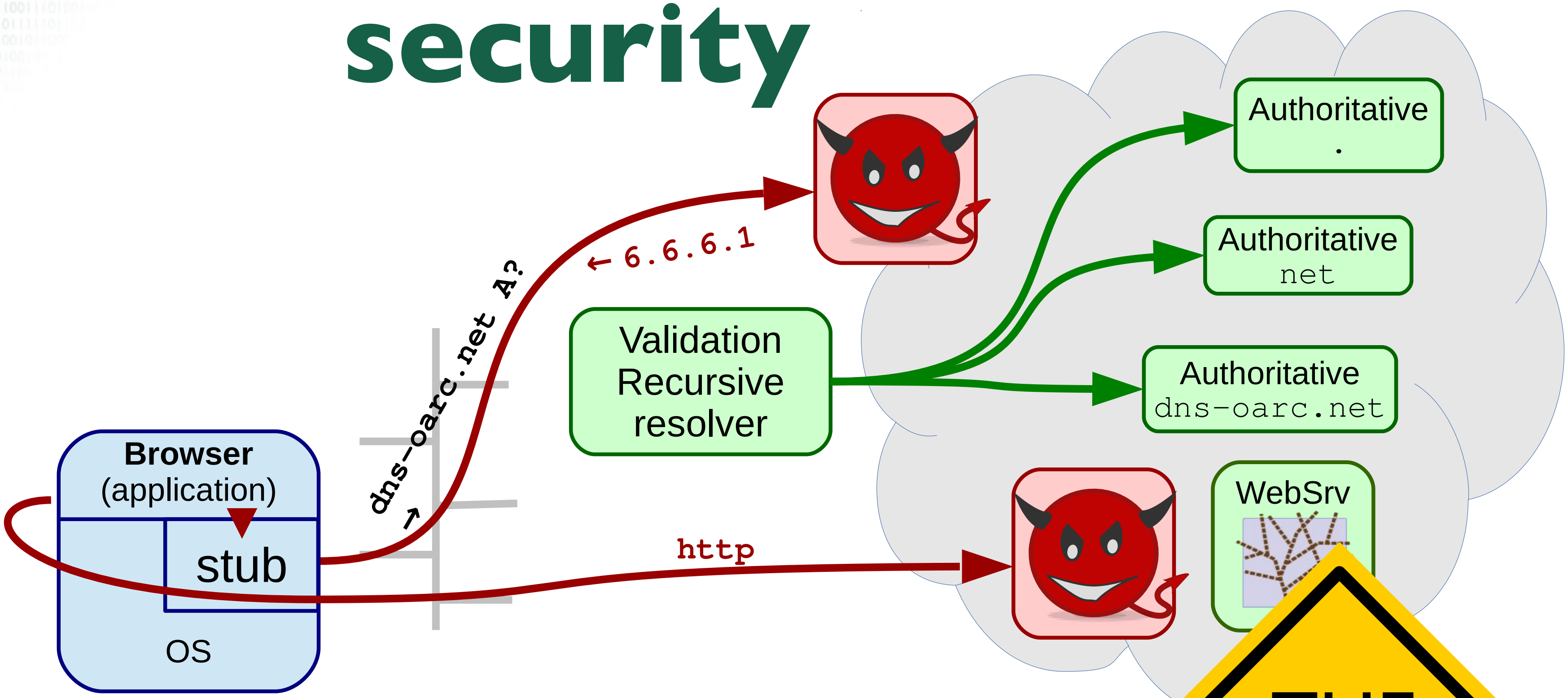
- Every “secure” connection is preceded by a DNS lookup
- The stub does the lookup at the request of the application
  The recursive resolver does all the heavy lifting

From the ground-up security

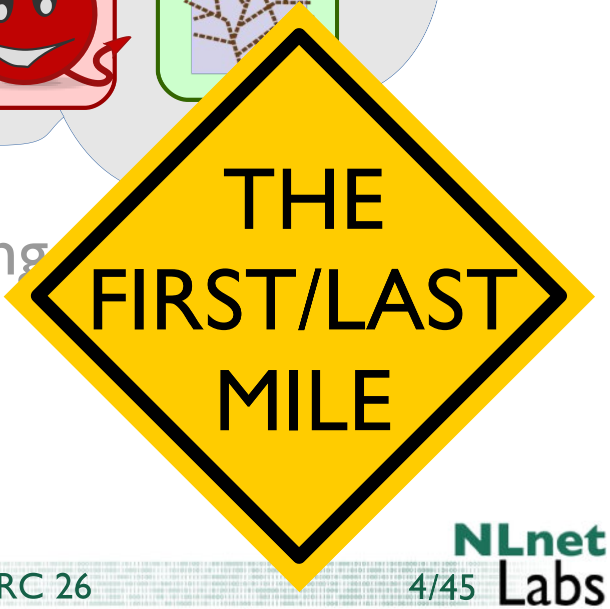


- DNSSEC protects against cache poisoning

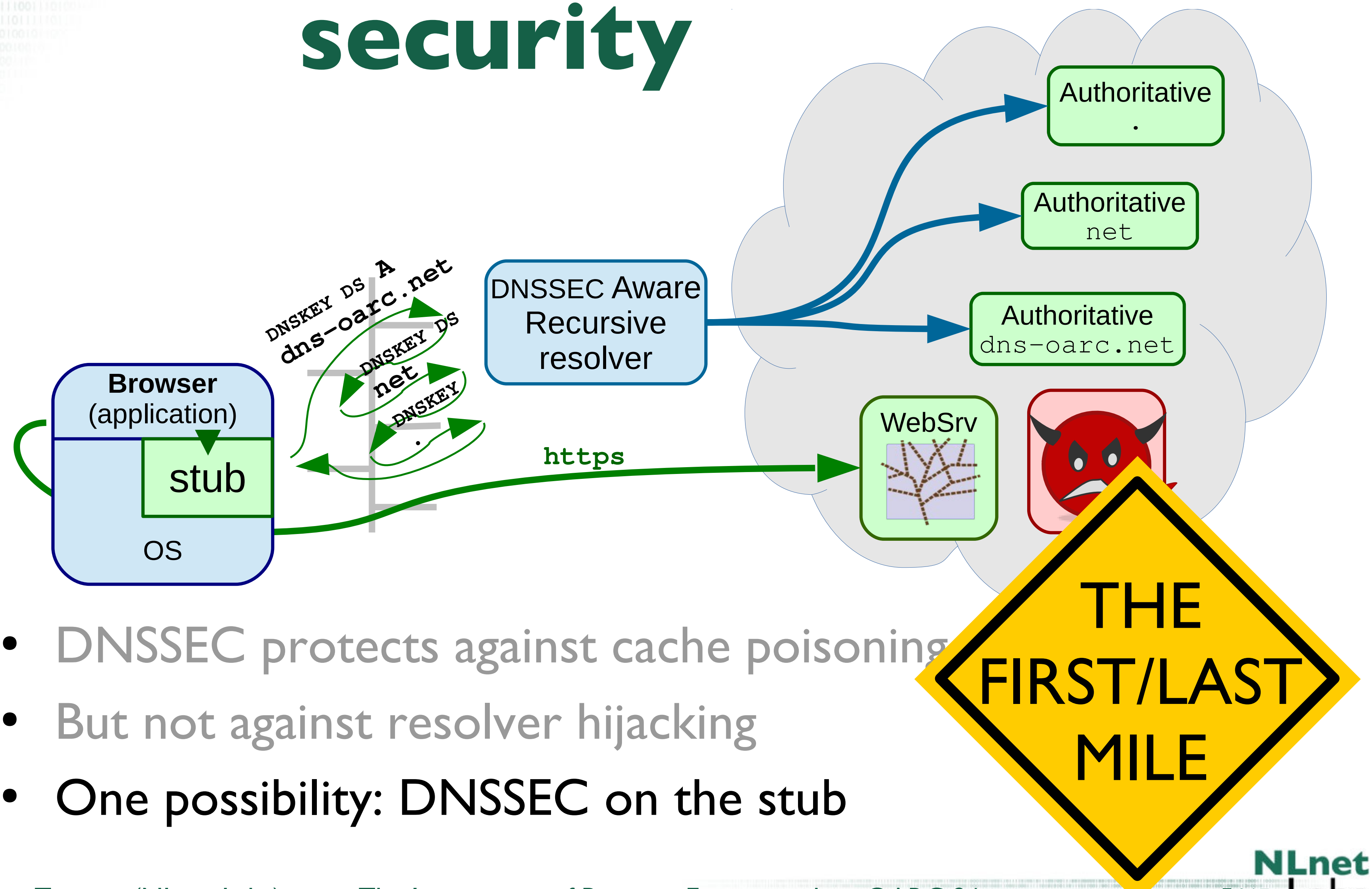
From the ground-up security



- DNSSEC protects against cache poisoning
- But not against resolver hijacking
  (i.e. ARP or DHCP hijacking or routing tricks)

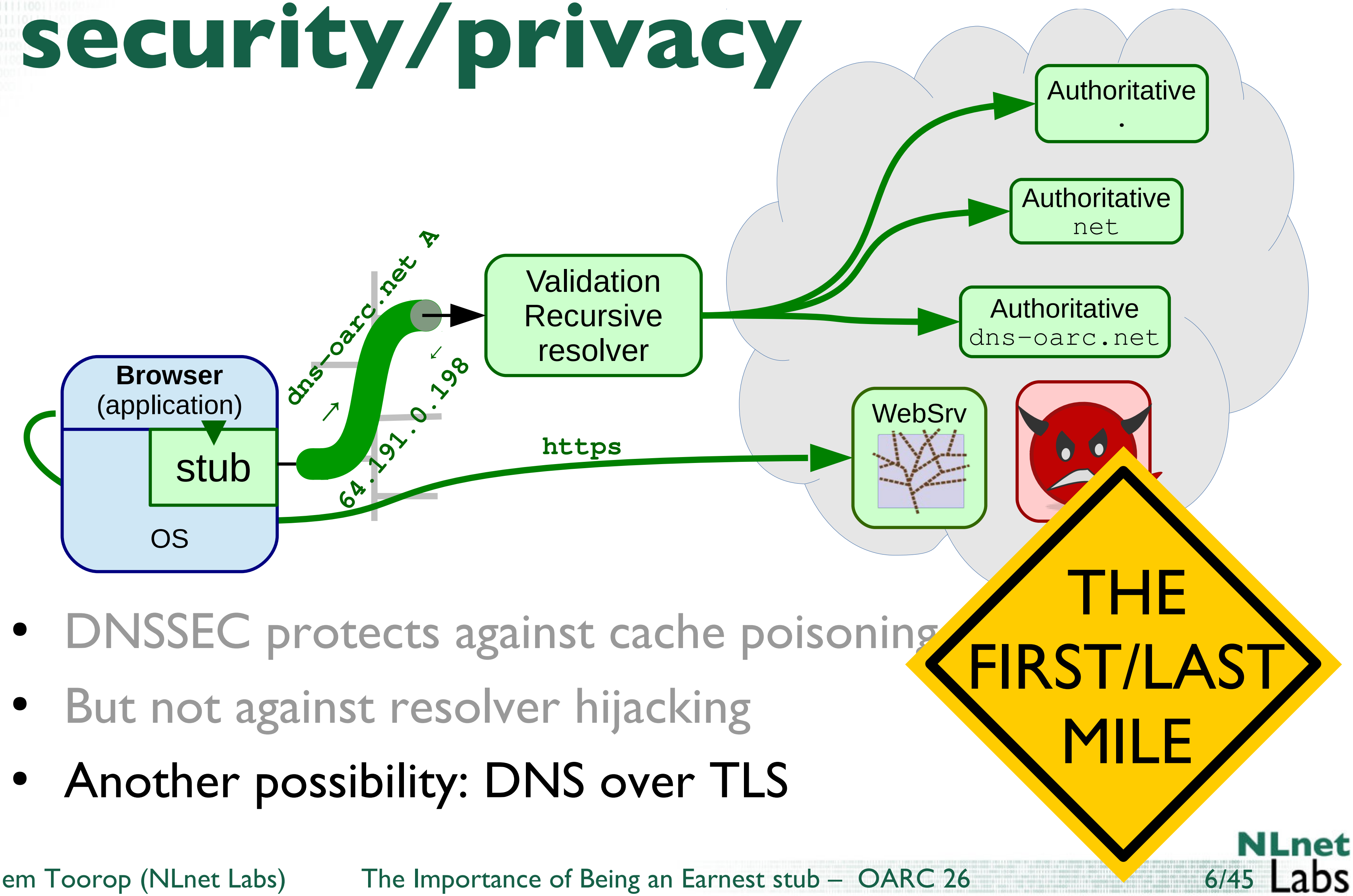


From the ground-up security



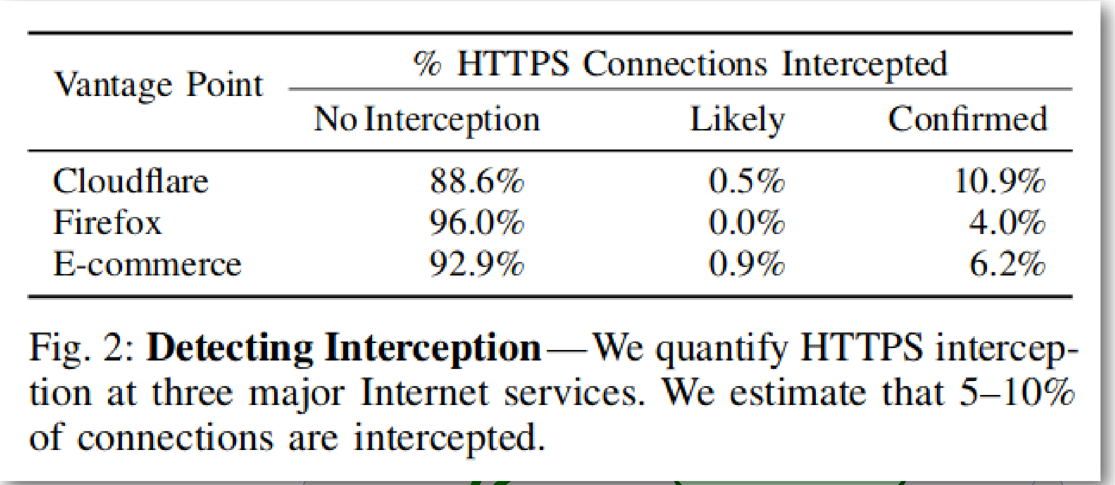
- DNSSEC protects against cache poisoning
- But not against resolver hijacking
- One possibility: DNSSEC on the stub

