DNS and its Security

CSE 548: Advanced Computer Network Security
(April 17, 2024)
Sana Habib

Outline

- What is DNS?
- DNS Threats and the Need for DNSSEC
- Object Security: DNSSEC
- Path Security: DOT, DOH, TLSv1.2,1.3, ESNI, ECH,
- Some Exercises

What the novice Internet user sees?

website
www.example.com

What a Techy Internet user sees?

HTTP
Server

93.184.216.34!

Step 1:
www.example.com?

DNS
Server
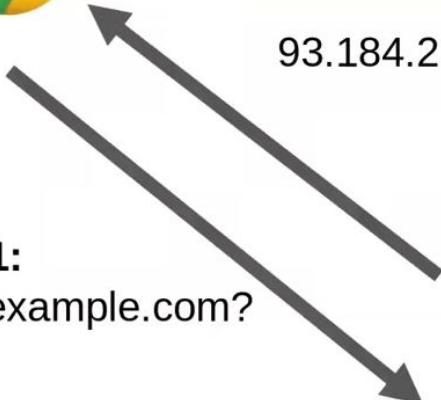
Step 2:
HTTP to 93.184.216.34

HTTP
Server

93.184.216.34!

Step 1:
www.example.com?

DNS
Server

5
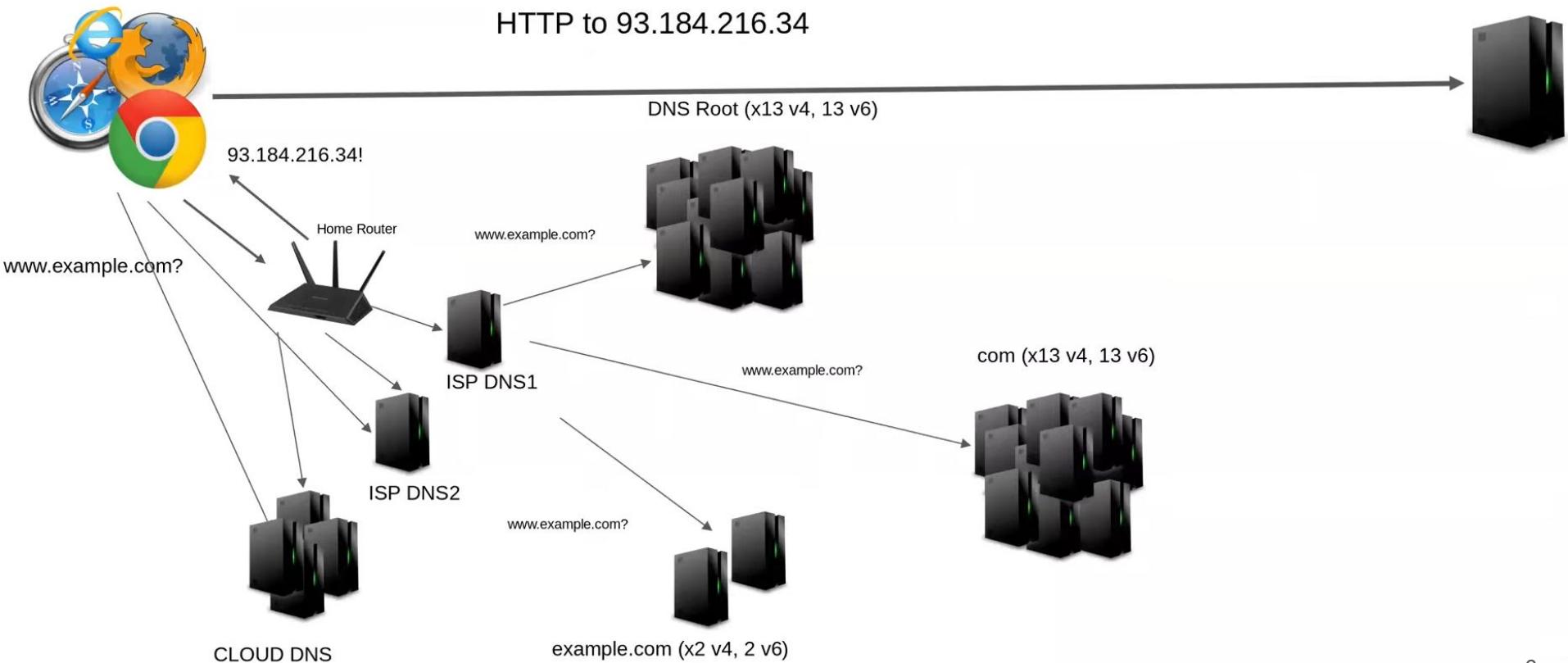
DNS is much much more complex!
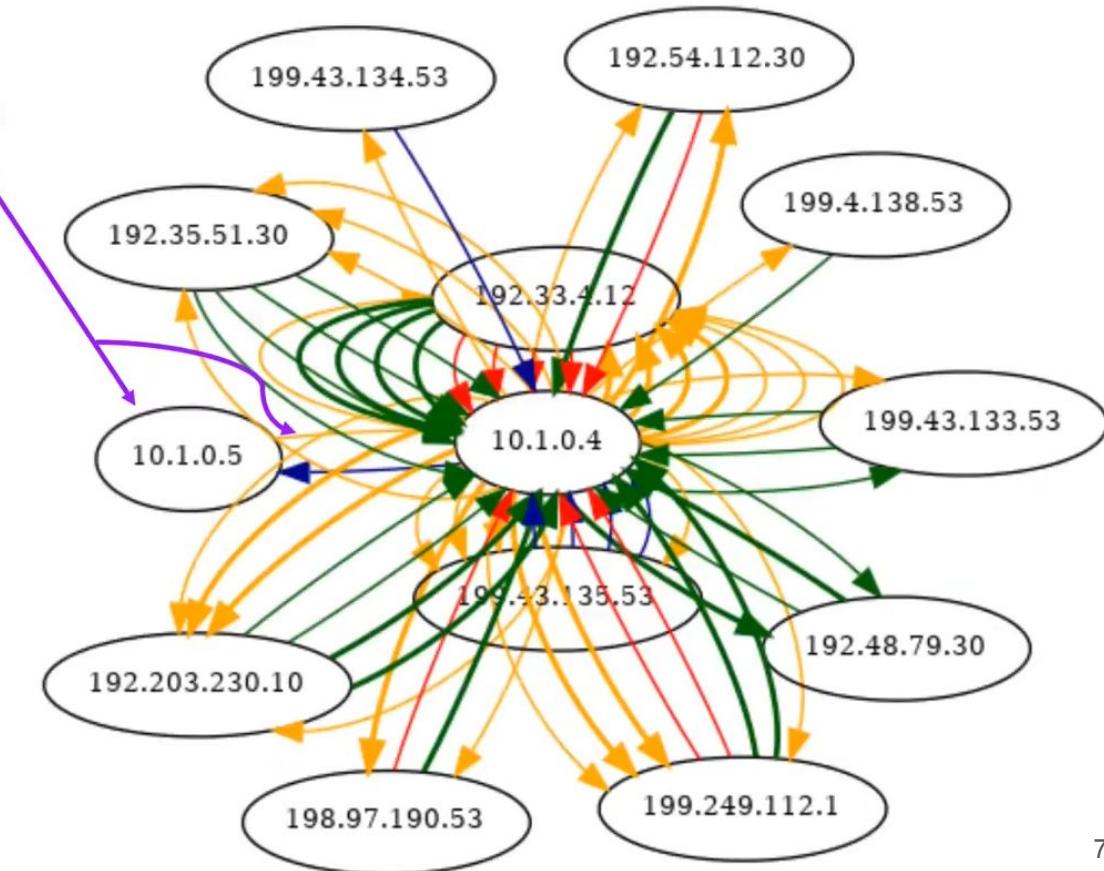
The example.com web page

