

23 October 2018

Why DNS Security and Privacy is important



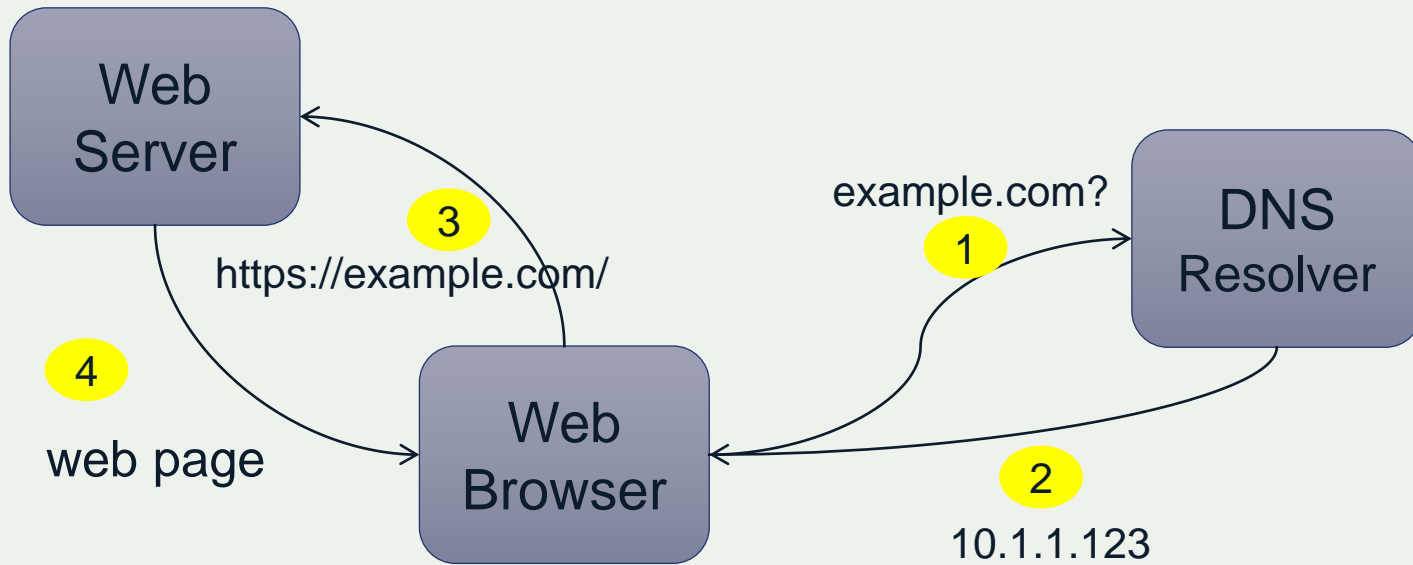
Kevin Meynell
Manager, Technical & Operational
Engagement
meynell@isoc.org

How the DNS works

- The DNS was designed in 1983 and was designed to be a scalable distributed system, operated on a hierarchical basis
- It has proved to be a successful and essential Internet service, but was designed in an era where the Internet was comprised of a relatively small group of users and systems
- Host computers use a DNS resolver to query DNS servers to discover the IP address(es) associated with a domain name
- DNS resolvers traditionally provided by ISPs, although increasingly by third-parties (Google, Cloudflare, Quad9, Dyn et. al.)
- Responses are cached for a specified time period (the Time-to-Live or TTL) so the DNS does not have to be queried again.