From the ground-up security/privacy



- DNSSEC protects against cache poisoning
- But not against resolver hijacking
- Another possibility: DNS over TLS

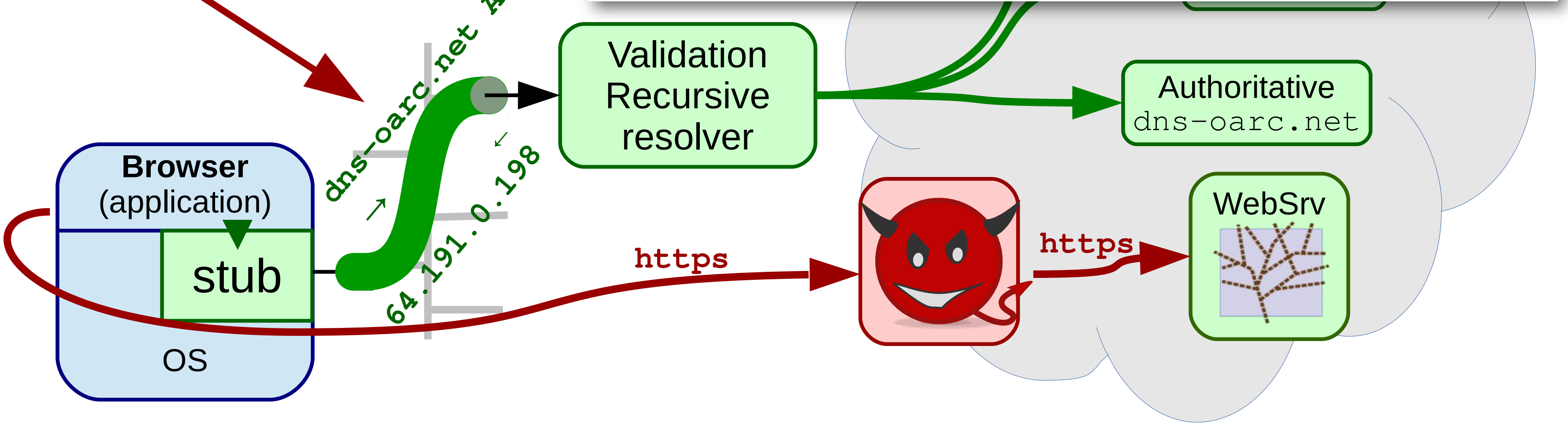
From th
security/p



Vantage Point	% HTTPS Connections Intercepted		
	No Interception	Likely	Confirmed
Cloudflare	88.6%	0.5%	10.9%
Firefox	96.0%	0.0%	4.0%
E-commerce	92.9%	0.9%	6.2%

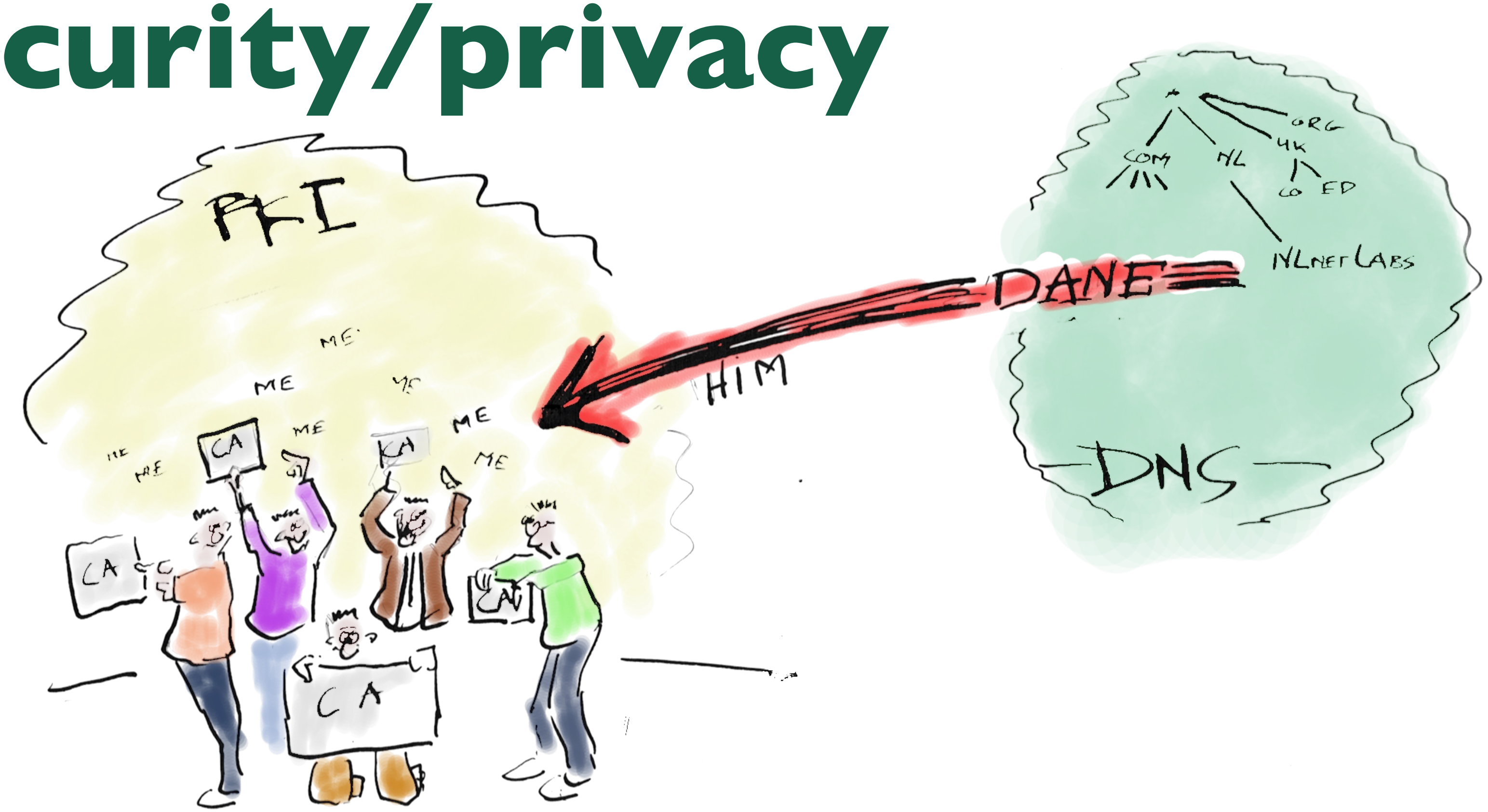
Fig. 2: **Detecting Interception**—We quantify HTTPS interception at three major Internet services. We estimate that 5–10% of connections are intercepted.

Applies to DNS over TLS too



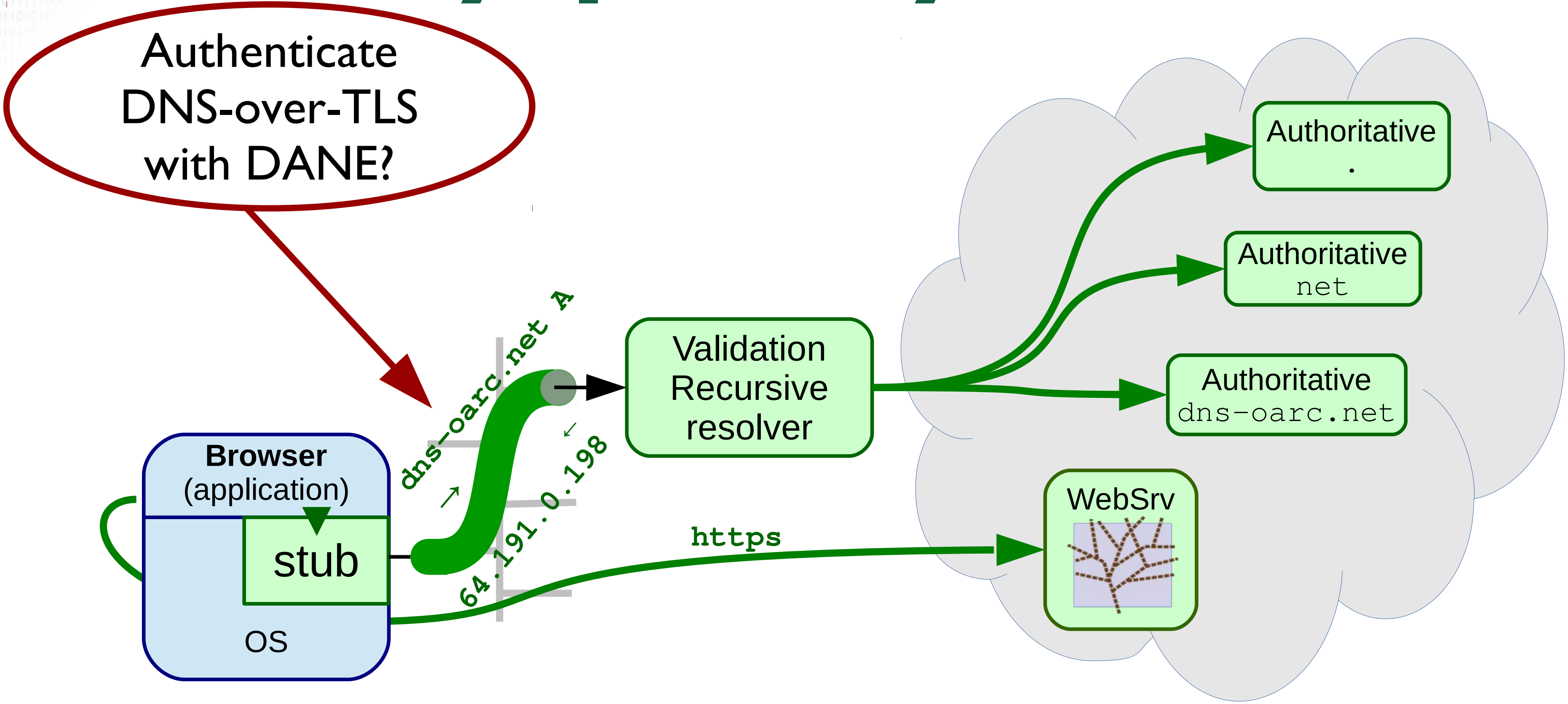
- TLS hijacking? **IS THAT POSSIBLE?!**
- Durumeric, Zakir, et al. "The Security Impact of HTTPS Interception." *Network and Distributed Systems Symposium (NDSS'17)*. 2017. https://www.internetsociety.org/doc/security-impact-https-interception