You make a single request

- Each line is a DNS request or response
- The center node is an ISP resolver

199.43.134.53
192.54.112.30
192.35.51.30
199.4.138.53
192.33.4.12
199.43.133.53
10.1.0.5
10.1.0.4
199.43.135.53
192.48.79.30
192.203.230.10
198.97.190.53
199.249.112.1
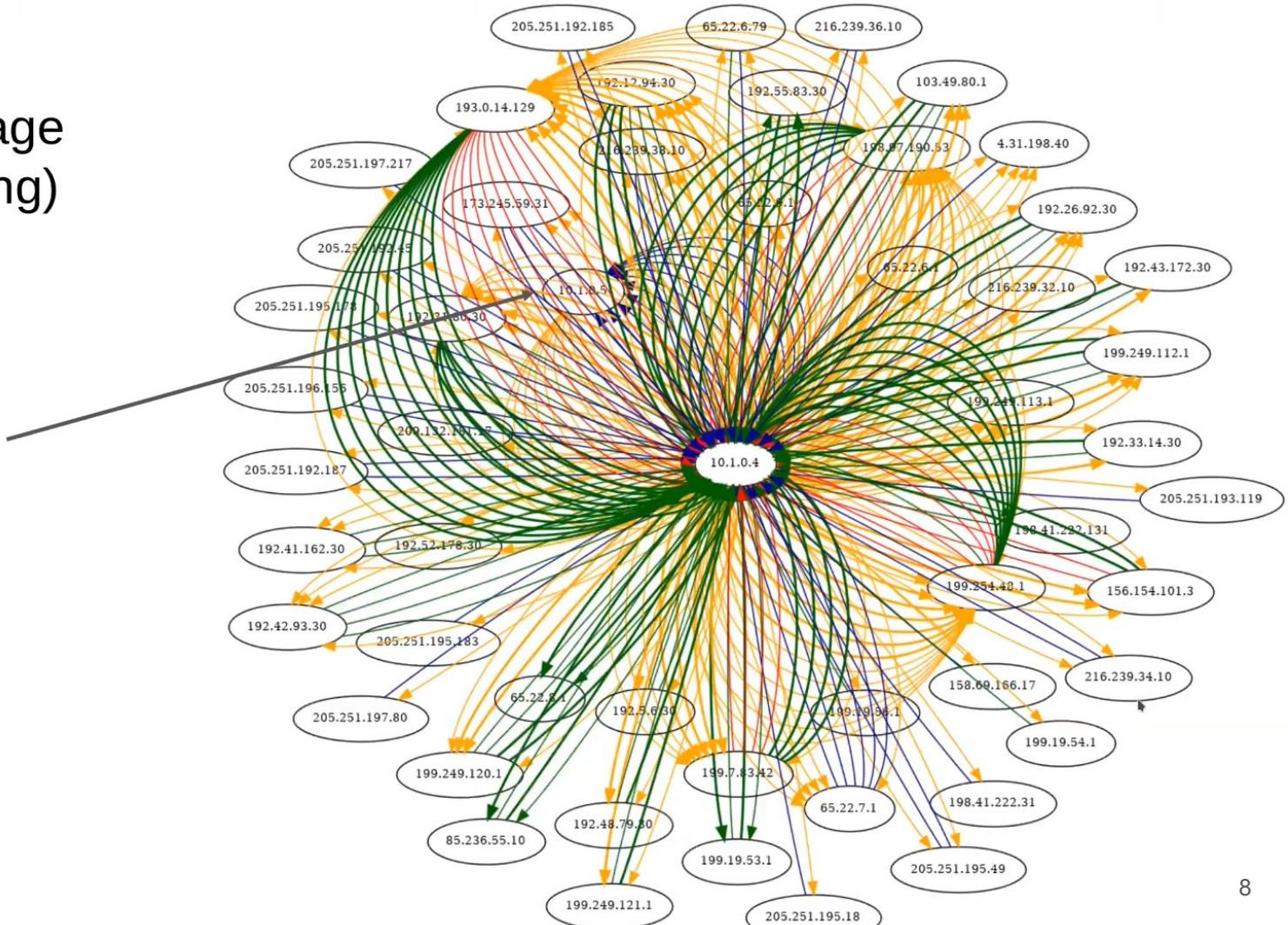
Query
Authoratative/DNSSEC
Truncated
Response
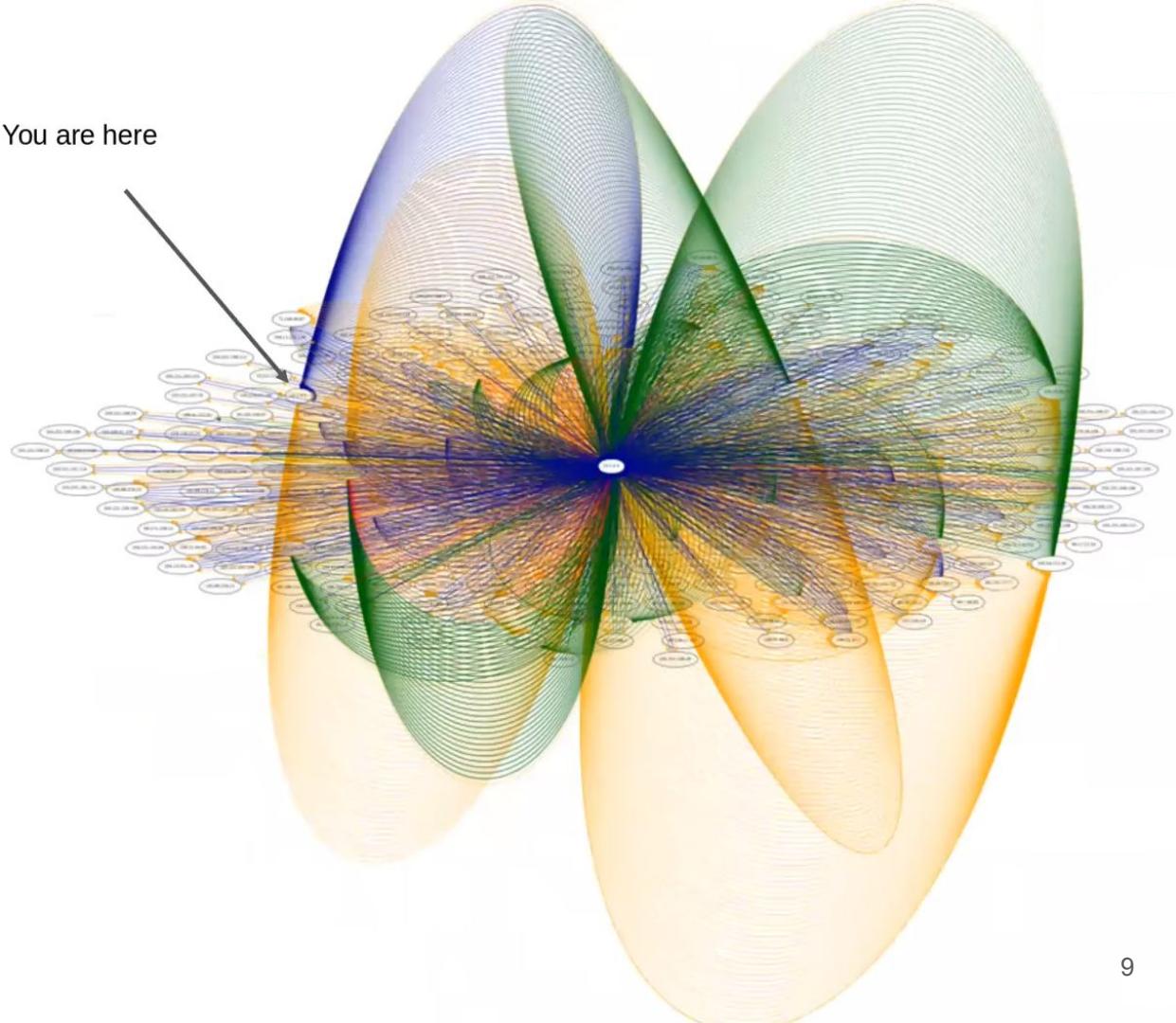
ietf.org web page (without caching)

You are here
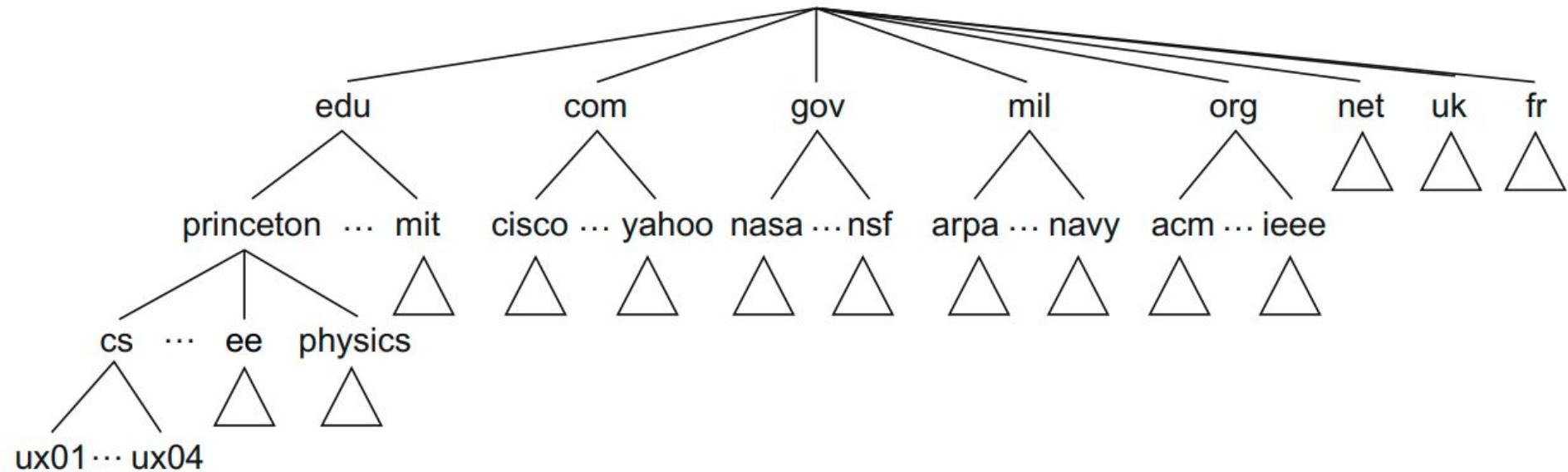
webmd.com

You are here

Example of Domain Hierarchy

https://cseweb.ucsd.edu/classes/wi19/cse124-a/courseoverview/compnetworks.pdf

What are parts of domain?

https://www.google.com,

Protocol, Subdomain, Domain Name, Top-level Domain

Root Domain (includes domain name and top level domain)

How about https://www.amazon.com/ and https://leetcode.com/?