A Normal DNS Interaction

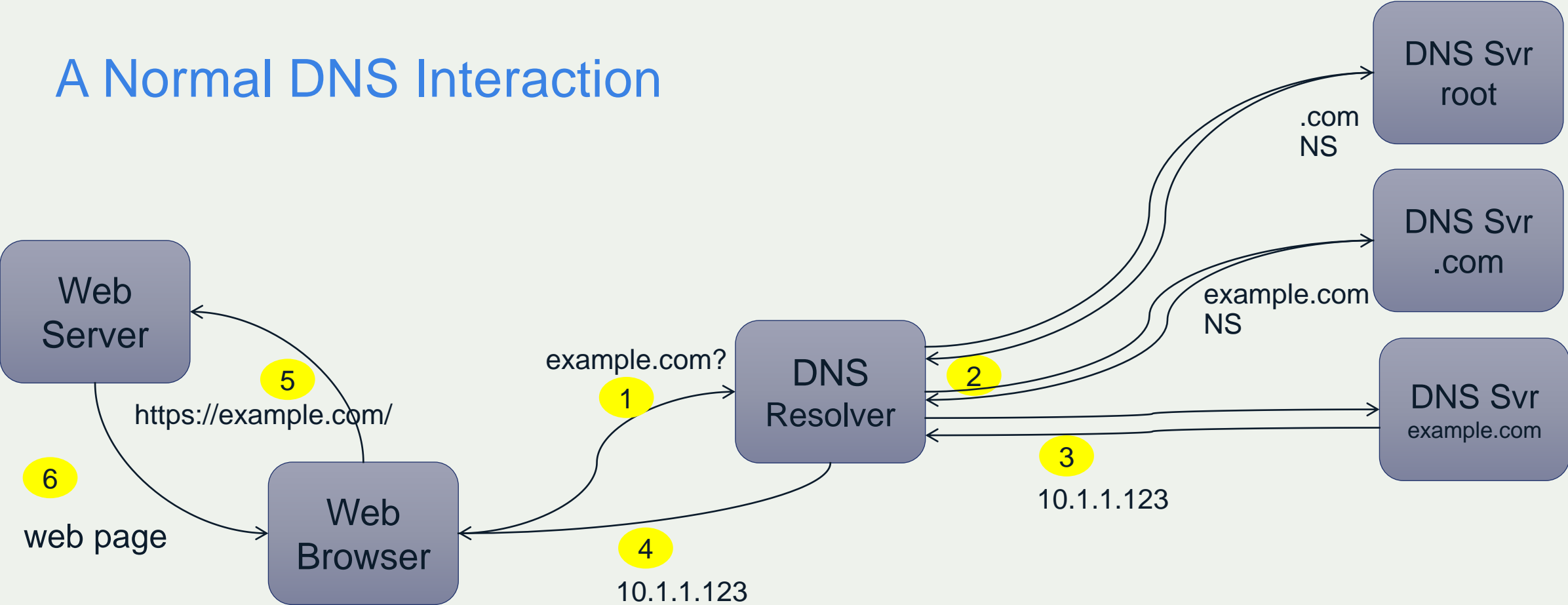


Resolver checks its local *cache*. If it has the answer, it sends it back.

`example.com 10.1.1.123`

If not...

A Normal DNS Interaction



DNS Limitations

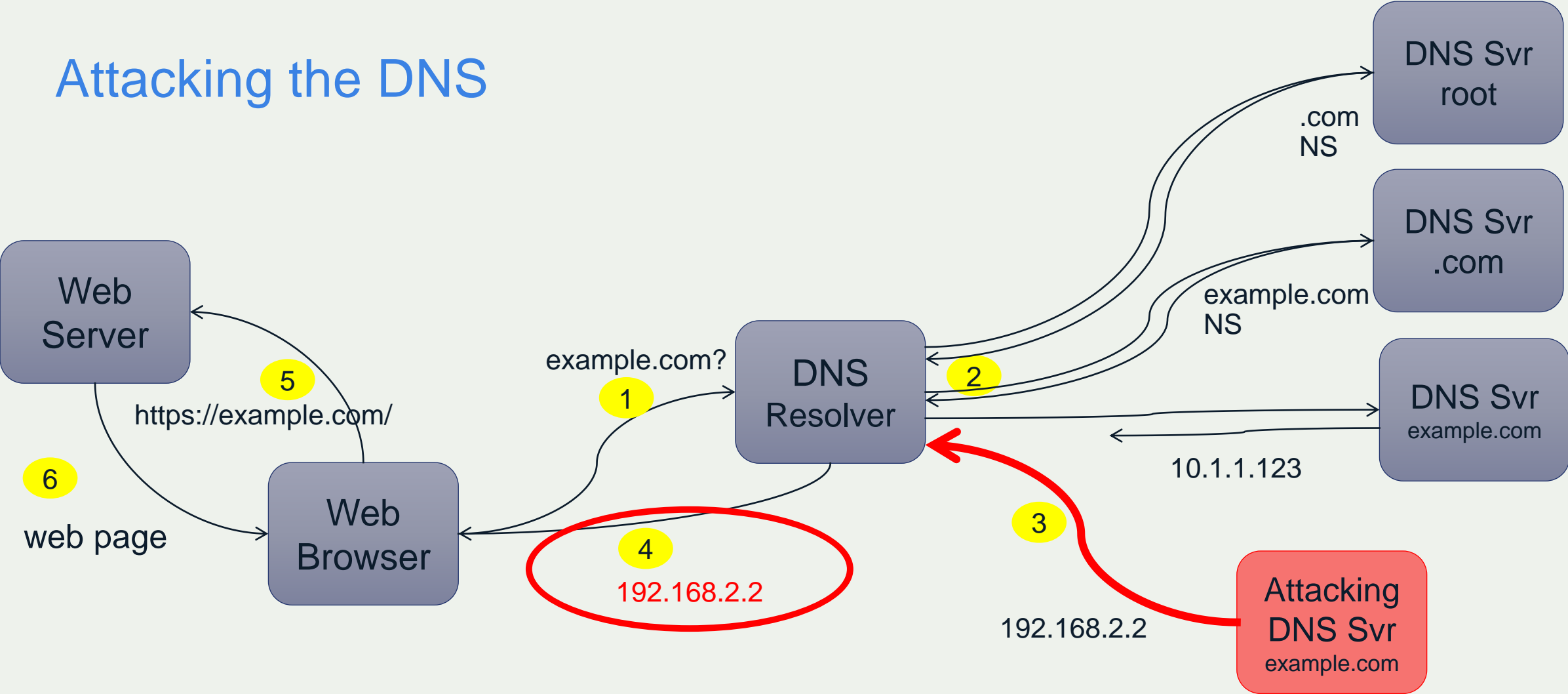
First result received by a DNS resolver is treated as the correct answer