From the ground-up security/privacy



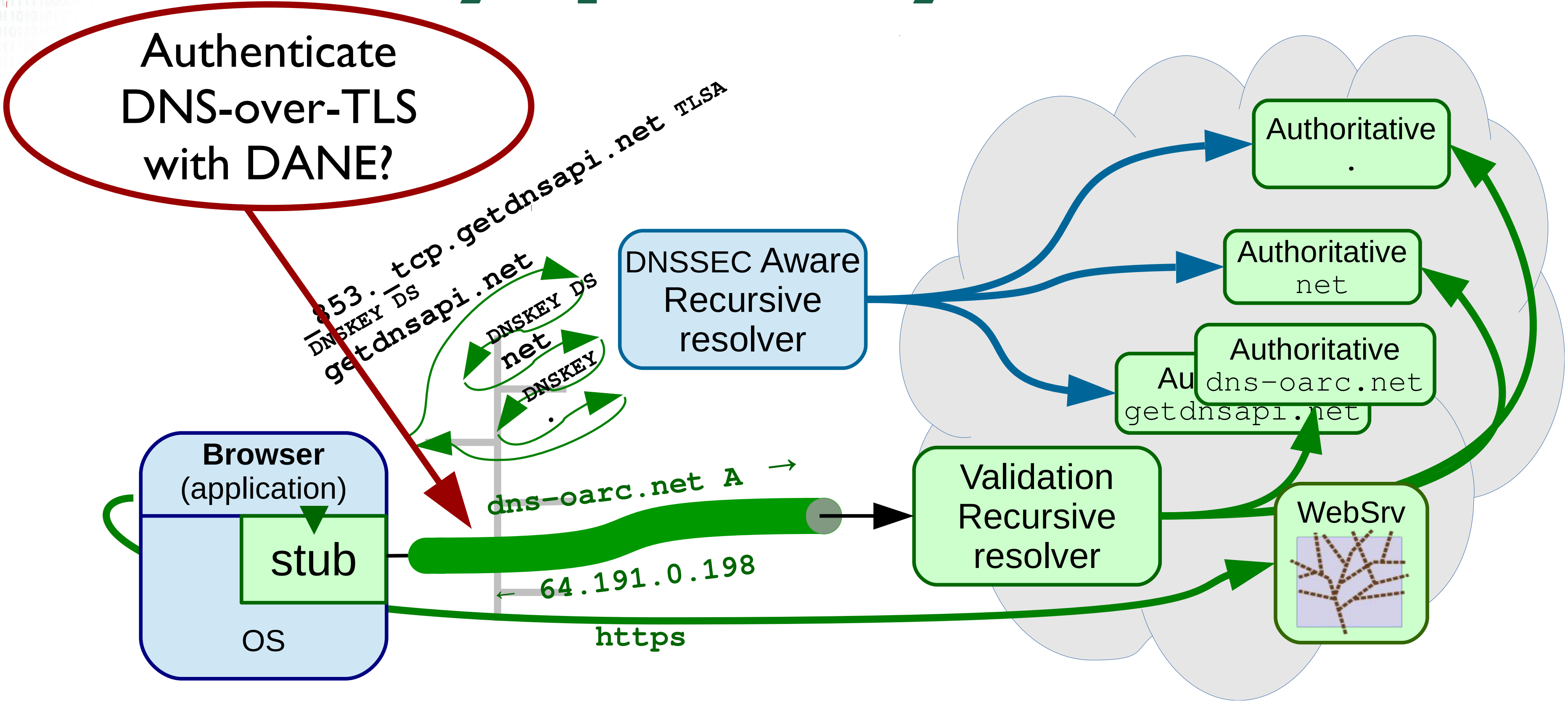
- Strengthen TLS security with the stub: DANE
  (DNS-based Authentication of Named Entities)
- Also signalling system for TLS support
  (For application without user interaction)

From the ground-up security/privacy



- Bootstrap the TLSA lookup with regular DNS?

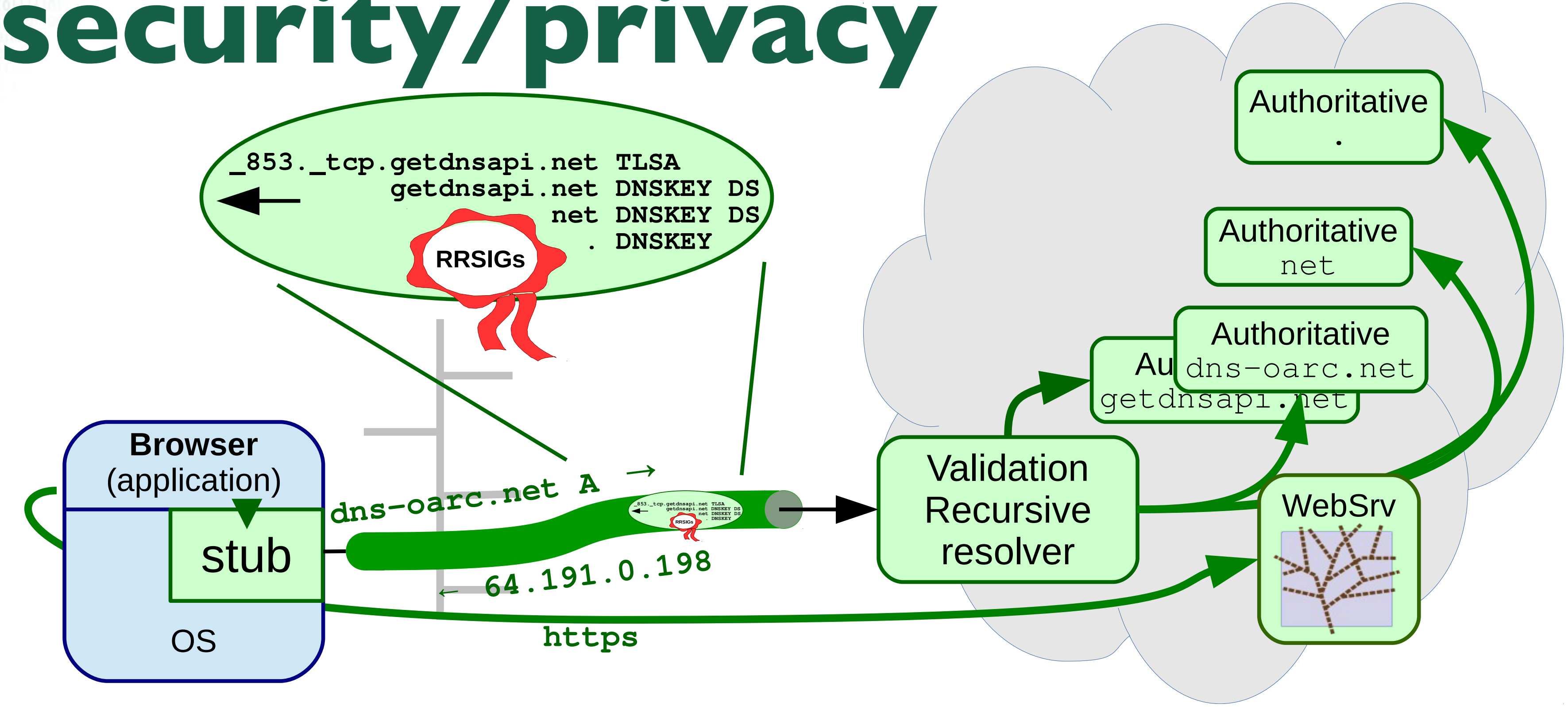
From the ground-up security/privacy



Authenticate DNS-over-TLS with DANE?

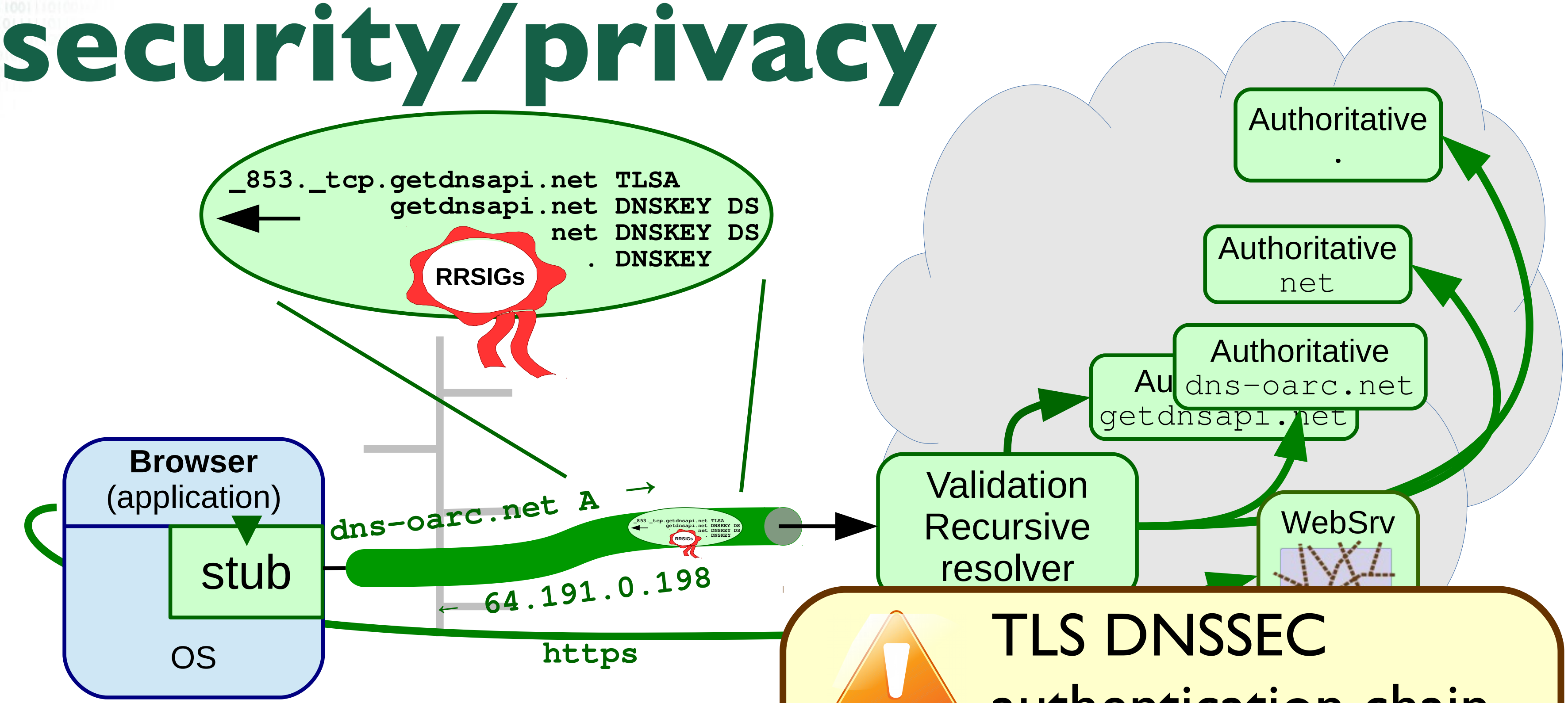
- Bootstrap the TLSA lookup with regular DNS?
  - Chicken and Egg problem

From the ground-up security/privacy



- Bootstrap the TLSA lookup with regular DNS?
- Have the TLSA record + the complete DNSSEC authentication chain embedded in a TLS extension https://tools.ietf.org/html/draft-ietf-tls-dnssec-chain-extension

From the ground-up security/privacy



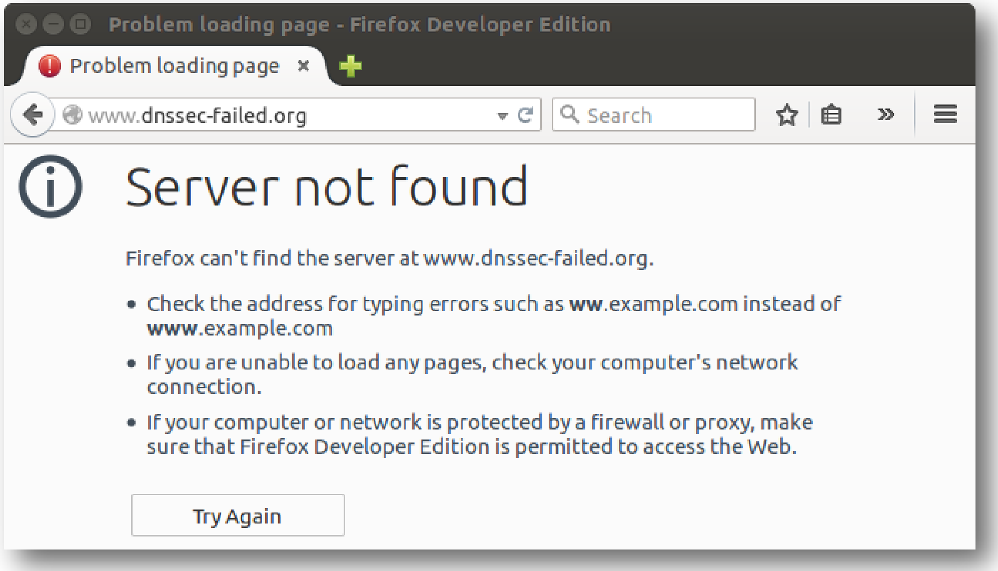
TLS DNSSEC authentication chain extension must be obligatory, to prevent the “Too many CA’s” problem

- Bootstrap the TLSA lookup v
- Have the TLSA record + the authentication chain embedde

https://tools.ietf.org/html/draft-ietf-tls-dnssec-chain-extension

From the ground-up security/privacy

DNSSEC Availability



Server not found

Firefox can't find the server at www.dnssec-failed.org.