Who maintains domains?

The Top Level Domain (TLD) or Top Level Zone is maintained by the Internet Corporation for Assigned Names and Numbers (ICANN) https://www.icann.org/

How DNS records are kept? Resource Records (RR)

Each name server implements the zone information as a collection of Resource Records (RR).

RR is a 5-tuple: <Name, Value, Type, Class, TTL>

Type → A, NS, CNAME, MX.

A → indicates value is an IP address.

NS → The value field gives the domain name for a host that is running a name server that knows how to resolve names within the specified domain.

CNAME—The value field gives the canonical name for a particular host; it is used to define aliases.

MX→The value field gives domain name for a host (running a mail server).

TTL → time to live shows how long a resource record is valid.

Linux Commands:

dig (domain information groper) → used for retrieving information about domain name servers.

nslookup (name server lookup) → allows you to query DNS service.

host → look up a variety of information available through the domain name system.

dog → a command-line DNS client for looking up DNS records for domain names.

whois → is a protocol that is used to look up information on domain names and IP addresses.

```
sana@pop-os:~$ dig google.com

; <<>> DiG 9.18.12-0ubuntu0.22.04.2-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51819
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;google.com.                    IN      A

;; ANSWER SECTION:
google.com.             221     IN      A       142.250.138.100
google.com.             221     IN      A       142.250.138.138
google.com.             221     IN      A       142.250.138.101
google.com.             221     IN      A       142.250.138.113
google.com.             221     IN      A       142.250.138.102
google.com.             221     IN      A       142.250.138.139

;; Query time: 3 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Wed Sep 20 01:57:20 PDT 2023
;; MSG SIZE  rcvd: 135
```

The first field is the NAME: The domain name being returned.

The second field (221) is the TTL in seconds.

IN is the CLASS. Here, IN stands for Internet.

A is the TYPE. Here, A stands for mapping a domain name to an IPv4 address.

The last field is the IP address.

```
sana@pop-os:~$ nslookup google.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.113.139
Name:   google.com
Address: 142.250.113.113
Name:   google.com
Address: 142.250.113.101
Name:   google.com
Address: 142.250.113.100
Name:   google.com
Address: 142.250.113.102
Name:   google.com
Address: 142.250.113.138
Name:   google.com
Address: 2607:f8b0:4023:1000::65
Name:   google.com
Address: 2607:f8b0:4023:1000::64
Name:   google.com
Address: 2607:f8b0:4023:1000::8b
Name:   google.com
Address: 2607:f8b0:4023:1000::66
```
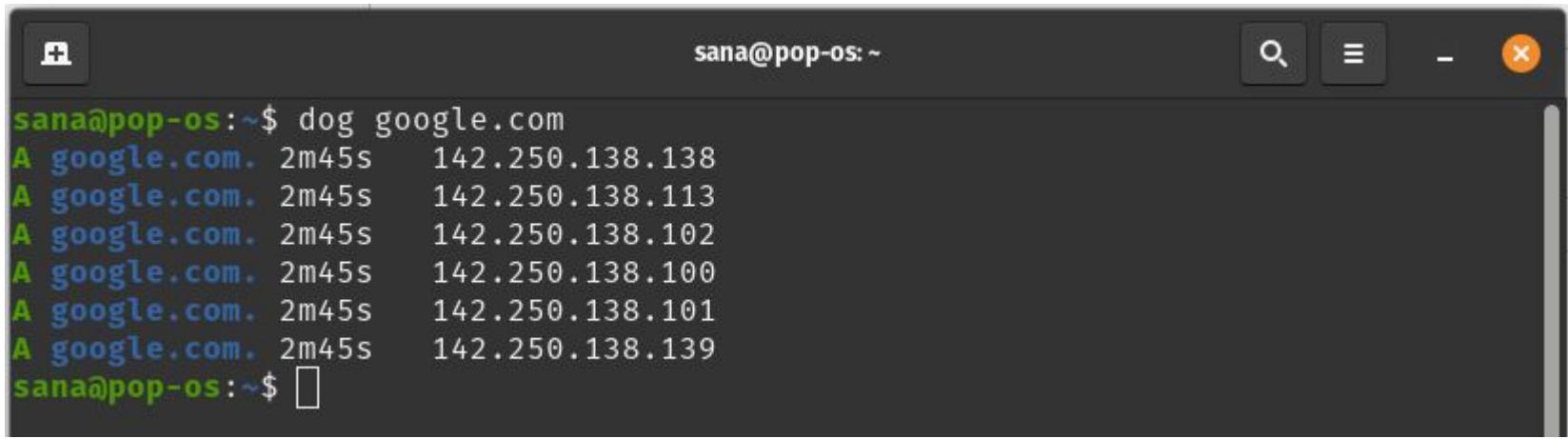
19
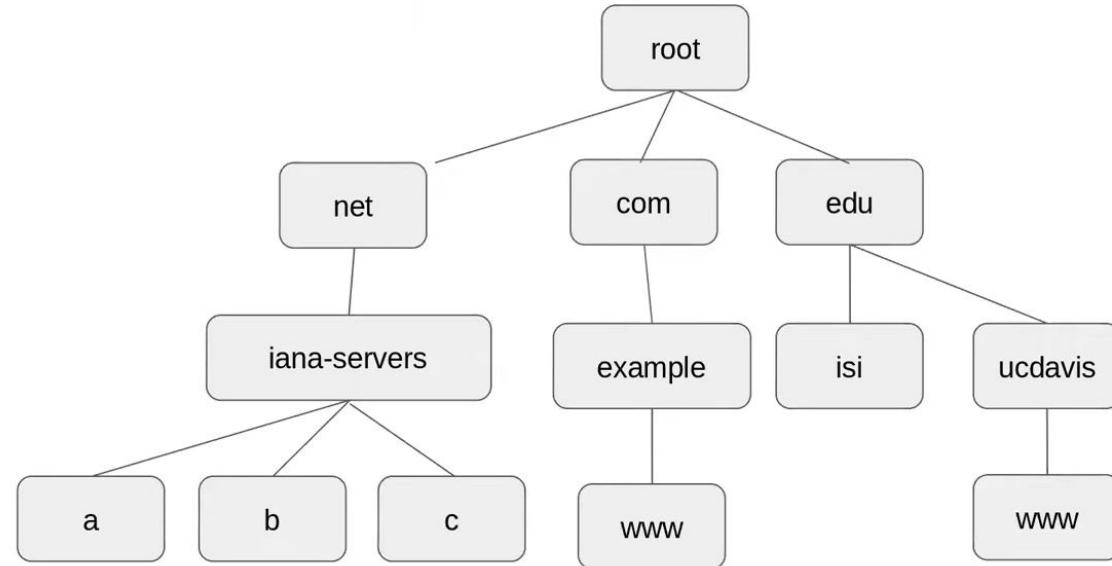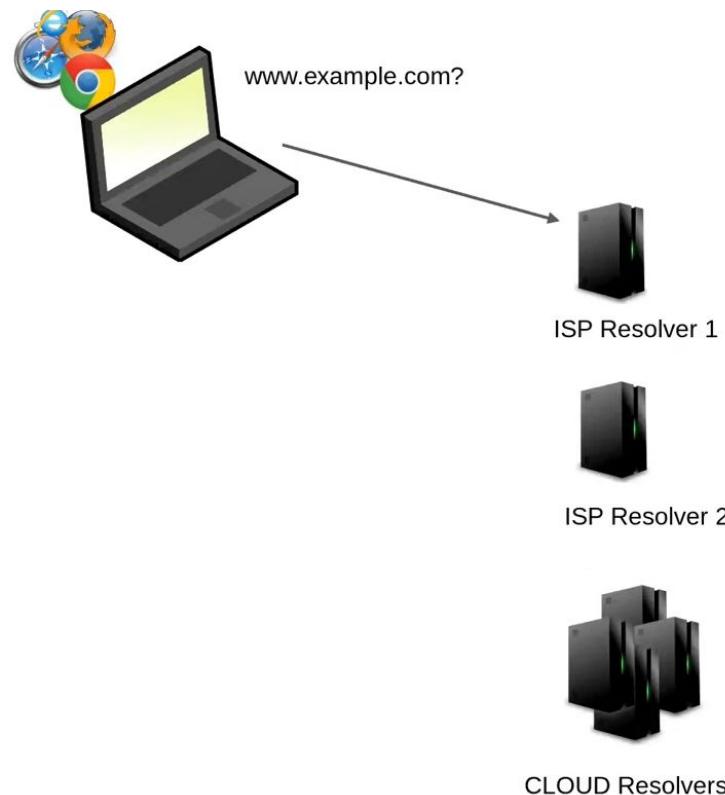
Subject Alternate Names (SAN)
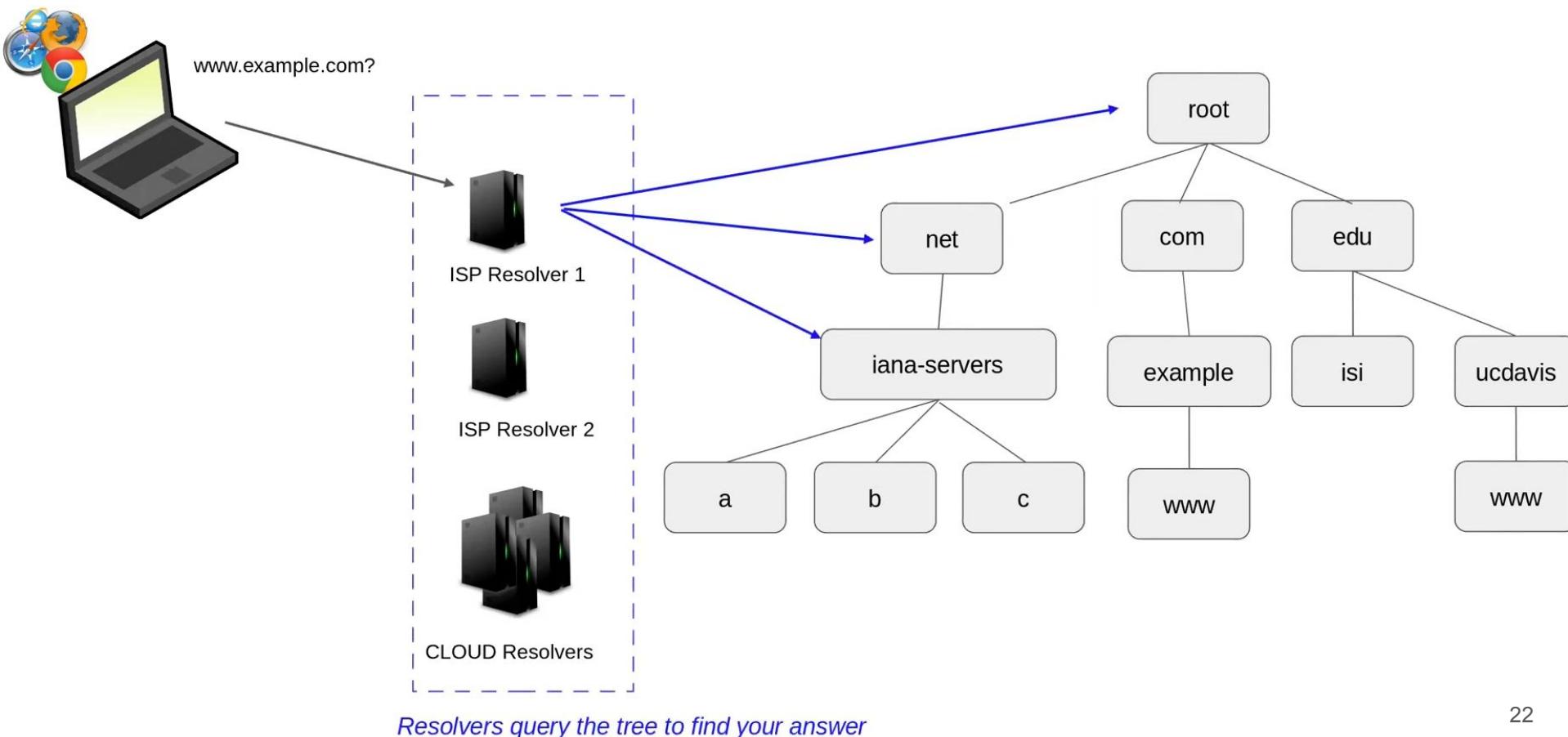
Alternate domain names for the same website.