The DNS is reliant on a host trusting that a response to a DNS query is correct

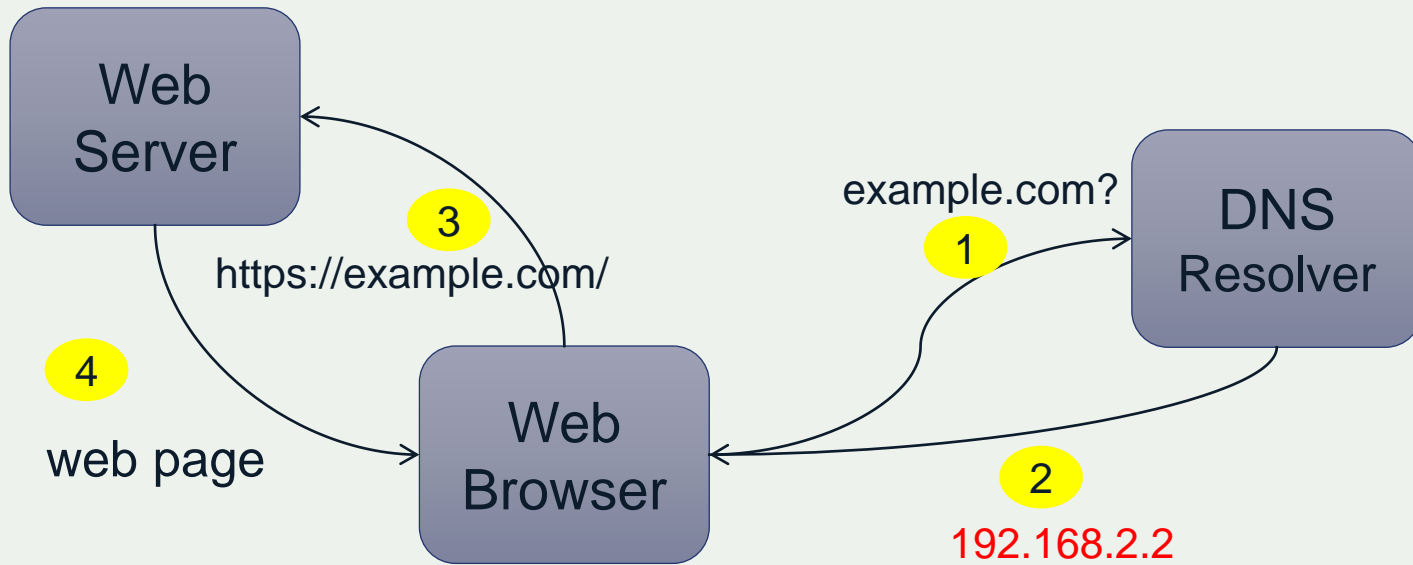
- Can be spoofed (cache poisoning) where incorrect/corrupt DNS data is introduced into DNS resolver,
- OR Man-in-the-Middle attack (e.g. compromised access router) directing DNS queries to a name server that returns forged responses
- Incorrect IP addresses result in traffic being diverted to another computer (e.g. that of an attacker)

DNS queries are not encrypted ← more on this later

Attacking the DNS



A Poisoned Cache



Resolver **cache** now has wrong data:

example.com **192.168.2.2**

This stays in the cache until the Time-To-Live (TTL) expires!

How does DNSSEC help?

DNSSEC = “DNS Security Extensions”

- Defined in RFCs 4033, 4034, 4035 with Operational Practices in RFC 4641
- Set of extensions to DNS that allow resolvers to authenticate the origin of data in the DNS (i.e. has not been modified in transit)
- Also provides authenticated denial-of-existence of DNS records
- DNSSEC introduces new DNS records:
 - RRSIG – a signature (‘hash’) of a set of DNS records (e.g. ‘www.isoc.org’, ‘mail.isoc.org’)
 - NSEC3 – used by resolvers to verify the non-existence of a record
 - DNSKEY – a public key that a resolver can use to validate RRSIG, authenticated by chain-of-trust
 - DS – holds the name of a delegated zone (e.g. ‘isoc.org’), referencing the hash of a DNSKEY record



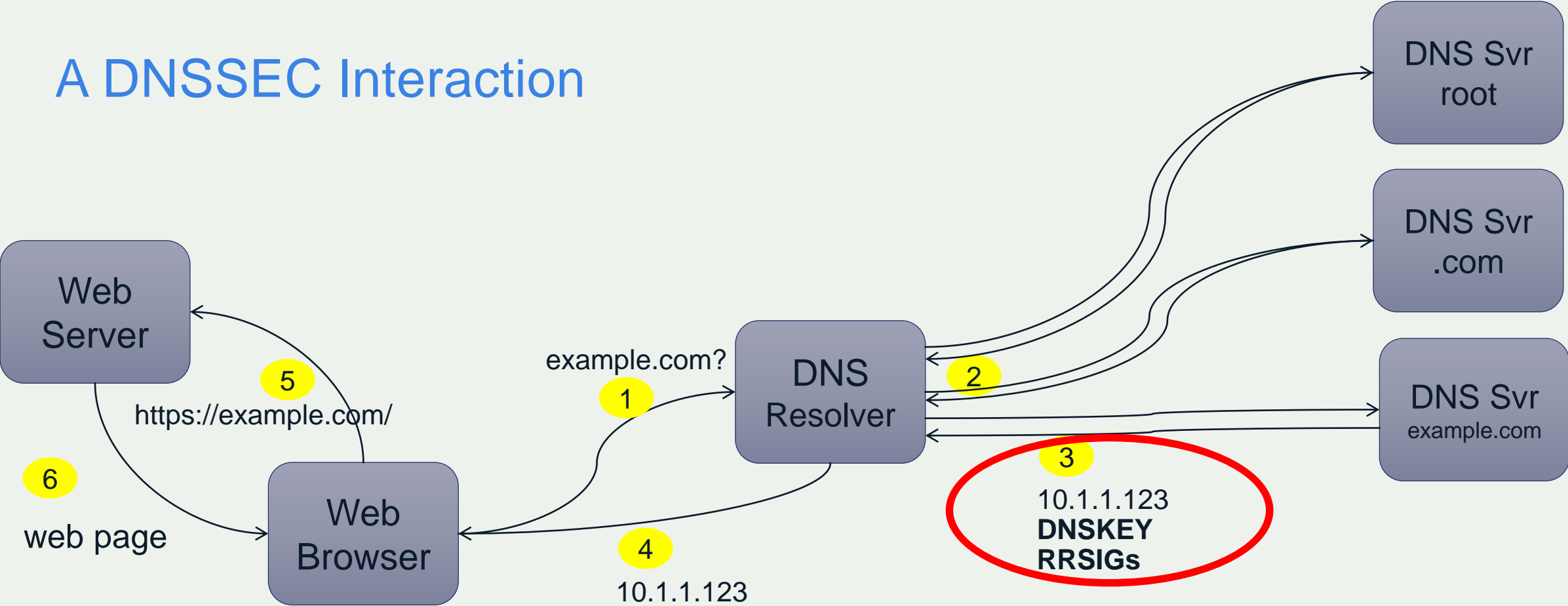
How does DNSSEC help?