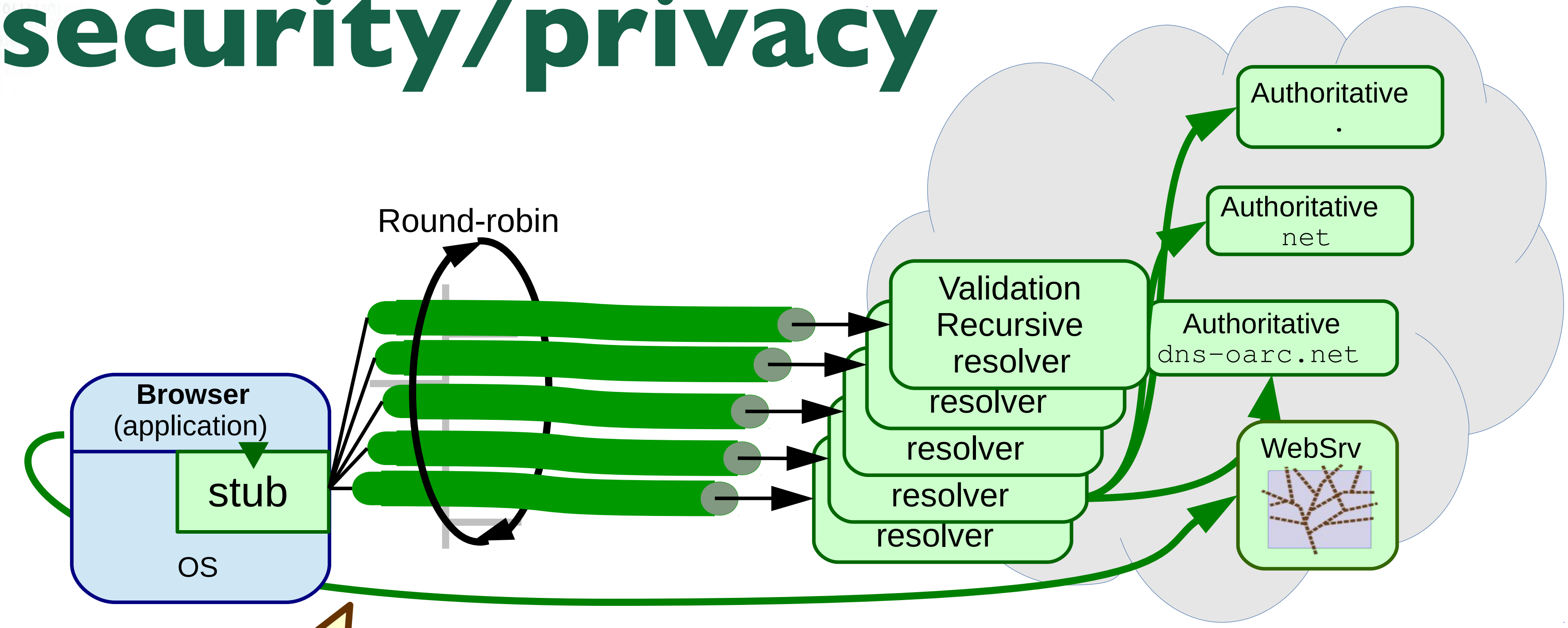
- Check the address for typing errors such as **ww**.example.com instead of **www**.example.com
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox Developer Edition is permitted to access the Web.

DNS Privacy status

- ❌ Clear text DNS
- ⚠ Private DNS
- 🔒 Authenticated Private DNS

- The stub is close to the application
  Inform status of DNSSEC and DNS Privacy

From the ground-up security/privacy



- Enhanced privacy by round-robining upstreams

From the ground-up security/privacy

- **Requirements for the versatile stub**

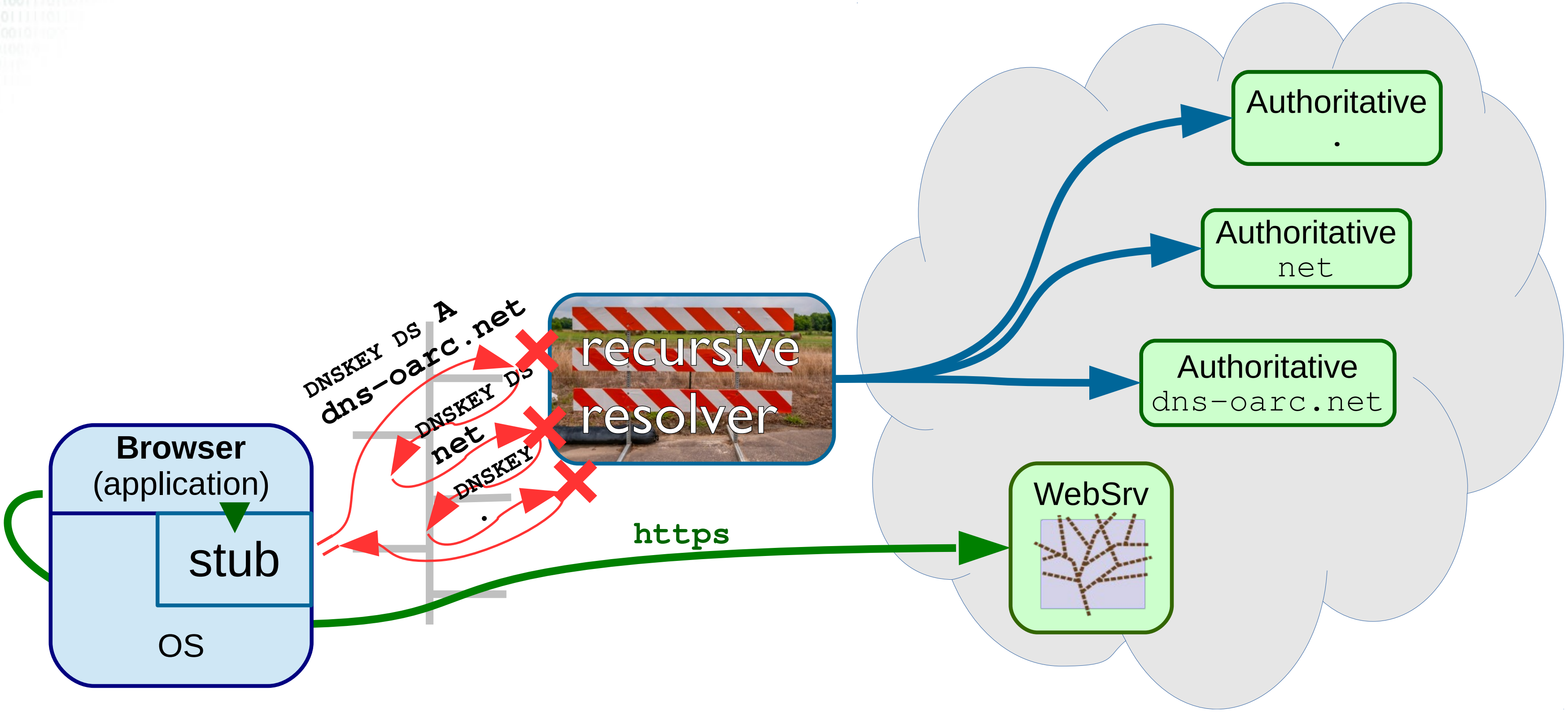
	DNSSEC	DNS over TLS	Non address lookups	API
Cross the first DNSSEC mile	X			
From the ground up Privacy		X		
Strengthened TLS authentication *(DANE)*	X		X	
Strengthened opportunistic TLS *(DANE)*	X		X	
Provide status of DNSSEC & DNS over TLS				X

From the ground-up security/privacy

- **Requirements for the versatile stub**

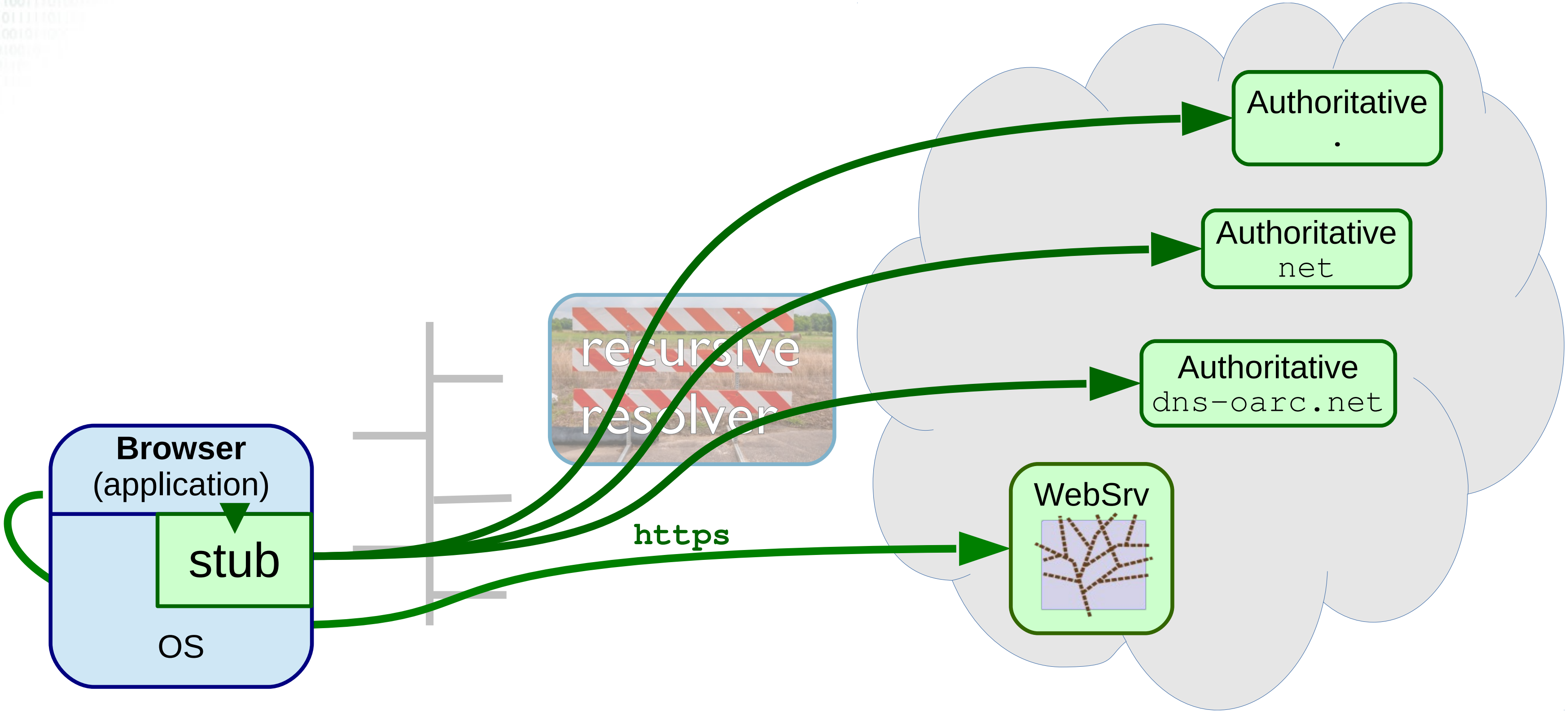
	DNSSEC	DNS over TLS	Non address lookups	API
Cross the first DNSSEC mile	X			
From the ground up Privacy		X		
Strengthened TLS authentication *(DANE)*	X		X	
Strengthened opportunistic TLS *(DANE)*	X		X	
Provide status of DNSSEC & DNS over TLS				X

DNSSEC Roadblocks



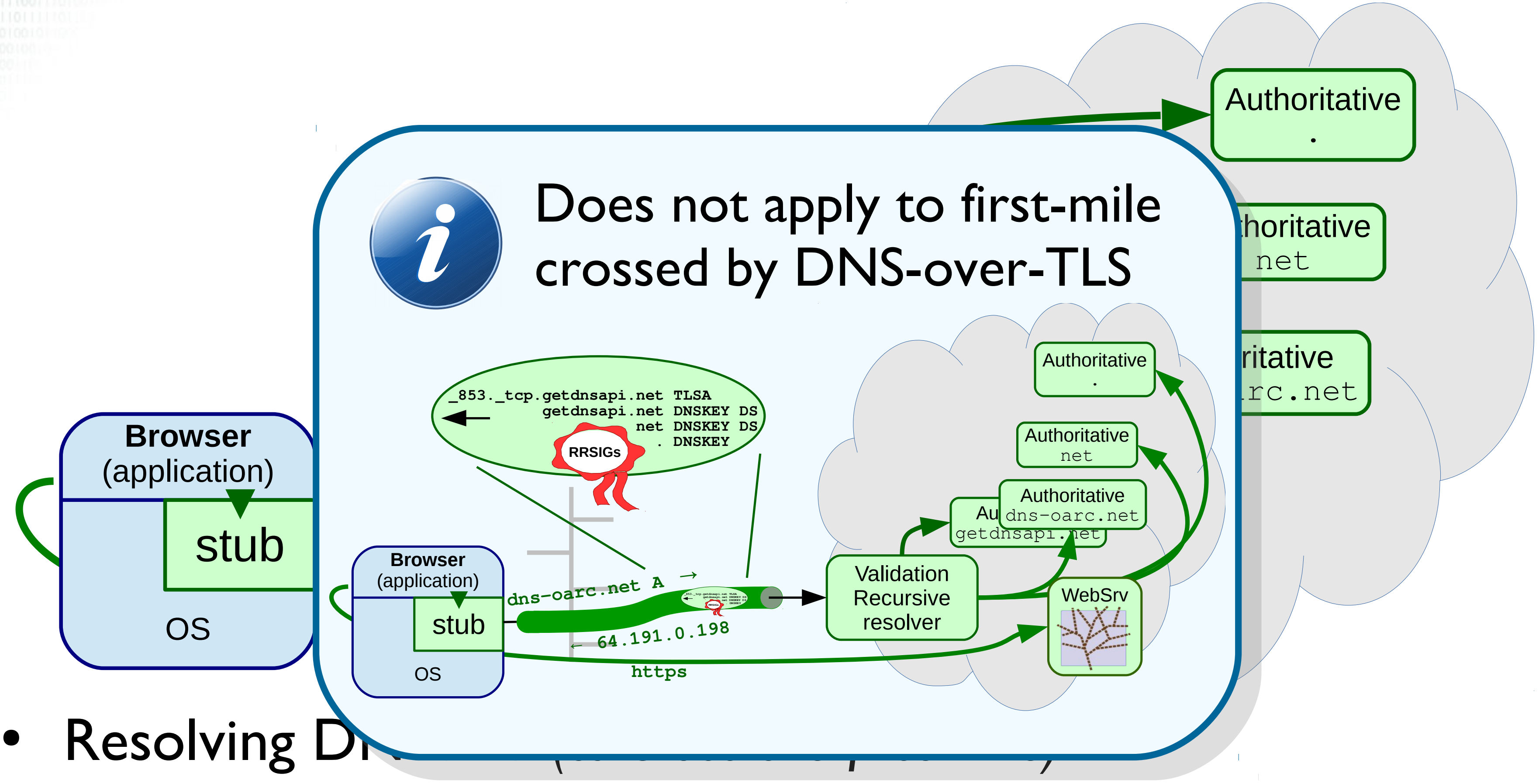
- Resolving DNSSEC *(to cross the first mile)* needs DNSSEC Aware recursive resolver