Exercise: What are the SANs for some of your favorite websites?
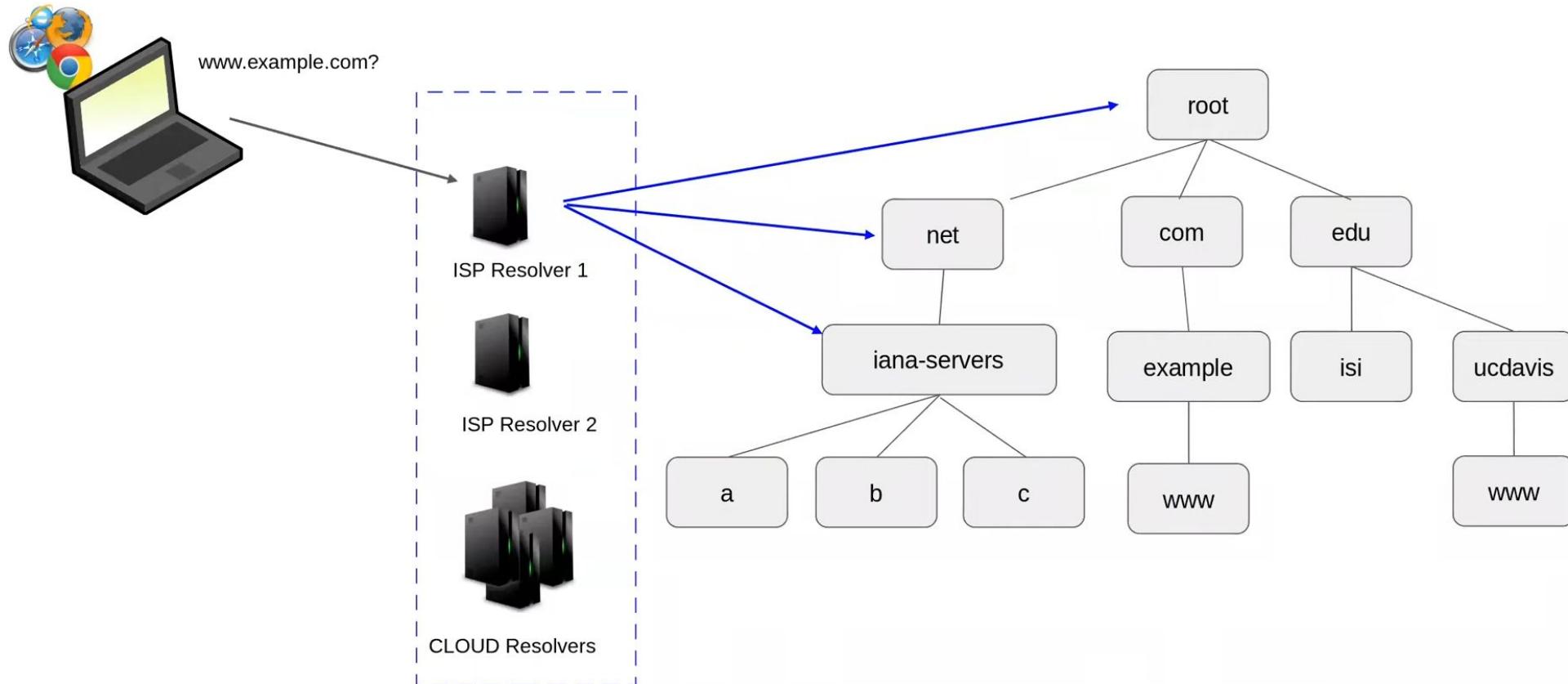
DNS Resolution: Resolvers do all the hard work!

Resolvers do all the hard work!

www.example.com?

ISP Resolver 1

ISP Resolver 2

CLOUD Resolvers

root

net

com

edu

iana-servers

example

isi

ucdavis

a

b

c

www

www

Resolvers query the tree to find your answer

Resolvers do all the hard work!

www.example.com?

ISP Resolver 1

ISP Resolver 2

CLOUD Resolvers

root

net   com   edu

iana-servers   example   isi   ucdavis

a   b   c   www   www

Resolvers query the tree to find your answer   *AND REMEMBER THIS ANSWER!*

23

Resolvers do all the hard work!

www.example.com?

DNS resolver types:
- Stub
- Recursive
- Forwarders
- Validating
- Pay Wall
(to be described later)