DNSSEC works by digitally-signing DNS records using public key cryptography

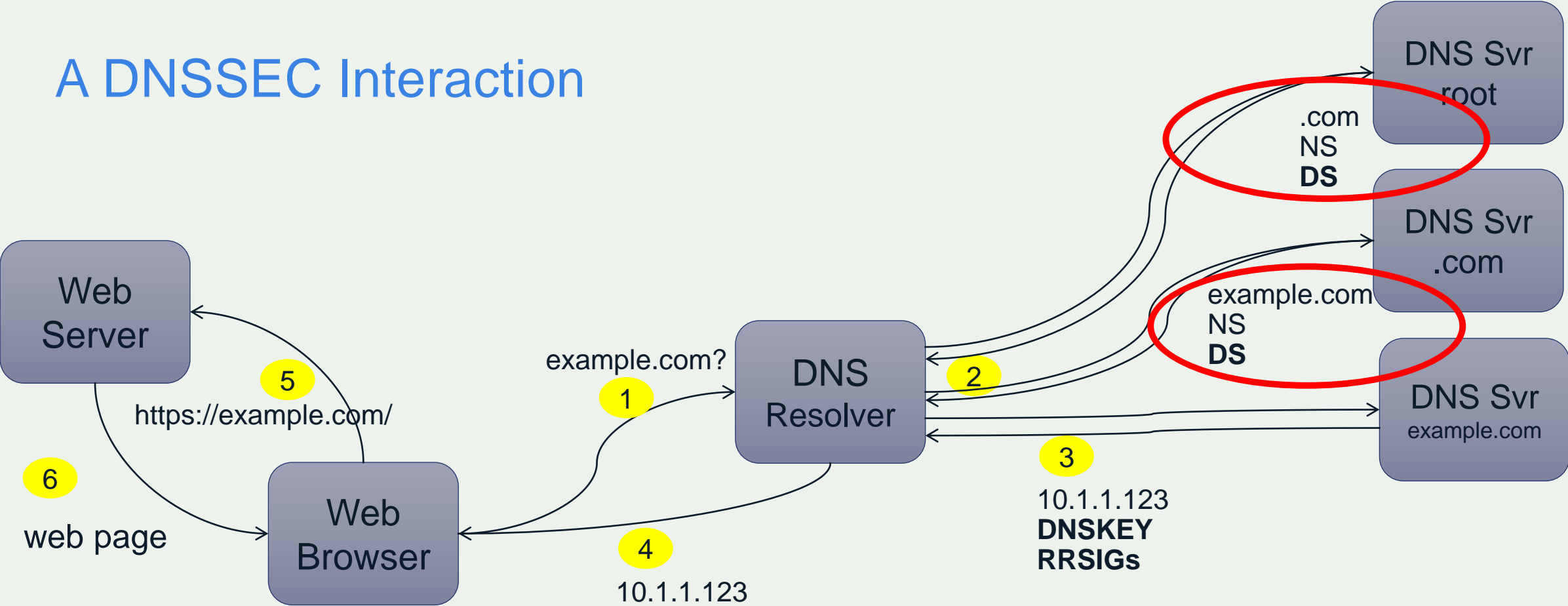
- Records can be authenticated via a chain-of-trust, starting with set of verified public keys for DNS root zone which is the trusted third party
- Domain owners sign their DNS zones with their own private keys, with authenticity of these keys established by the parent zone signing a hash of these keys, and so on up to the root zone in the hierarchy
- This is why it's important for TLDs to sign their zones, otherwise the chain-of-trust breaks!

Ensures that information entered into DNS by the domain name holder is the same information received by the end user.