DNSSEC Roadblocks



- Resolving DNSSEC *(to cross the first mile)* needs DNSSEC Aware recursive resolver
- DNSSEC Roadblock Avoidance https://tools.ietf.org/html/rfc8027
  +Full recursion capability

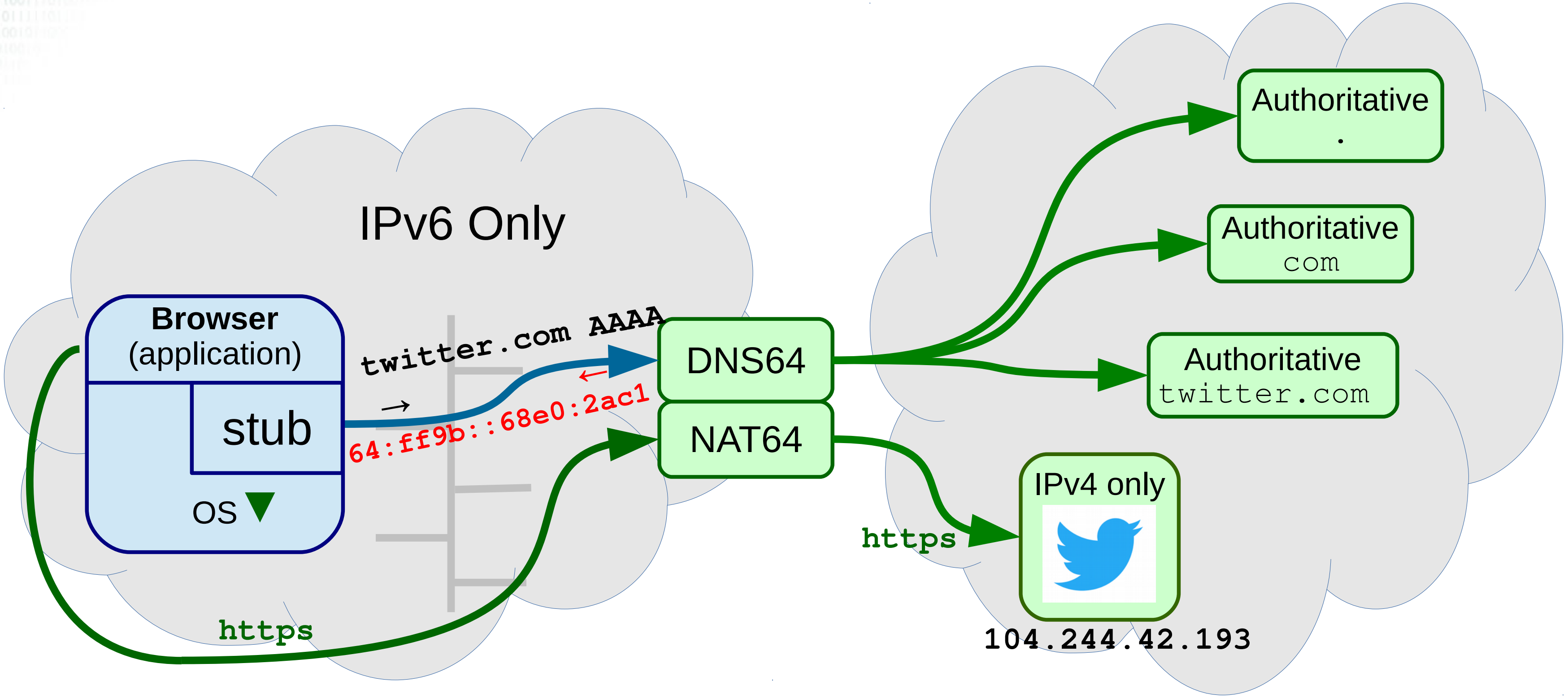
DNSSEC Roadblocks



Does not apply to first-mile crossed by DNS-over-TLS

- Resolving D
  needs DNSSEC Aware recursive resolver
- DNSSEC Roadblock Avoidance https://tools.ietf.org/html/rfc8027
  +Full recursion capability

DNSSEC Roadblocks

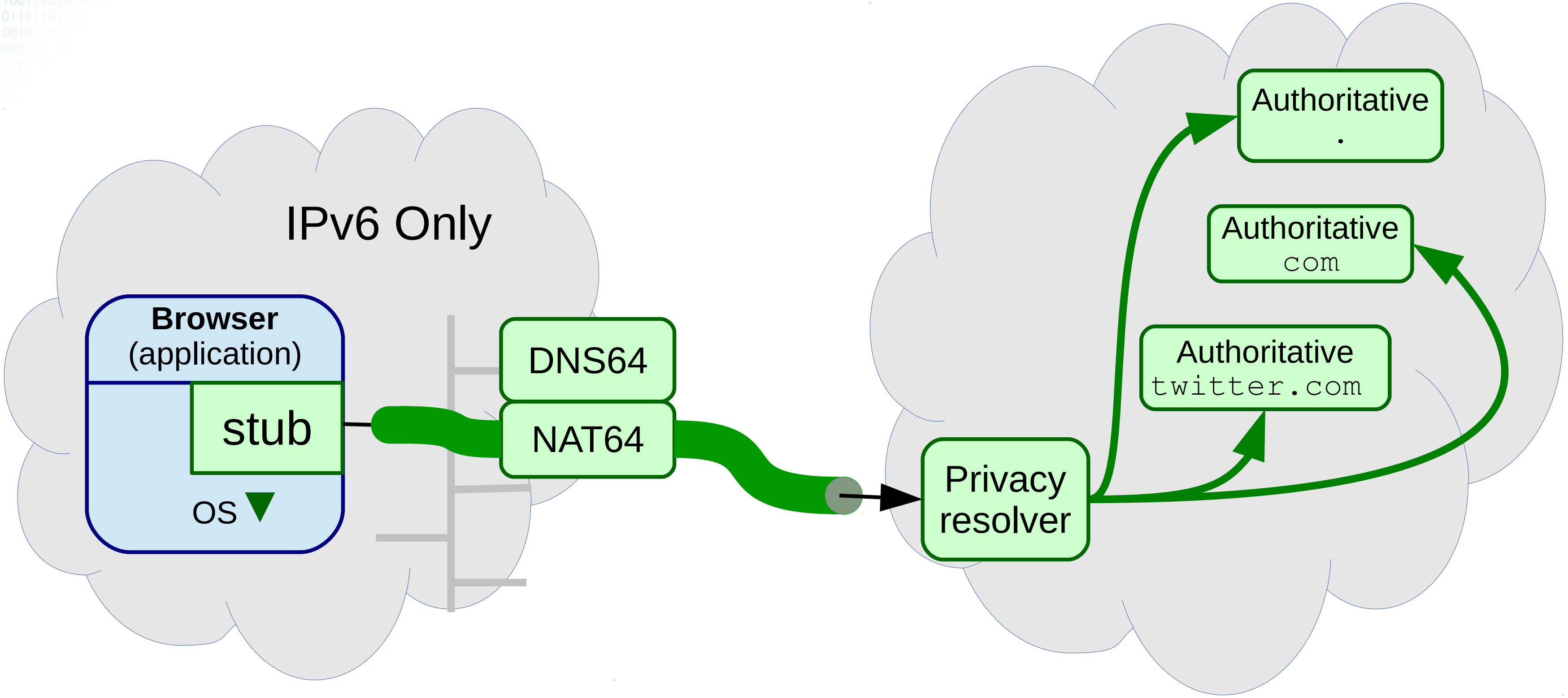


- DNSSEC Roadblock Avoidance https://tools.ietf.org/html/rfc8027
- IPv6 Address Synthesis Prefix Discovery

  https://tools.ietf.org/html/rfc7050

  +DNS64 capability https://tools.ietf.org/html/rfc6147

DNSSEC Roadblocks



- DNSSEC Roadblock Avoidance https://tools.ietf.org/html/rfc8027
- IPv6 Address Synthesis Prefix Discovery

https://tools.ietf.org/html/rfc7050

+DNS64 capability https://tools.ietf.org/html/rfc6147

DNSSEC Roadblocks

Root
KSK
Rollover

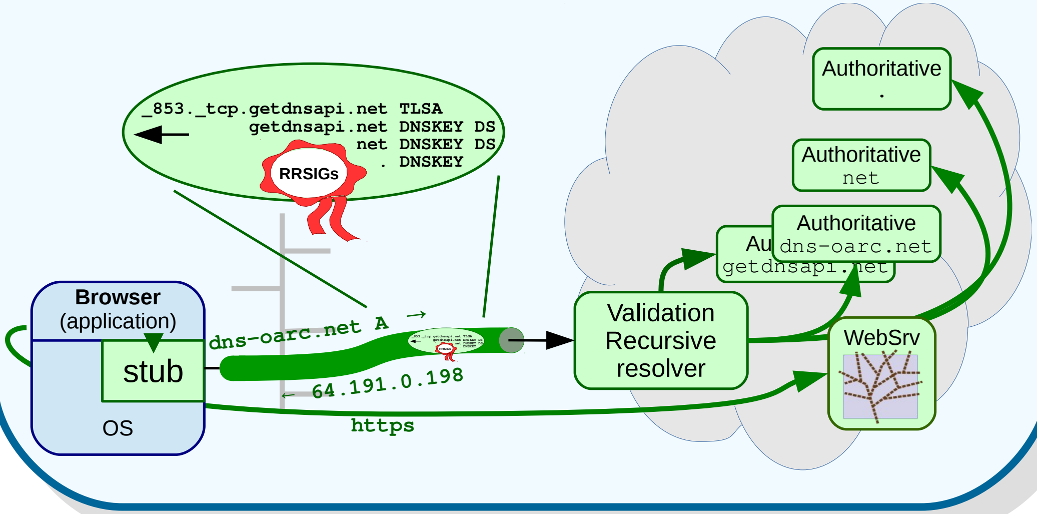
- DNSSEC validating stubs must do RFC5011

DNSSEC Roadblocks

Root

r

In-band RFC5011 tracking with DNSSEC auth chain TLS extension



- DNSSEC v

DNSSEC Roadblocks

Root KSK Rollover

- DNSSEC validating stubs must do RFC5011
- A stub library for DANE has no system config
  +bootstrap DNSSEC capability: https://tools.ietf.org/html/rfc7958
- A stub library for DANE runs with user's privileges