ISP Resolver 1

ISP Resolver 2

CLOUD Resolvers

root

net

com

edu

iana-servers

example

isi

ucdavis

a

b

c

www

www

Resolvers query the tree to find your answer   *AND REMEMBER THIS ANSWER!*

Structure of DNS…… The DNS 'tree'
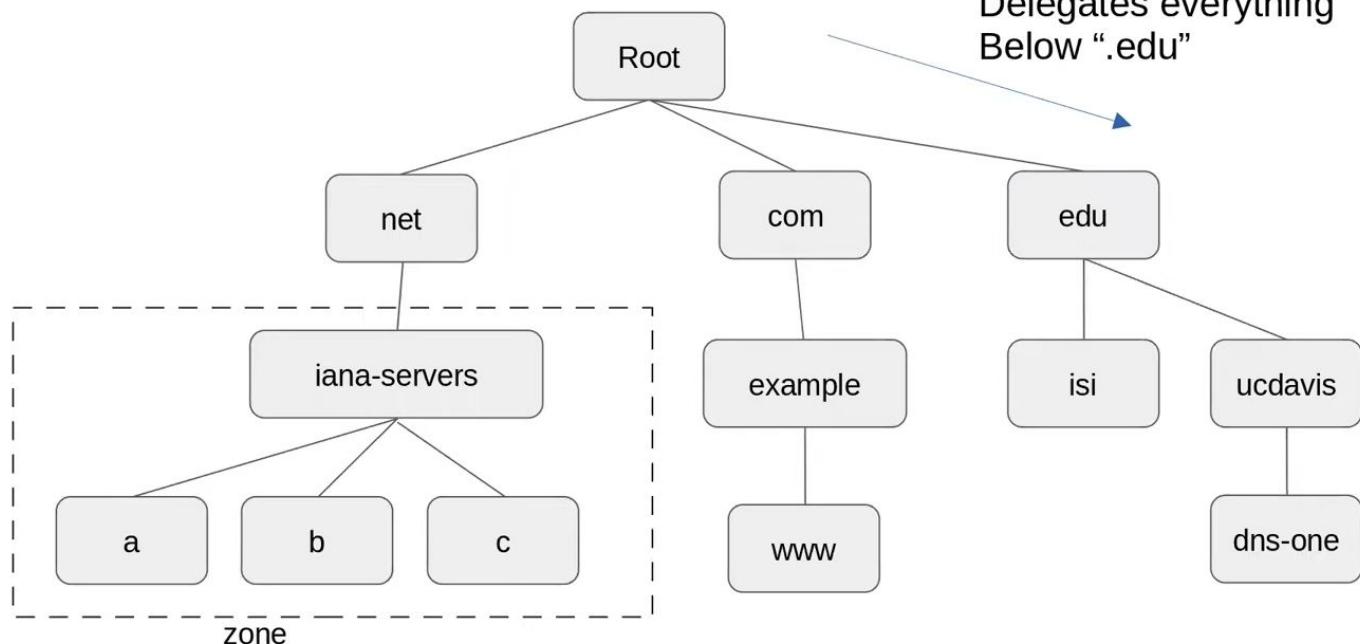
The Root (aka ".")

Top Level Domains
(TLDs)

Second Level Domains
(SLDs)

Structure of DNS…… The DNS 'tree'

The Root (aka ".")

Top Level Domains
(TLDs)

Second Level Domains
(SLDs)

Root

Delegates everything
Below ".edu"

net   com   edu

iana-servers   example   isi   ucdavis

a   b   c

www

dns-one

zone

Structure of DNS…… The DNS 'tree'

The Root (aka ".")

Top Level Domains (TLDs)

Second Level Domains (SLDs)

Root

Delegates everything
Below ".edu"

.edu delegates
ucdavis.edu

net

com

edu

iana-servers

example

isi

ucdavis

a

b

c

www

dns-one

zone

Some information needed in both parent and child zones

ucdavis.edu. 172800 NS dns-two.ucdavis.edu.
ucdavis.edu. 172800 NS dns-one.ucdavis.edu.
ucdavis.edu. 172800 NS dns-three.ucdavis.edu.

.edu zone

root

edu

com

example

www

ucdavis

dns-one   dns-two   dns-three

.ucdavis.edu zone

28

Some information needed in both parent and child zones

ucdavis.edu. 172800 NS dns-two.ucdavis.edu.
ucdavis.edu. 172800 NS dns-one.ucdavis.edu.
ucdavis.edu. 172800 NS dns-three.ucdavis.edu.

Parents must be able to tell
clients how to get to child-zones

And about crytographic DNSSEC links

.edu zone

root

edu

com

example

www

ucdavis

dns-one

dns-two

dns-three

.ucdavis.edu zone

ucdavis.edu. 172800 NS dns-two.ucdavis.edu.
ucdavis.edu. 172800 NS dns-one.ucdavis.edu.
ucdavis.edu. 172800 NS dns-three.ucdavis.edu.

Parents must be able to tell
clients how to get to child-zones

And about crytographic DNSSEC links

The child is the authoritative source!
(mostly)

.edu zone

root

edu

com

example

www

ucdavis

dns-one

dns-two

dns-three

.ucdavis.edu zone

What happens when things don't match…

```
ucdavis.edu.  172800  NS  dns-two.ucdavis.edu.
ucdavis.edu.  172800  NS  dns-one.ucdavis.edu.
```

```
ucdavis.edu.  172800  NS  dns-one.ucdavis.edu.
ucdavis.edu.  172800  NS  dns-three.ucdavis.edu.
```

.edu zone

root

edu

com

example

www

ucdavis

dns-one

dns-two

dns-three

.ucdavis.edu zone

What happens when things don't match?

If *dns-two.ucdavis.edu* can't answer, this is a "lame delegation" (it's not authoritative, but .edu thinks it is)

```
ucdavis.edu.  172800  NS  dns-two.ucdavis.edu.
ucdavis.edu.  172800  NS  dns-one.ucdavis.edu.
```

```
ucdavis.edu.  172800  NS  dns-one.ucdavis.edu.
ucdavis.edu.  172800  NS  dns-three.ucdavis.edu.
```

.edu zone

root

edu

com

example

www

ucdavis

dns-one

dns-two

dns-three

.ucdavis.edu zone

32

What happens when things don't match?

If *dns-two.ucdavis.edu* can't answer, this is a "lame delegation" (it's not authoritative, but .edu thinks it is)

ucdavis.edu.   172800   NS   dns-two.ucdavis.edu.
ucdavis.edu.   172800   NS   dns-one.ucdavis.edu.

ucdavis.edu.   172800   NS   dns-one.ucdavis.edu.
ucdavis.edu.   172800   NS   dns-three.ucdavis.edu.

Unfortunately **many** zones exist with exactly this problem

The result is timeouts and delays for clients

.edu zone

root

edu

com

example

www

ucdavis

dns-one   dns-two   dns-three

.ucdavis.edu zone

33

What happens when things don't match?

If *dns-two.ucdavis.edu* can't answer, this is a "lame delegation"
(it's not authoritative, but .edu thinks it is)

```
ucdavis.edu.  172800  NS  dns-two.ucdavis.edu.
ucdavis.edu.  172800  NS  dns-one.ucdavis.edu.
```

```
ucdavis.edu.  172800  NS  dns-one.ucdavis.edu.
ucdavis.edu.  172800  NS  dns-three.ucdavis.edu.
```

Unfortunately **many** zones exist with exactly this problem
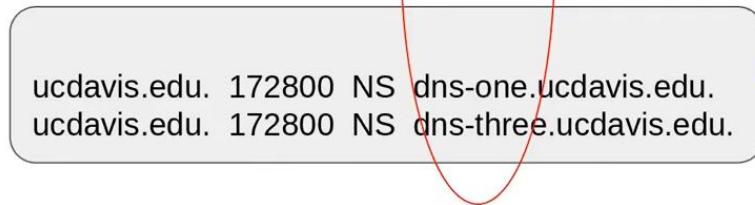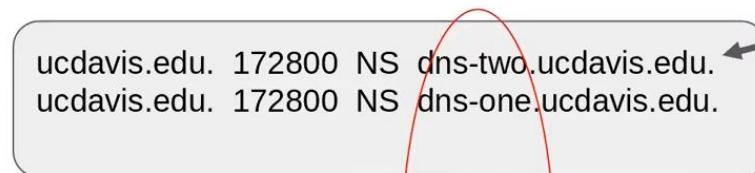
The result is timeouts and delays for clients

When you add Security records EVEN MORE THINGS MUST MATCH

.edu zone

root

edu

com

example

www

ucdavis

dns-one

dns-two

dns-three

.ucdavis.edu zone

34

DNS Threats and Need for DNSSEC

According to the International Data Corporation's (IDC), a European Security Service, Global DNS Threat Reports, the DNS attacks are continuously on the rise and impact major organizations and businesses.
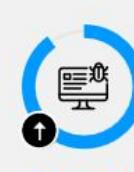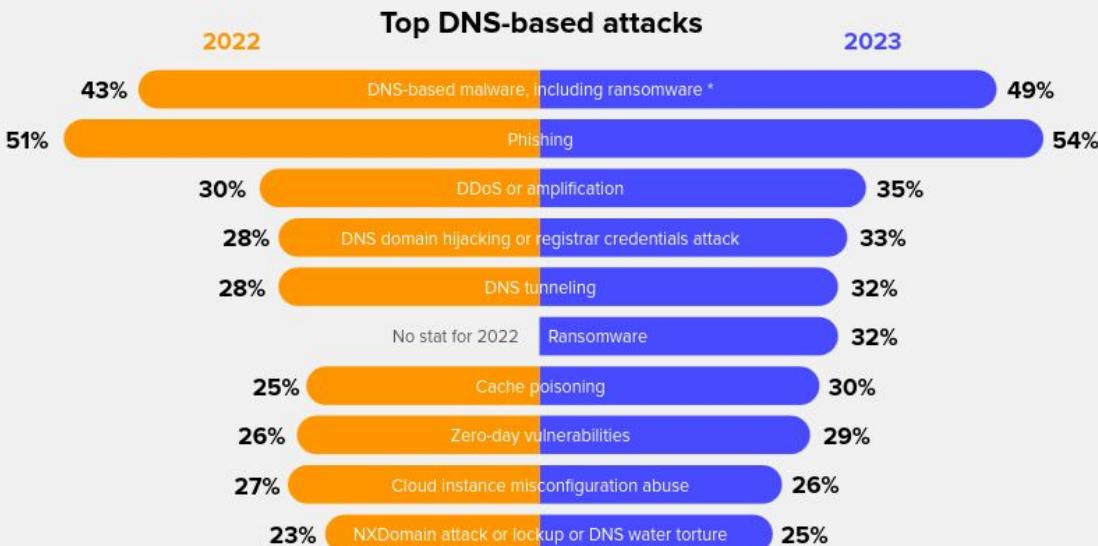
[1] IDC 2021 Global DNS Threat Report. https://efficientip.com/resources/idc-dns-threat-report-2021/
[2] IDC 2022 Global DNS Threat Report. https://efficientip.com/resources/idc-dns-threat-report-2022/
[3] IDC 2023 Global DNS Threat Report.
https://efficientip.com/wp-content/uploads/2023/09/IDC-2023-DNS-Threat-Report.pdf
[4] https://www.idc.com/getdoc.jsp?containerId=IDC_P44612

Threat Landscape

Considering the essential role of DNS in the functioning of the Internet and the enterprise network, as well as its potential as a vector for the most dangerous cyberattacks, organizations need to prioritize DNS security as part of their overall cybersecurity strategy.