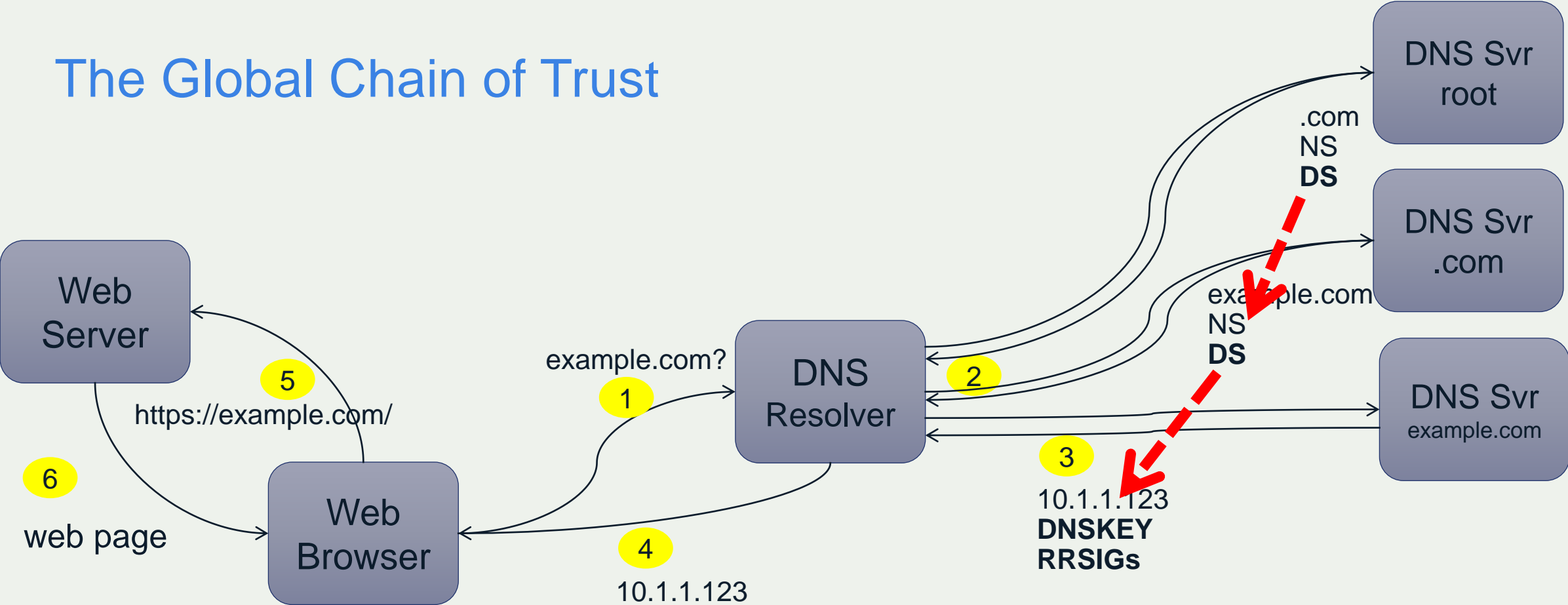
A DNSSEC Interaction



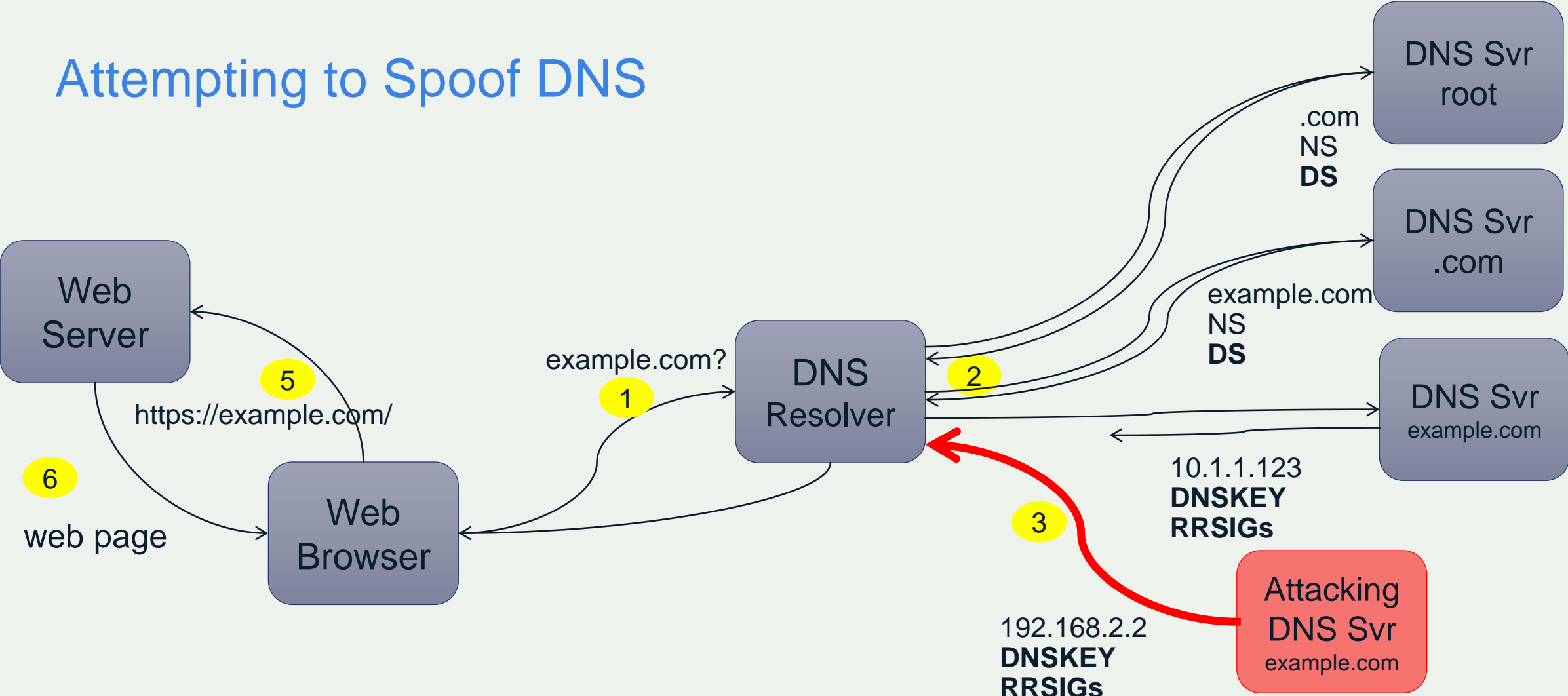
A DNSSEC Interaction



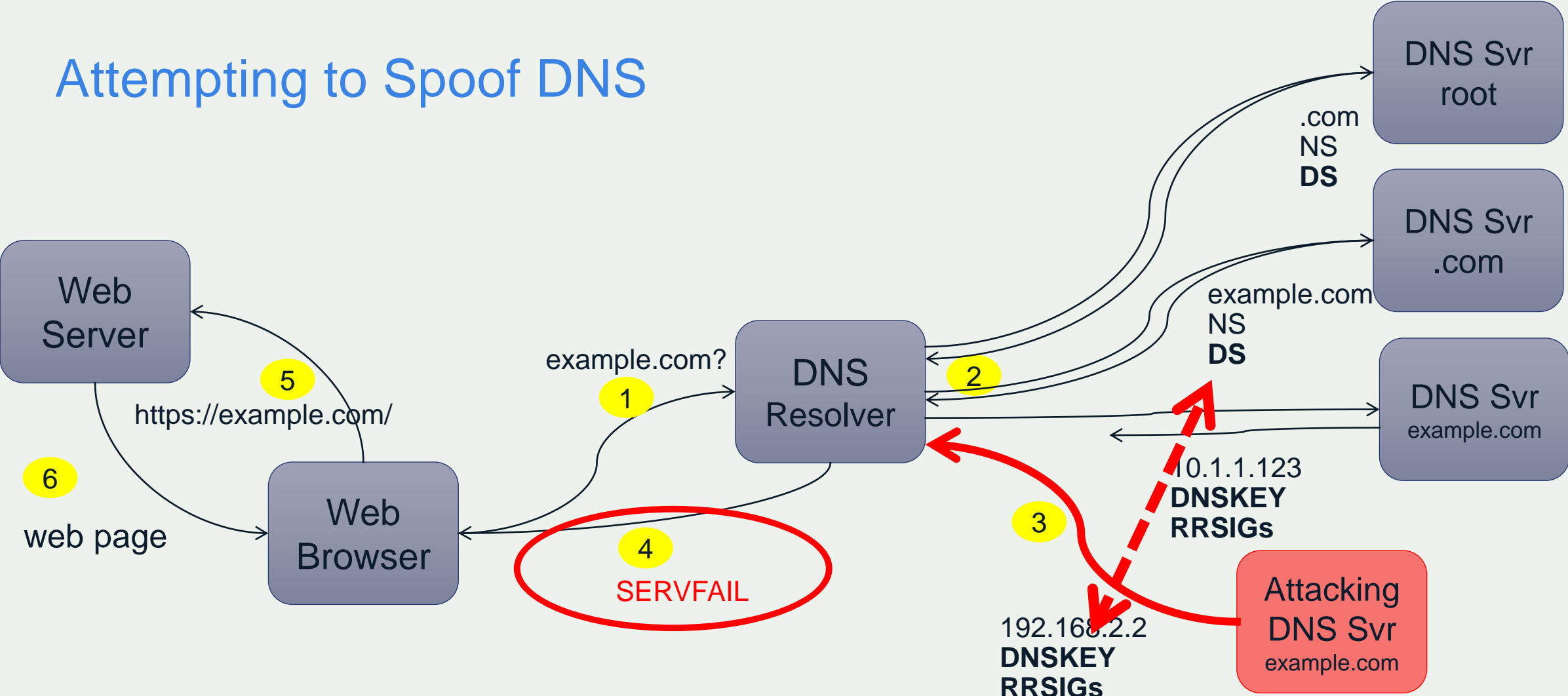
The Global Chain of Trust



Attempting to Spoof DNS



Attempting to Spoof DNS

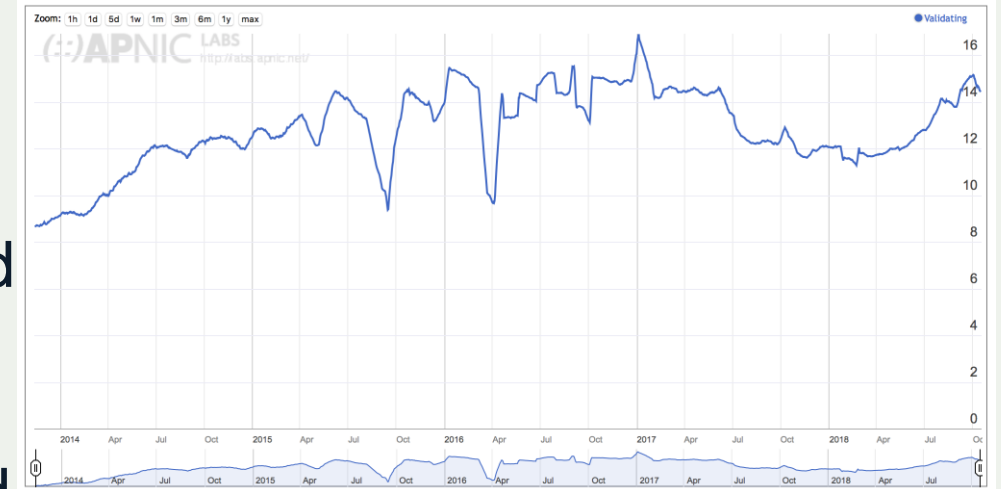


DNSSEC Validation – Current State

- About 15% of all global DNS queries validated
- 34.4% of Central Asian DNS queries validated

Uzbekistan (.uz)	46.8%
Tajikistan (.tj)	21.0%
Kazakhstan (.kz)	14.9%
Kyrgyzstan (.kg)	13.8%
Turkmenistan (.tm)	0%

Use of DNSSEC Validation for World (XA)