DNSSEC Roadblocks

DNSSEC stubs capability requirements

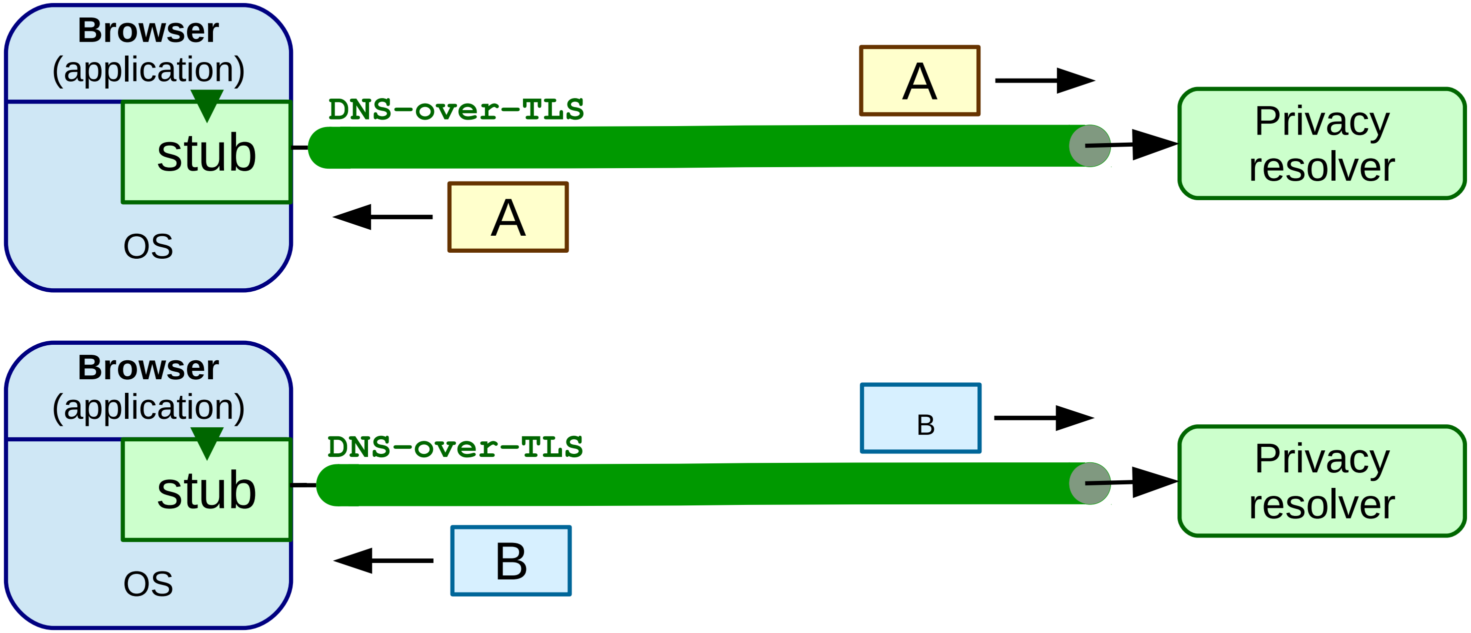
DNSSEC validation	*(various)*
DNSSEC Roadblock Avoidance	*RFC8027*
IPv6 Prefix Discovery	RFC7050
IPv6 Address Synthesis	RFC6147
Automated Trust Anchor Updates	RFC5011
Automated Initial Trust Anchor retrieval	RFC7958

From the ground-up security/privacy

- **Requirements for the versatile stub**

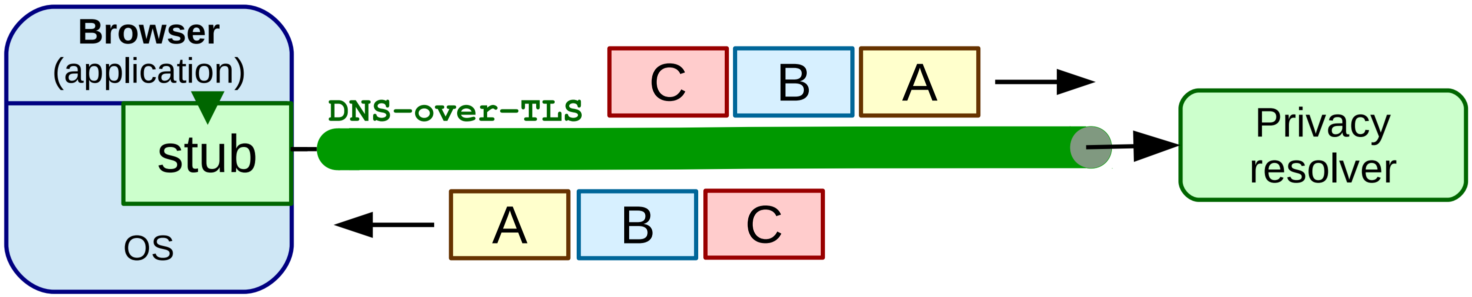
	DNSSEC	DNS over TLS	Non address lookups	API
Cross the first DNSSEC mile	X			
From the ground up Privacy		X		
Strengthened TLS authentication *(DANE)*	X		X	
Strengthened opportunistic TLS *(DANE)*	X		X	
Provide status of DNSSEC & DNS over TLS				X

Requirements for DNS-over-TLS



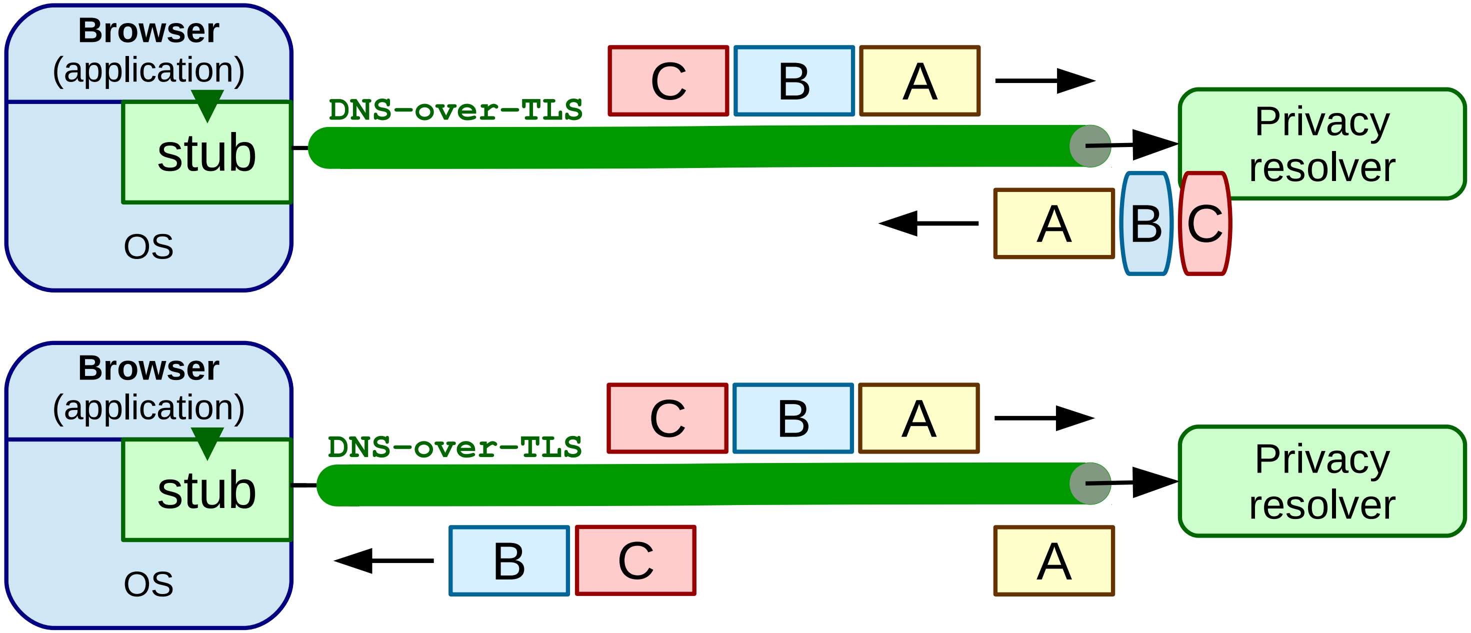
- TCP fastopen *(optional)* https://tools.ietf.org/html/rfc7413
- Connection reuse https://tools.ietf.org/html/rfc7766
- EDNS0 keepalive https://tools.ietf.org/html/rfc7828
- EDNS0 padding https://tools.ietf.org/html/rfc7830

Requirements for DNS-over-TLS



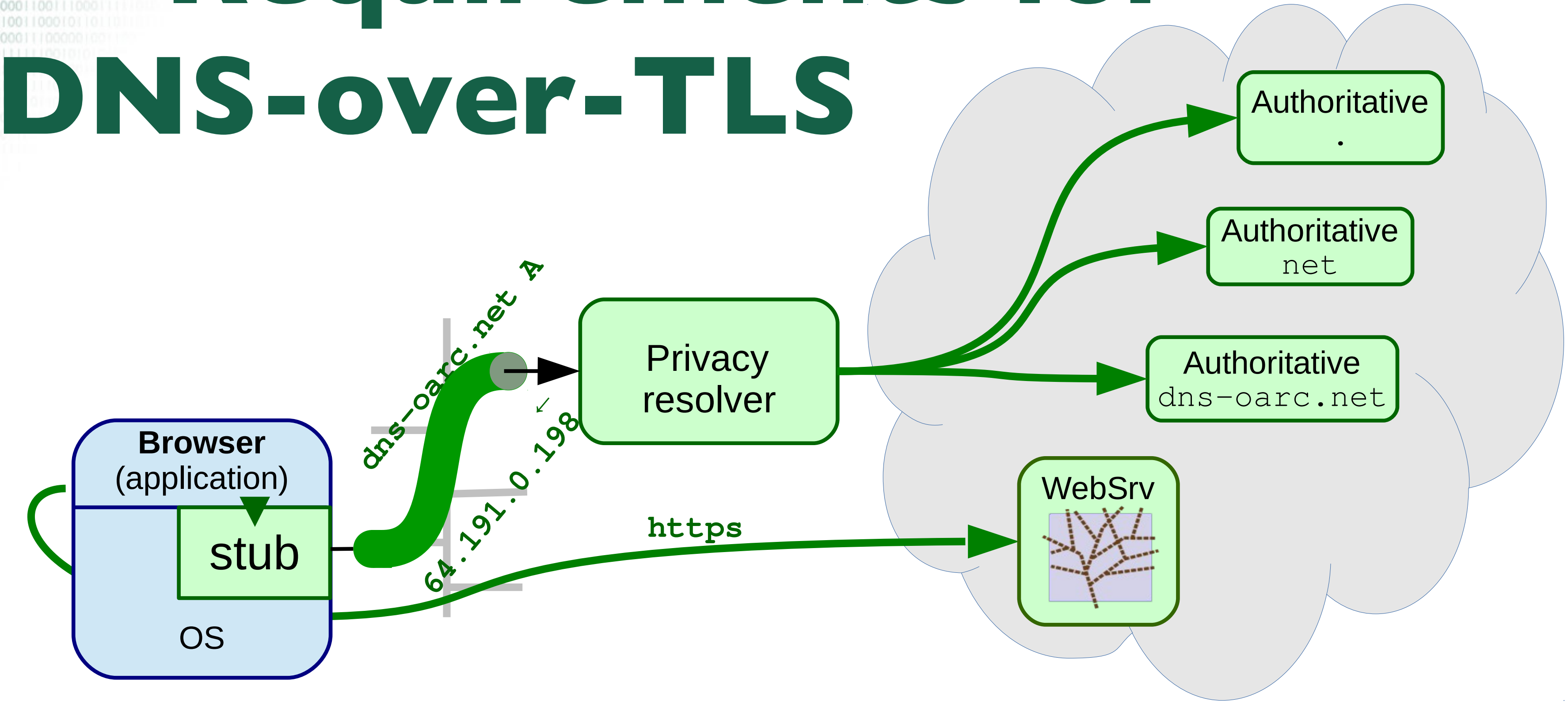
- Connection reuse (Q/R, Q/R, Q/R)
- Pipe-lining of queries (Q,Q,Q,R,R,R)

Requirements for DNS-over-TLS



- Connection reuse (Q/R, Q/R, Q/R)
- Pipe-lining of queries (Q,Q,Q,R,R,R)
- Process Out-Of-Order-Responses (Q_1,Q_2, R_2, R_1)