Top DNS-based attacks

| Attack	2022	2023
DNS-based malware, including ransomware *	43%	49%
Phishing	51%	54%
DDoS or amplification	30%	35%
DNS domain hijacking or registrar credentials attack	28%	33%
DNS tunneling	28%	32%
Ransomware	No stat for 2022	32%
Cache poisoning	25%	30%
Zero-day vulnerabilities	26%	29%
Cloud instance misconfiguration abuse	27%	26%
NXDomain attack or lockup or DNS water torture	23%	25%

90%
experienced an attack

7.5
attacks on average

Attack sizes remain high

51%
over 5Gb/s

Ransomware:

A trend in ransomware is the use of multi-extortion: Attackers not only encrypt the data but also threaten to leak sensitive information if the ransom is not paid. This tactic is particularly effective against organizations that store large amounts of sensitive data, as the potential damage from a data breach can be much greater than the cost of paying the ransom.

The expansion of the threat landscape and the increasing sophistication of cyberattacks mean organizations need greater visibility and control over network activity. A purpose-built DNS security solution will enhance the security posture and help prevent data breaches and other cyberthreats, while actionable DNS data can be leveraged for threat intelligence.

37

Other Attacks…

Rogue DHCP Server.   DNS Cache Poisoning. DNS Hijacking/Redirection Attacks. DNS Tunneling.   DNS Hijacking - Framing/Phishing.   Subdomain Hijacking.

Domain Squatting. C2 Domains.   Botnet Domains.   Grayware.

Wildcard DNS. Dangling DNS.   Command and Control.

Data Theft.   CNAME Cloaking.   Strategically Aged Domains. DNS Infiltration.

Direct DNS DoS Attack.   DNS Amplification.   DNS Reflection Attack. Bogus domain Attack.

Sloth Domain Attack. Phantom Domain Attack. Pseudo Random Subdomain Attack.

Cache Poisoning

www.example.com?

Resolver

Cache Poisoning

www.example.com?

Resolver

Many

```
root
├── com
│   └── example
│       └── www
└── org
    ├── ietf
    └── icann
        └── ns
```

Cache Poisoning

www.example.com?

Resolver

Many
Many
Requests

root

com

org

example

ietf

icann

www

ns

</>

Cache Poisoning

www.example.com?

Resolver

www.example.com
is at 6.6.6.6!

</>

Many
Many
Requests

root

com

org

example

ietf

icann

www

ns

Cache Poisoning

www.example.com?

Resolver

www.example.com
is at 6.6.6.6!

Many
Many
Requests

Without security,
First answer wins

root

com

org

example

ietf

icann

www

ns

Cache Poisoning

www.example.com?

Resolver

www.example.com
is at 6.6.6.6!

Many
Many
Requests

Without security,
First answer wins

And the resolver
remembers the answer
for a long time!

root

com

org

example

ietf

icann

www

ns

44

Early and Easiest Cache Poisoning **Attacks**
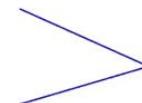
- Extra data in the additional section was considered authentic
  - I know you asked for www.example.com,
  - but did you know www.yourbank.com is at 6.6.6.6?   (The Kashpureff Attack, 1997)
    - It is! And you should trust me!
- Protocol attacks
  - Just **flood the DNS resolver** with response packets and hope they accept them
  - Possibly **guessing IP/protocol values** (name, source address, DNS ID field)   (Amit Klein, 2007)
  - Answer is cached for **TTL chosen by the attacker**

DNS Cache Spoofing (Dan Kaminsky, 2008)
Kaminsky NS Poisoning (<rnd>.isc.org)

com is at ….
    com is at ….
        com is at ….

root

com

Resolver

Early and easiest methods to **combat cache poisoning**

- Ignore non-authoratative answers
  - Only accept answers to questions you asked
  - Exception: parents can supply "glue" address records for in-zone child NS servers
- Resolvers must check that:
  - The IP source and UDP port is correct (handled by the UDP stack)
  - The DNS ID field is correct
- Senders must make it harder for attackers to guess these
  - Randomize the source port number
  - Randomize the ID field
- Note that this
  - Isn't cryptographically strong
  - Doesn't work at all for on-path attackers that can see and copy the requests

32 bits of randomness

Can we fix it?

Yes, we can!

DNS Publication Architecture

Domain Registration Area

Publication Area

DNS Publication Architecture

Domain Registration Area

Publication Area

DNS Zone Content Starts
Here

Zone Owner
Registrant
(ietf.org)

DNS Publication Architecture

Domain Registration Area

Publication Area

Registrars
(Network Solutions)

**DNS Zone Content Starts
Here**

Zone Owner
Registrant
(ietf.org)

DNS Publication Architecture

Domain Registration Area

Publication Area

Registries
(PIR: .org)

Registrars
(Network Solutions)

DNS Zone Content Starts
Here

Zone Owner
Registrant
(ietf.org)

DNS Publication Architecture

Registries
(PIR: .org)

Data Served By
Authoritative DNS Servers
(ns0.amsl.com)

Registrars
(Network Solutions)

**DNS Zone Content Starts
Here**

Zone Owner
Registrant
(ietf.org)

52

DNS Publication Architecture

Domain Registration Area

Publication Area

Registries
(PIR: .org)

Data Served By
Authoritative DNS Servers
(ns0.amsl.com)

Registrars
(Network Solutions)

Queried by Recursive Resolvers
(ISPs, Quads, etc)

**DNS Zone Content Starts
Here**

Zone Owner
Registrant
(ietf.org)

53

DNS Publication Architecture

Registries
(PIR: .org)

Data Served By
Authoritative DNS Servers
(ns0.amsl.com)

Registrars
(Network Solutions)

Queried by Recursive Resolvers
(ISPs, Quads, etc)

**DNS Zone Content Starts
Here**

DNS Users
and applications

Zone Owner
Registrant
(ietf.org)

54

DNS Publication Architecture

DNSSEC/DoT/DoH/DoQ specs HERE
Today's primary conversation

Domain Registration Area

Publication Area

Registries
(PIR: .org)

Data Served By
Authoritative DNS Servers
(ns0.amsl.com)

Registrars
(Network Solutions)

Queried by Recursive Resolvers
(ISPs, Quads, etc)

**DNS Zone Content Starts
Here**

DNS Users
and applications

Zone Owner
Registrant
(ietf.org)

DNS Publication Architecture

DNSSEC/DoT/DoH/DoQ specs HERE
Today's primary conversation

Domain Registration Area

Publication Area

Registries
(PIR: .org)

Data Served By
Authoritative DNS Servers
(ns0.amsl.com)

Registrars
(Network Solutions)

Queried by Recursive Resolvers
(ISPs, Quads, etc)

DNS Zone Content Starts Here

DNS Users
and applications

Zone Owner
Registrant
(ietf.org)

56

DNS Publication Architecture

DNSSEC/DoT/DoH/DoQ specs HERE
Today's primary conversation

Domain Registration Area

Publication Area

Registries
(PIR: .org)

Data Served By
Authoritative DNS Servers
(ns0.amsl.com)

Registrars
(Network Solutions)

Queried by Recursive Resolvers
(ISPs, Quads, etc)

**DNS Zone Content Starts
Here**

DNS Users
and applications

Zone Owner
Registrant
(ietf.org)

DNS Data Protection Mechanisms

Object security

DNS Data Protection Mechanisms

Object security

Path security

Object Security -- DNSSEC

- DNSSEC adds cryptographic **signatures on record data**
- Public/Private keys
- Signed at or near the data's origin
- Verifiable in the middle
- Verifiable at the end
- Only provides integrity protection -- **no privacy protection**

Path Security -- DOT, DOH, DOQ, ...