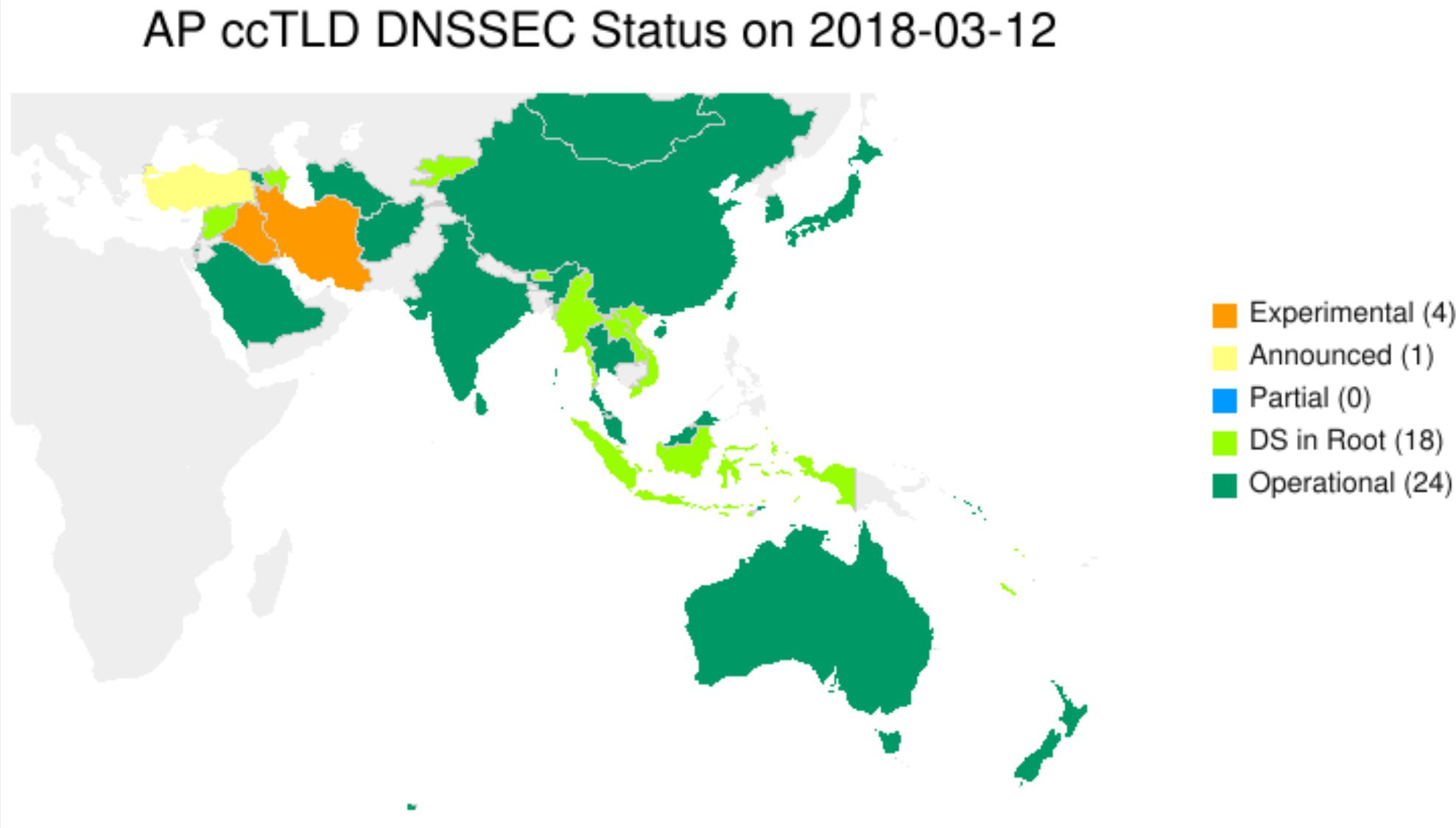


<http://stats.labs.apnic.net/dnssec>

Use of DNSSEC Validation for Central Asia (XR)



DNSSEC Deployment by ccTLD (Asia)




Why do I need DNSSEC if I already have a TLS/SSL certificate?

A TLS (formerly SSL) X.509 certificate serves two functions:

- Authenticates the identity of domain name holder
- Used to encrypt web connections using HTTPS, but can also be used for e-mail, instant messaging, streaming, VoIP and other applications

However...

- Third-party CAs are (currently) needed to validate domain holders
- Root certificate trust established through distribution of root certificates in operating systems or browsers (most commonly Microsoft, Apple and Mozilla certification programmes)
- Inherent weakness is that CAs are able to issue certificates for any name or organisation
- CAs have issued incorrect certificates in the past, and risk increases as number of CAs increases

 Not ideal for automated systems such as e-mail servers

DNS-Authenticated Name Entities (DANE)

DANE is a protocol that allows certificates to be cryptographically bound to DNS names

- Domain administrators can certify the keys used for establishing TLS sessions by storing them in the DNS
- Aims to improve inherent situation whereby any CA can issue certificate for any name
- Requires DNS records to be signed with DNSSEC
- Uses TLSA records to associate a TLS certificate or public key with a domain name
- Clients need to be DNSSEC and DANE aware

DNS TLSA Records

DANE is a protocol that allows certificates to be cryptographically bound to DNS names

- DNS TLSA records associate certificates or public keys with domains
- Defined in RFC 6698
- Usage, Selector and Matching Types fields determine which CAs, certificates and hashes can be accepted
- TLSA record for mx.go6lab.si

```
_25._tcp.mx.go6lab.si. IN    TLSA    3 0 1  
B4B7A46F9F0DFEA0151C2E07A5AD7908F4C8B0050E7CC25908DA05E2 A84748ED
```

The above is a hash of TLS certificate on mx.go6lab.si

Let's Encrypt & DANE Experiment

DANE is a protocol that allows certificates to be cryptographically bound to DNS names