Requirements for DNS-over-TLS

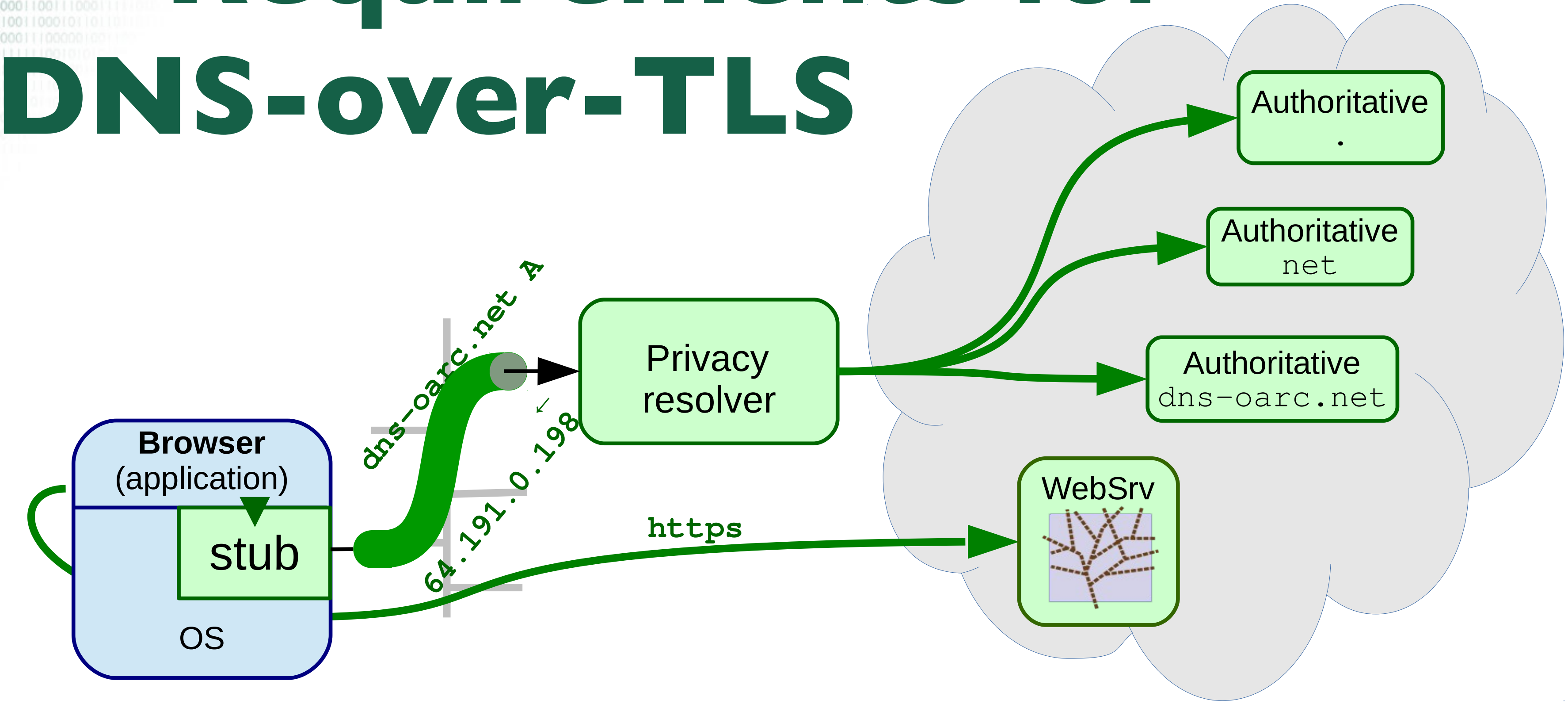


- Strict or Opportunistic usage profiles?

https://tools.ietf.org/html/draft-ietf-dprive-dtls-and-tls-profiles-09

1) Authenticated Private DNS
2) Private DNS
3) Clear text DNS

Requirements for DNS-over-TLS

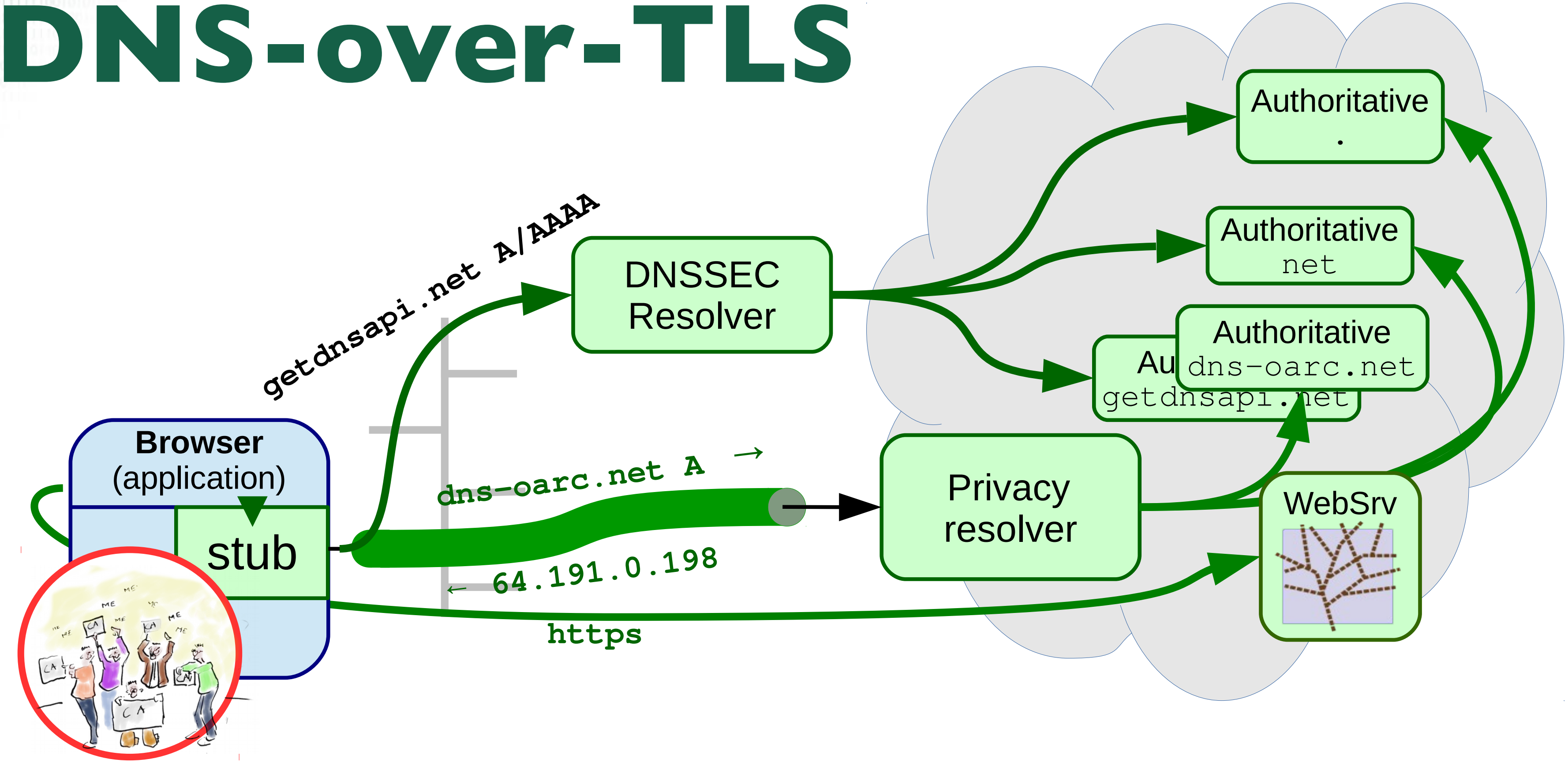


- Strict or Opportunistic usage profiles?

RFC7858 (DNS-over-TLS)
defined direct SPKI authentication only

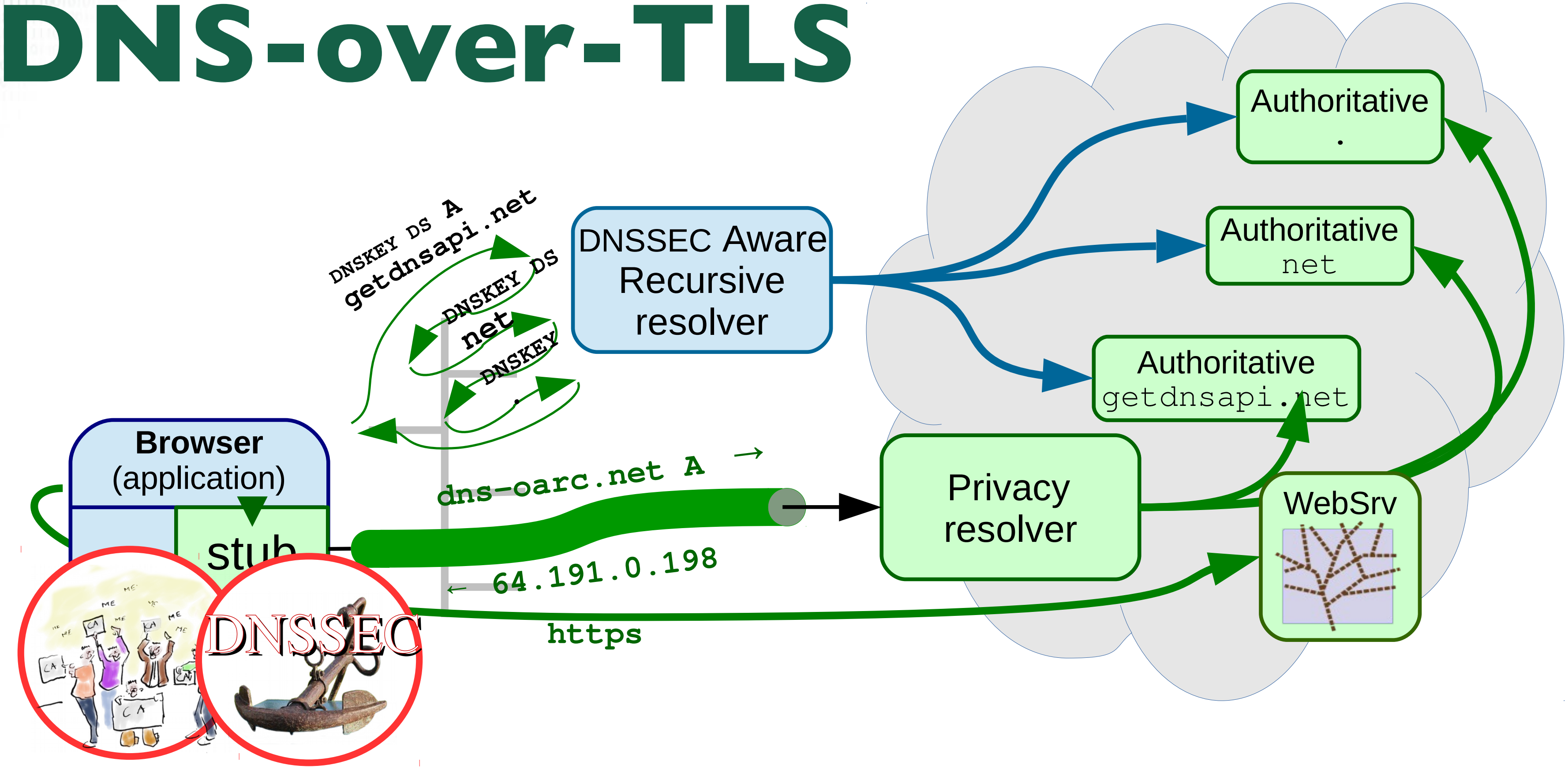
3) Clear text DNS

Requirements for DNS-over-TLS



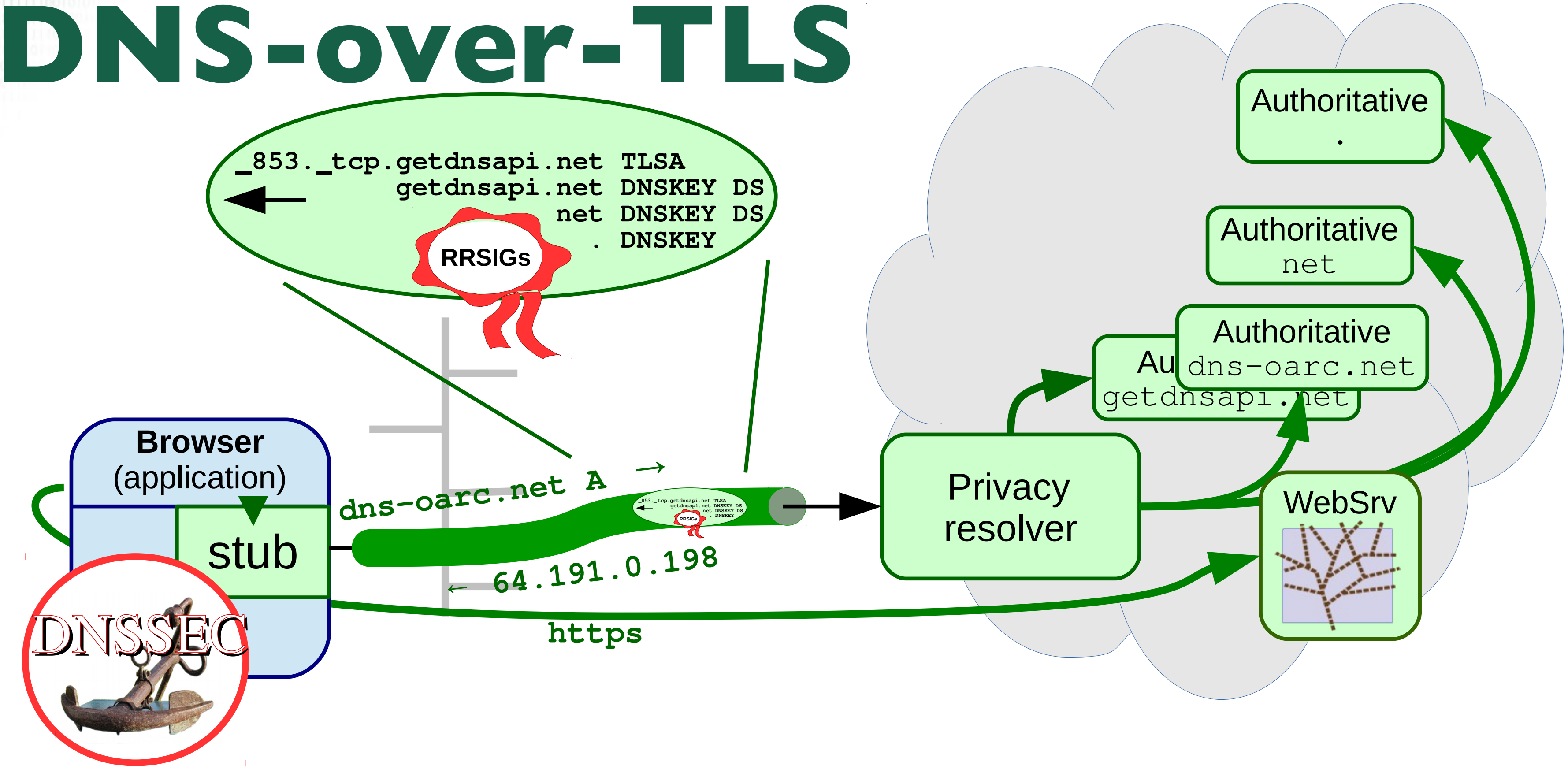
- Regular PKIX authentication
  (bootstrap address lookup with regular DNS(SEC))