- Tunnel's answers securely between two points
- Provides integrity protection and encryption
  - But, offers **point-to-point protection only**
  - Verifies **who you got it from**, but **not what it is**

Path vs Object Differences

Object security *(DNSSEC)*

- Pro: Ensures end-to-end integrity is available everywhere
- Con: **Doesn't provide privacy protection**
- Pro: Distributed trust model with minimal configuration (typically 1 trust anchor)
- Con: Typically not deployed all the way to the user ("the last mile")

Path security *(DoT, DOH, …)*

- Pro: Provides point-to-point integrity **and privacy** protection
- Con: Doesn't verify data actually came from the origin *(trust everyone in the path?)*
- Con: For true security, **requires that every link be protected & trusted**
- Pro: Solves the last mile problem

Object vs Path Security

They have very different <u>complementary</u> properties

Object vs Path Security

They have very different <u>complementary</u> properties

Object ~~vs~~ and Path Security Compatibility

They have very different <u>complementary</u> properties

Path and Object security

Stronger together:

Object security protects the data

Object security ties the date to its source

Path security fixes the last mile problem

Path security provides encryption

Protects clients from on-path eavesdropping
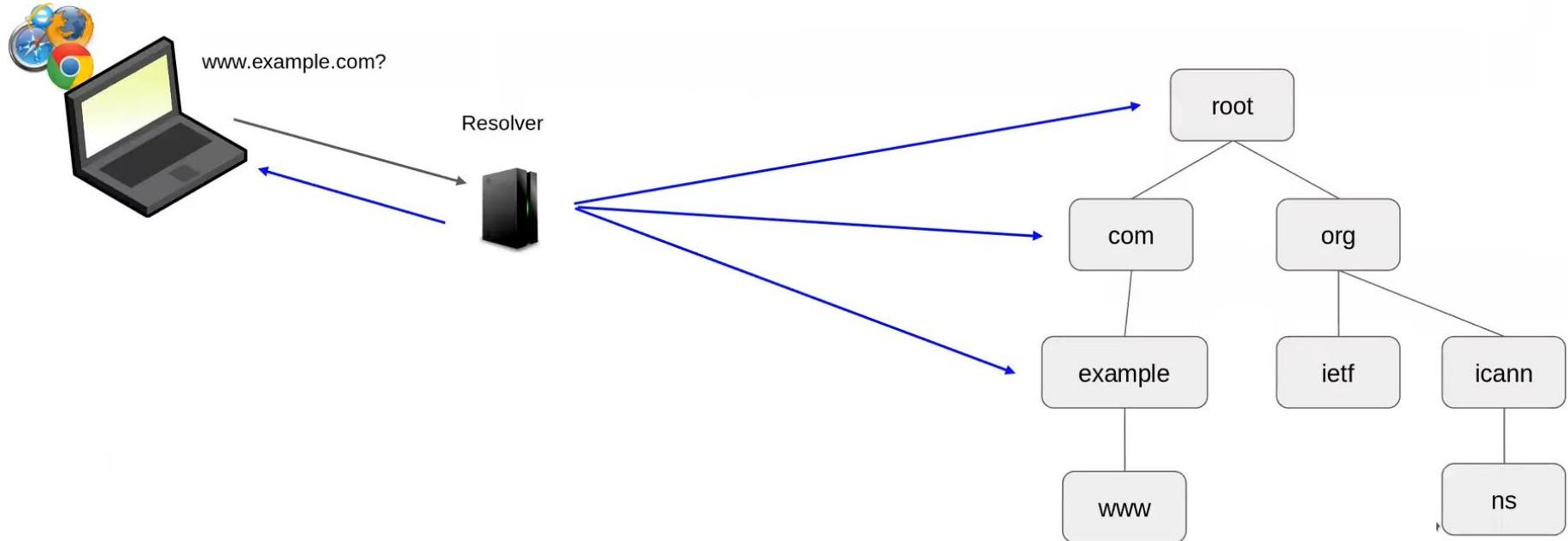
DNS Security Technologies

Object security   (integrity, but no encryption)

- DNSSEC   DNS records signed by their authoritative source   RFC4033+
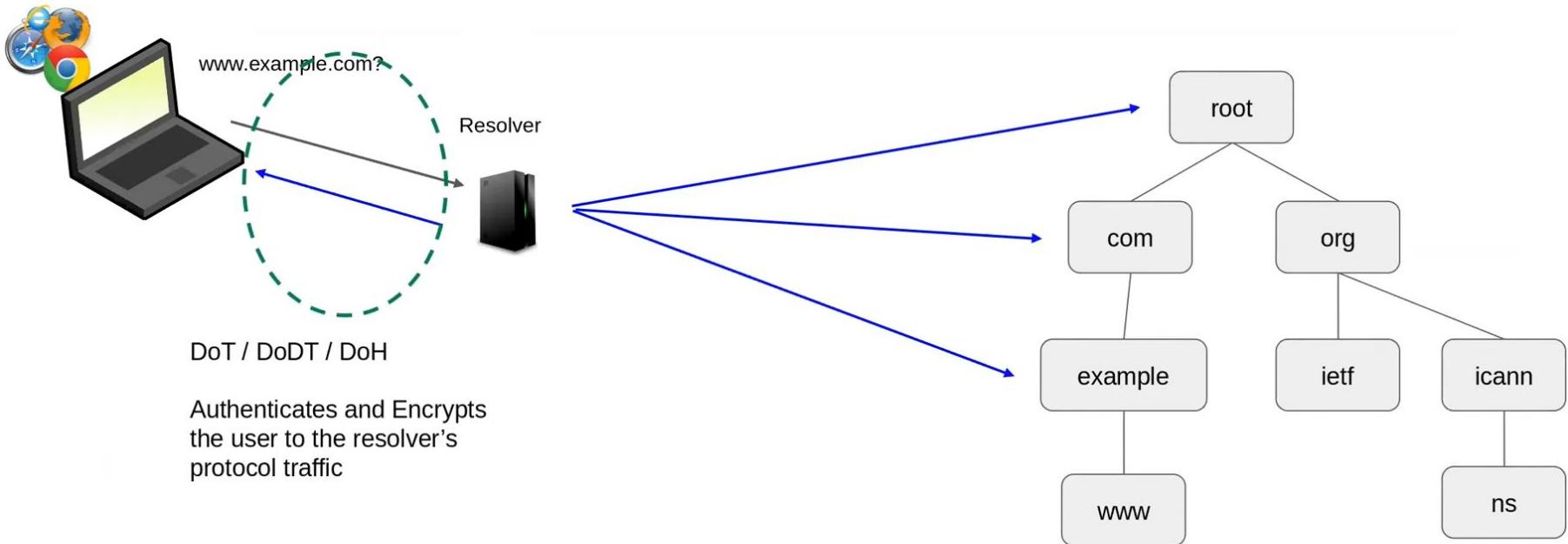
Path security   (point-to-point only integrity, and provides encryption)