- Internet Society & Go6Lab experimented with DANE & e-mail
- Setup DNSSEC on PowerDNS and signed with OpenDNSSEC
- Used Postfix mail server with TLS support (3.0.1, although 3.3.1 now available)
- Requested and downloaded Let's Encrypt certificate (<https://www.letsencrypt.com/>)
- Generated TLSA record for mx.go6lab.si (using chaingen)
- Added TLSA record to DNS
- Verified TLS certificates on domains known to have DANE support (e.g. nlnetlabs.nl)
- Developed script that fetched top 1 million Alexa domains and sent an e-mail to each of them

Experiment Results

ISOC / Go6Lab tested SMTP connections to Alexa Top 1 million domain names

- 70% of attempted sessions were encrypted with TLS
- 60% of TLS sessions were established with trusted certificate
- Other sessions used untrusted certificate or opportunistic TLS
- Although only 128 sites could be verified with TLSA by DANE ☹️



Moral of the Story

70% of e-mail can already be encrypted in some way – so enable TLS on your server!

Let's Encrypt offers free certificates and automated ordering/installation, so no excuse not to make encryption the default on the Internet

Using DANE is not that difficult and can be automated

But...

We need more DNSSEC signed domains!

We need even more DANE/TLSA verified servers!



What DNSSEC (and DANE) does not do!

Does not ensure the correctness of DNS records – only that they haven't been modified without the owner's consent

Does not protect against threats against hosts (e.g. DDoS attacks, buffer overruns in code)

Does not keep DNS records private – either on the servers or 'on the wire'

- The IETF has developed mechanisms to provide confidentiality to DNS transactions through the DNS PRIVate Exchange (DPRIVE) Working Group
- Aim is to encrypt queries and responses to/from DNS servers to prevent pervasive monitoring

DNS Privacy

Most activities on the Internet start with a DNS query – translate human readable names to IP addresses


The DNS is a globally distributed system crossing international boundaries and using servers in many countries to provide resilience.

DNS queries are (by default) sent in clear text using UDP or TCP which allows passive eavesdropping (some VPNs also leak DNS queries)

DNS queries reveal what sites an individual (or host) is communicating with

Some ISPs log DNS queries to their resolvers and share this information with third-parties in ways not known or obvious to end users.

Some ISPs embed also user information (e.g. MAC address) within DNS queries to their resolvers that allows fingerprinting of individual users

Whilst the data in the DNS is public, individual transactions made by an end user should  not be public

DNS Privacy: The Solutions

The IETF DNS PRIVate Exchange (DPRIVE) Working Group has recently developed mechanisms to encrypt queries and responses to/from DNS resolvers and provide confidentiality to DNS transactions

- DNS-over-TLS (DoT) RFC 7858
- DNS-over-DTLS RFC 8094 ← currently no implementations
- DNS-over-HTTPS (DoH) awaiting RFC publication

DNS Privacy: Implementations

Public Resolvers

- Quad9 (9.9.9.9 and 2620:fe::fe), DoT
- Cloudflare (1.1.1.1, 1.0.0.1, 2606:4700:4700::1111 and 2606:4700:4700::1001), DoT & DoH
- CleanBrowsing (various), DoT & DoH

Clients

- Stubby 1.3+, DoT
- Unbound, 1.6.7+, DoT
- Knot resolver 2.0+, DoT
- Mozilla Firefox 62+, DoH
- Android 9 Pie, DoT



DNS Privacy: Caveats

Resolvers and clients need to be upgraded to support DoT and DoH

Currently limited support for DoT and DoH in operating systems

DoT and DoH only encrypt DNS communications between client (stub-resolver) and recursive resolver, not between recursive resolver and authoritative DNS servers

Resolver provider has potential to monitor and log transactions, so needs to be trusted

DoT and DoH does not ensure the integrity of information returned from an authoritative server, so DNSSEC should also be used to cryptographically assert DNS entries are correct.

BUT... important components in improving the security and confidentiality of the DNS²⁷

DNS Flag Day

Important to be aware of...

- DNSSEC and other extended features of the DNS require EDNS0 (Extension Mechanisms for DNS – RFC 6891)
- Properly implemented name servers should either reply with an EDNS0 compliant response, or provide a regular DNS response if they don't understand.
- Workarounds had to be incorporated to allow resolvers to retry if name servers don't respond correctly, but these cause unnecessary retries, delays, and prevent the newer features of the DNS being used.
- The vendors of the most commonly used DNS software (BIND, Unbound, PowerDNS and Knot) will remove these workarounds as of 1 February 2019
- Consequence is that hostnames served by broken DNS implementations will no longer be resolved

📢 Please check if your domain is affected! <https://dnsflagday.net>

References

Internet Society Deploy360

<https://www.isoc.org/deploy360/>

DNSSEC Deployment Maps

<https://www.isoc.org/deploy360/dnssec/maps/>

APNIC Labs DNSSEC Measurements

<https://stats.labs.apnic.net/dnssec/>

DNS Privacy Project

<https://dnsprivacy.org/>

Let's Encrypt

<https://www.letsencrypt.com/>

HOWTO Implementing DANE

<https://www.isoc.org/blog/2016/01/lets-encrypt-certificates-for-mail-servers-and-dane-part-1-of-2/>

Thank you.

Kevin Meynell
Content & Resource Manager
meynell@isoc.org

Visit us at
www.internetsociety.org
Follow us
[@internetsociety](https://twitter.com/internetsociety)

Galerie Jean-Malbuisson 15,
CH-1204 Geneva,
Switzerland.
+41 22 807 1444

1775 Wiehle Avenue,
Suite 201, Reston, VA
20190-5108 USA.
+1 703 439 2120