Requirements for DNS-over-TLS



- Regular PKIX authentication
- Authenticate with DANE
  (stricter opportunistic with TLSA signalling)

Requirements for DNS-over-TLS



- Regular PKIX authentication
- Authenticate with DANE
- DNSSEC authentication chain TLS extension

Requirements for DNS Privacy

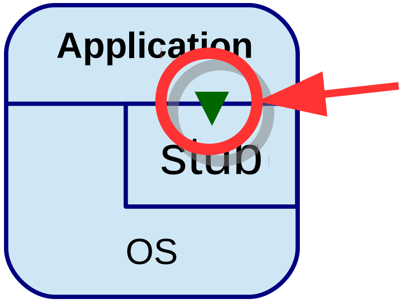
DNS-over-TLS	RFC7858
Reuse / Pipelining / OOOR	RFC7766
TCP Fastopen	RFC7413
ENDS0 keepalive	RFC7828
ENDS0 padding	RFC7830
PKIX support for authentication	*(various)*
DNSSEC support *(for address lookup and authentication)*	*(various)*

From the ground-up security/privacy

- **Requirements for the versatile stub**

	DNSSEC	DNS over TLS	Non address lookups	API
Cross the first DNSSEC mile	X			
From the ground up Privacy		X		
Strengthened TLS authentication *(DANE)*	X		X	
Strengthened opportunistic TLS *(DANE)*	X		X	
Provide status of DNSSEC & DNS over TLS				X

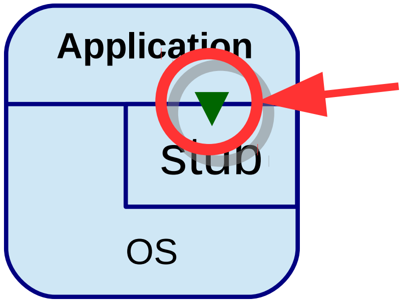
Non address lookups - Application Interface



`getaddrinfo()` **and** `getnameinfo()`

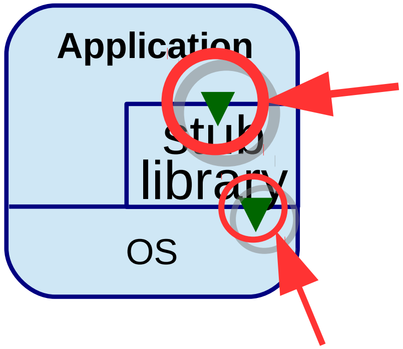
(POSIX standard extended by RFC3493 for IPv6)

Non address lookups - Application Interface



`getaddrinfo()` **and** `getnameinfo()`

(POSIX standard extended by RFC3493 for IPv6)



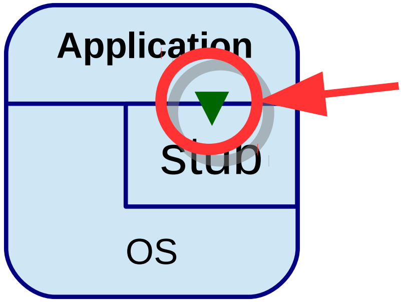
Talk to upstreams directly with a library:

- ~~`libresolv`~~, `libval`, `ldns`, `libunbound`, `libgetdns`

Learn upstreams from OS

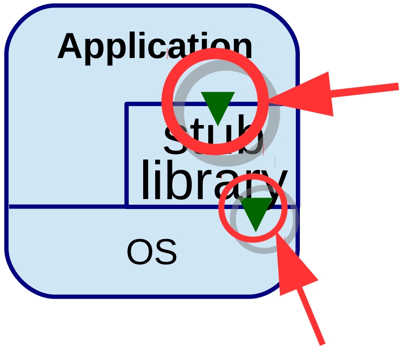
- `/etc/resolv.conf`, `NetworkManager`, **registry...**

Non address lookups - Application Interface



Applications using `getaddrinfo()` API will not get the versatile stub features

(first DNSSEC mile coverage, DNS privacy)



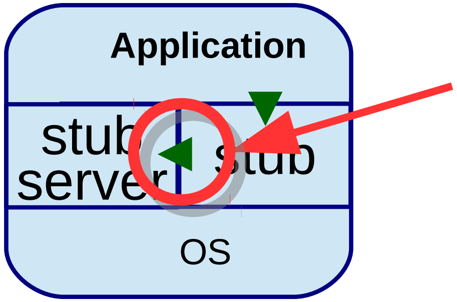
Talk to upstreams directly with a library:

- ~~`libresolv`~~, `libval`, `ldns`, `libunbound`, `libgetdns`

Learn upstreams from OS

- `/etc/resolv.conf`, `NetworkManager`, registry...

Non address lookups - Application Interface



Stub server listening on `127.0.0.1:53`

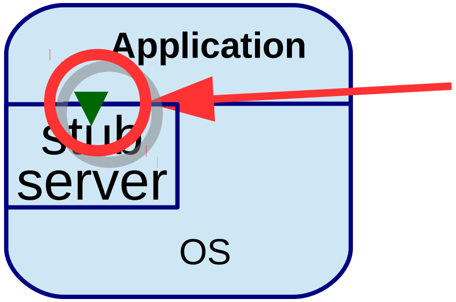
- `getaddrinfo()` and `getnameinfo()` use system stub which uses stub server

Stubby

Dnssec-Trigger

Dnsmasq

Non address lookups - Application Interface



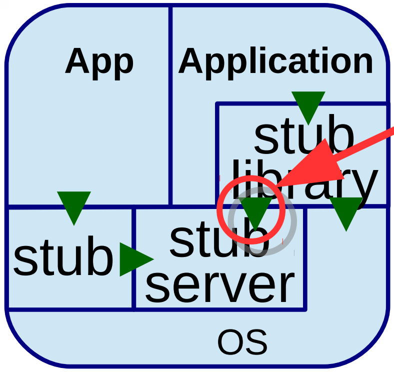
`getaddrinfo()` and `getnameinfo()` use systemd-resolved via nsswitch module

- Stub server listening on `127.0.0.53:53`

`systemd-resolved.service`

systemd-resolved

Non address lookups - Application Interface



Talk to stub server via a library:

- ~~`libresolv`~~, `libval`, `ldns`, `libunbound`, `libgetdns`

`systemd-resolved.service`

systemd-resolved

`127.0.0.53:53`

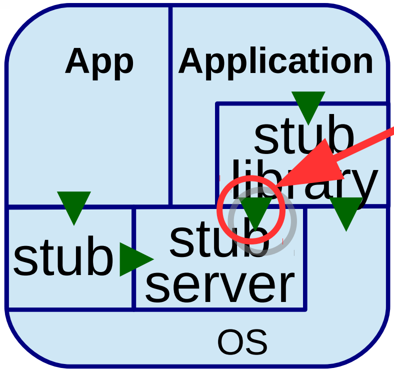
getdns

Stubby

Dnssec-Trigger

Dnsmasq

Non address lookups - Application Interface



Talk to stub server via a library:

- ~~`libresolv`~~, `libval`, `ldns`, `libunbound`, `libgetdns`

`systemd-resolved.service`

systemd-resolved ✗

`127.0.0.53:53`

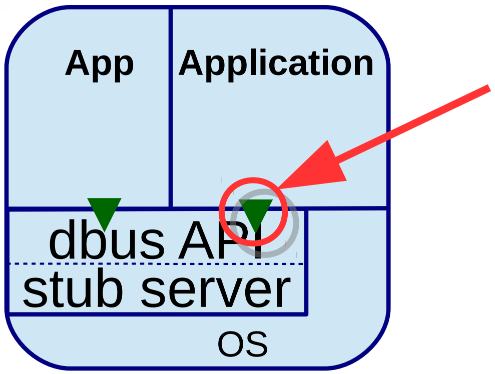
getdns

Stubby ✓

Dnssec-Trigger ✓

Dnsmasq ✓

Non address lookups - Application Interface



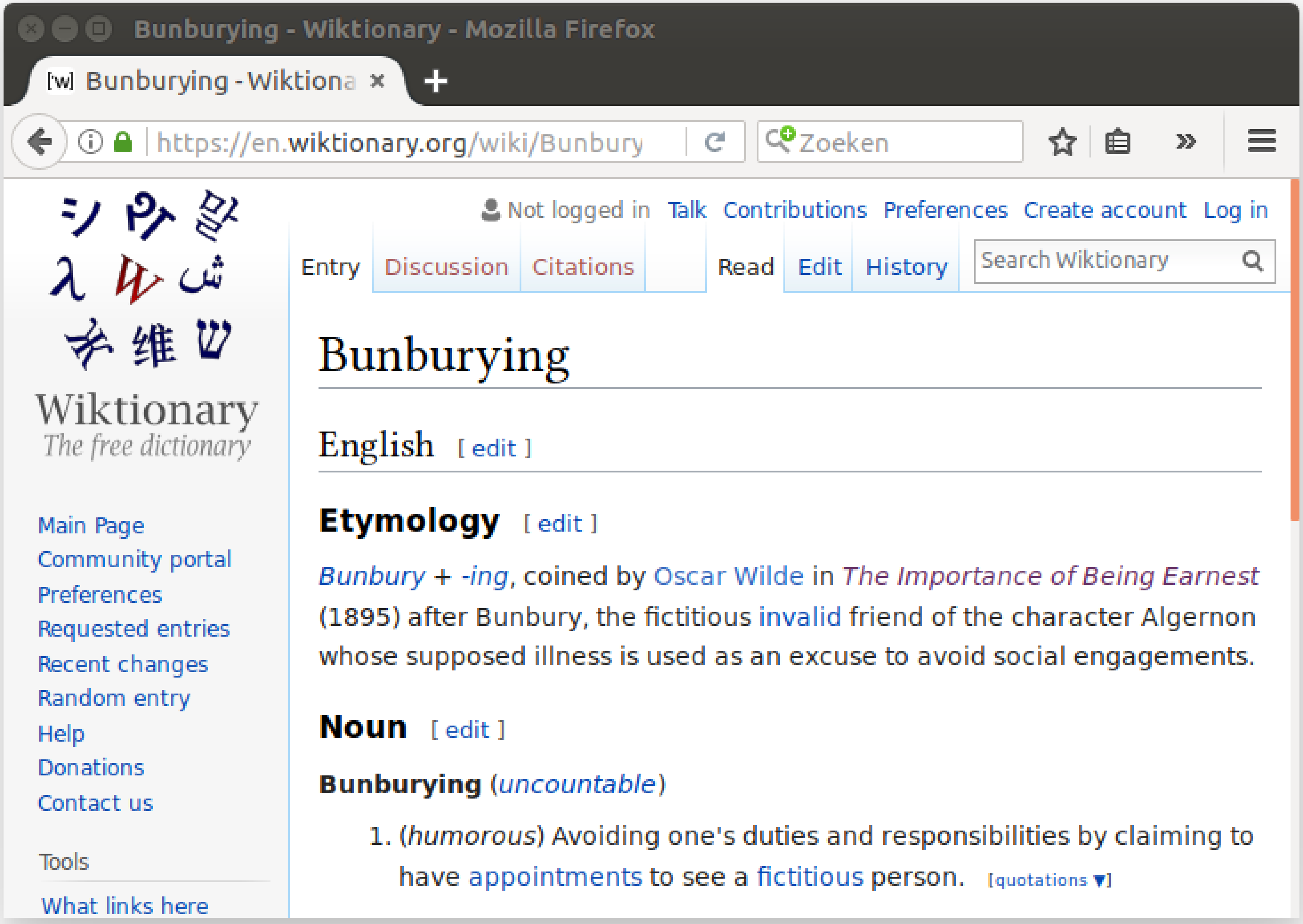
Talk to stub server via the dbus API

- https://www.freedesktop.org/wiki/Software/systemd/resolved/

`systemd-resolved.service`

systemd-resolved

The Importance of Being an Earnest stub



Bunburying

English [edit]

Etymology [edit]

Bunbury + *-ing*, coined by Oscar Wilde in *The Importance of Being Earnest* (1895) after Bunbury, the fictitious invalid friend of the character Algernon whose supposed illness is used as an excuse to avoid social engagements.

Noun [edit]

Bunburying (*uncountable*)

1. (*humorous*) Avoiding one's duties and responsibilities by claiming to have appointments to see a fictitious person. [quotations ▼]