- DoT   DNS over TLS   RFC7858
- DoDT   DNS over DTLS   RFC8094
- DoH   DNS over HTTPS   RFC8484
- TSIG   DNS over DNS with shared keys   RFC8945
- DNS Curve: point-to-point encryption and authentication with elliptic curve
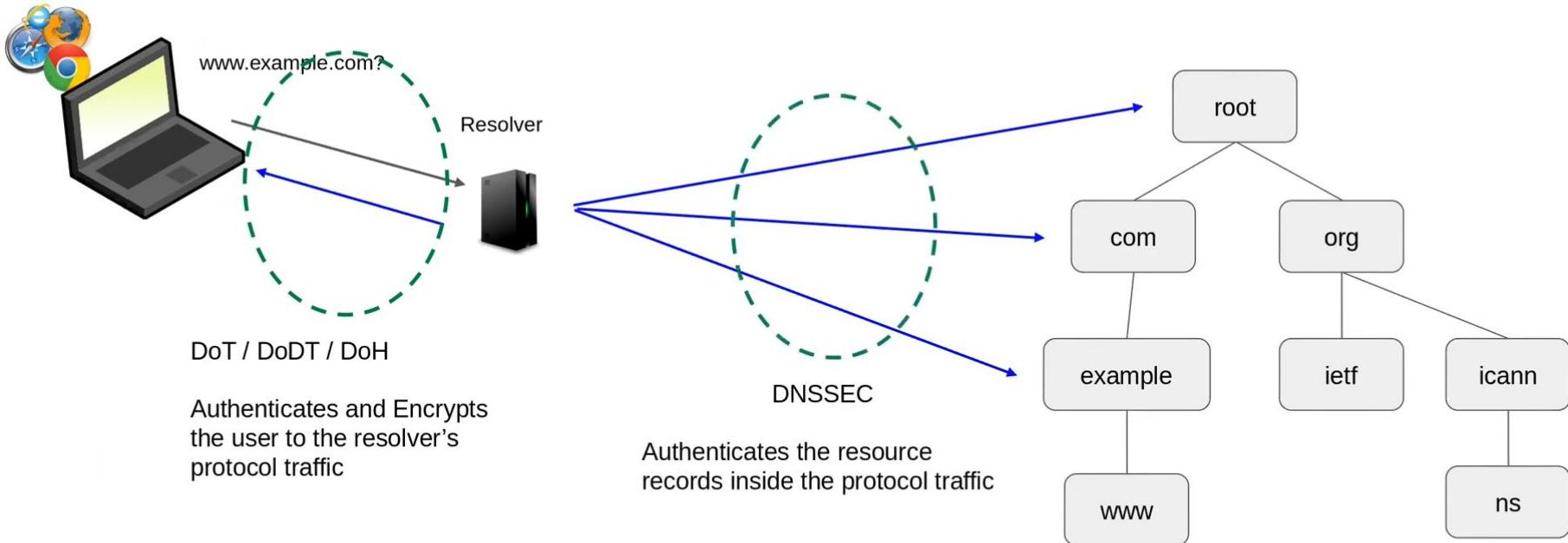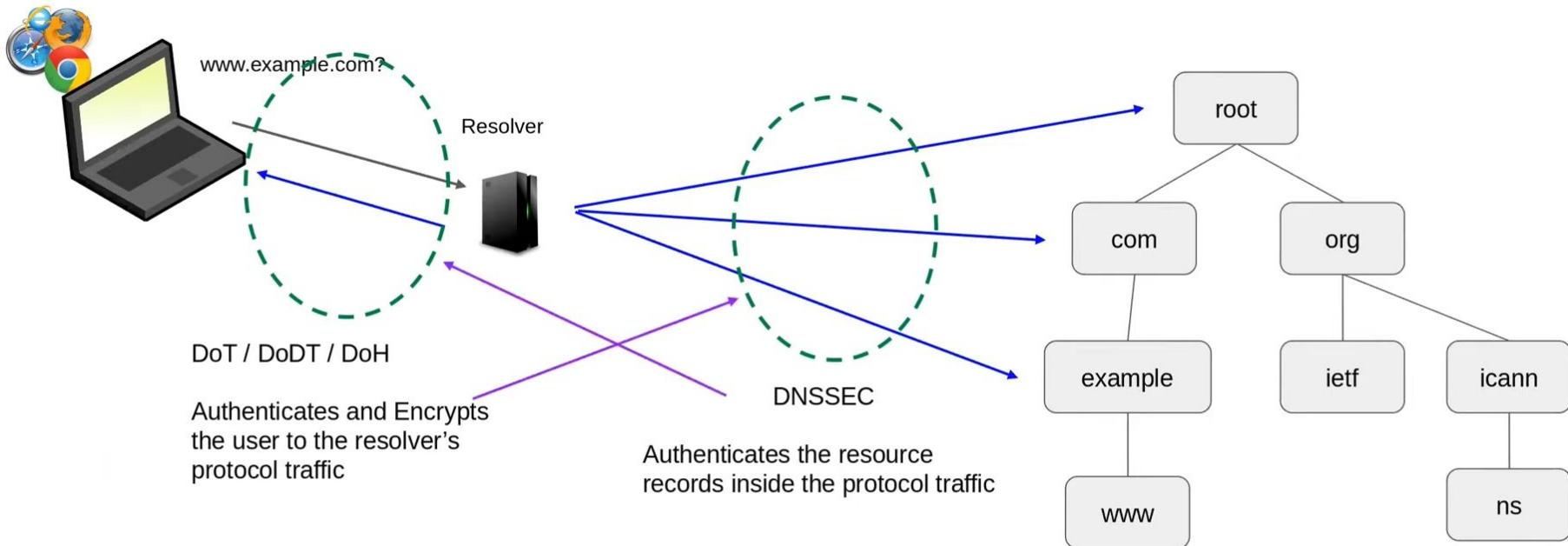
Security Protection Areas

Security Protection Areas

www.example.com?

Resolver

DoT / DoDT / DoH

Authenticates and Encrypts
the user to the resolver's
protocol traffic

root

com   org

example   ietf   icann

www   ns

Security Protection Areas

www.example.com?

Resolver

root

com

org

example

ietf

icann

www

ns

DoT / DoDT / DoH

Authenticates and Encrypts
the user to the resolver's
protocol traffic

DNSSEC

Authenticates the resource
records inside the protocol traffic

Security Protection Areas

www.example.com?

Resolver

DoT / DoDT / DoH

Authenticates and Encrypts
the user to the resolver's
protocol traffic

DNSSEC

Authenticates the resource
records inside the protocol traffic

root

com   org

example   ietf   icann

www   ns

Can be used in both places,
but typically isn't

DNSSEC: Object Security at Work

DNSSEC secures records no matter where they are

93.184.216.34!

Home Router

www.example.com?

www.example.com?

DNS Root (x13 v4, 13 v6)

ISP DNS1

com (x13 v4, 13 v6)

www.example.com?

ISP DNS2

www.example.com?

CLOUD DNS

example.com (x2 v4, 2 v6)

DNSSEC secures records no matter where they are

93.184.216.34!

DNS Root (x13 v4, 13 v6)

Home Router

www.example.com?

www.example.com?

ISP DNS1

com (x13 v4, 13 v6)

www.example.com?

ISP DNS2

www.example.com?

CLOUD DNS

example.com (x2 v4, 2 v6)

DNSSEC secures records no matter where they are

93.184.216.34!

Home Router

www.example.com?

DNS Root (x13 v4, 13 v6)

www.example.com?

ISP

com (x13 v4, 13 v6)

www.example.com?

ISP DNS2

www.example.com?

CLOUD DNS

example.com (x2 v4, 2 v6)

DNSSEC secures records no matter where they are

DNS Root (x13 v4, 13 v6)

93.184.216.34!

Home Router

www.example.com?

www.example.com?

com (x13 v4, 13 v6)

www.example.com?

ISP DNS2

www.example.com?

CLOUD DNS

www.example.com?

example.com (x2 v4, 2 v6)

DNSSEC secures records no matter where they are

93.184.216.34!

www.example.com?

Home Router

DNS Root (x13 v4, 13 v6)

www.example.com?

www.example.com?

ISP

ISP DNS2

com (x13 v4, 13 v6)

www.example.com?

CLOUD DNS

www.example.com?

example.com (x2 v4, 2 v6)

DNSSEC secures records no matter where they are

DNS Root (x13 v4, 13 v6)

93.184.216.34!

Home Router

www.example.com?

www.example.com?

www.example.com?

com (x13 v4, 13 v6)

ISP

ISP DNS2

www.example.com?

CLOUD DNS

www.example.com?

example.com (x2 v4, 2 v6)

DNSSEC – hierarchical object signing security – "chain of trust"

www.example.com?

Resolver

public/private
Key pairs

root

com   edu

example   isi   ucdavis

www   www

DNSSEC – hierarchical object signing security – "chain of trust"

www.example.com?

Resolver

public/private
Key pairs

root

com

edu

example

isi

ucdavis

www

www

DNSSEC – hierarchical object signing security – "chain of trust"

www.example.com?

Resolver

public/private
Key pairs

root

com   edu

example   isi   ucdavis

www   www

Signs **root**
Signs edu's DS

DNSKEY
DS   RRSIG

DNSSEC – hierarchical object signing security – "chain of trust"

www.example.com?

Resolver

public/private
Key pairs

root

com

edu

example

isi

ucdavis

www

www

Signs **root**
Signs edu's DS

DNSKEY
DS

RRSIG

Signs **edu**
Signs ucdavis.edu's DS

DNSKEY
DS

RRSIG

DNSSEC – hierarchical object signing security – "chain of trust"

www.example.com?

Resolver

public/private
Key pairs

root

com   edu

example   isi   ucdavis

www   www

Signs **root**
Signs edu's DS

DNSKEY
DS
RRSIG

Signs **edu**
Signs ucdavis.edu's DS

DNSKEY
DS
RRSIG

Signs **ucdavis.edu**
Signs www.ucdavis.edu

DNSKEY
A
RRSIG

DNSSEC – hierarchical object signing security – "chain of trust"

www.example.com?

Resolver

AD=1
All good!

public/private
Key pairs

root

com   edu

example   isi   ucdavis

www   www

Signs **root**
Signs edu's DS

DNSKEY → RRSIG
DS

Signs **edu**
Signs ucdavis.edu's DS

DNSKEY → RRSIG
DS

Signs **ucdavis.edu**
Signs www.ucdavis.edu

DNSKEY → RRSIG
A

DNSSEC Component Records   *(we use the DNS to secure the DNS)*

- **DNSKEY:**   The public key used to verify a zone's signatures

- **DS:**   A cryptographic hash of a child's *DNSKEY* for the parent to publish

- **CDS:**   Key rolling record for parents to query in children

- **RRSIG:**   A signature record that signs <CLASS, QTYPE, QNAME, CLASS>

  - Example: One RRSIG exists for all three *ucdavis.edu* NS records: *<IN, NS, ucdavis.edu>* but another exists for the single *<IN, A, ucdavis.edu>* IPv4 record

- **NSEC:**   Proves other names don't exist between two names

  - NSEC apple orange: banana can't exist

- **NSEC3:**   A hashing version of NSEC – use only if you need it (hint: you don't)

Real World Views of ~~DNSSEC~~

```
# dig ucdavis.edu NS

;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 6

;; ANSWER SECTION:
ucdavis.edu.                    9985    IN NS dns-three.ucdavis.edu.
ucdavis.edu.                    9985    IN NS dns-one.ucdavis.edu.
ucdavis.edu.                    9985    IN NS dns-two.ucdavis.edu.

;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN A        128.120.252.10
dns-one.ucdavis.edu.    9985    IN A        128.120.252.9
dns-three.ucdavis.edu.  9985    IN A        169.237.243.171
dns-two.ucdavis.edu.    9986    IN AAAA     2607:f810:3f0:2::2
dns-three.ucdavis.edu.  9987    IN AAAA     2607:f810:ce0:10::2
```

Real World Views of ~~DNSSEC~~

Your resolver is DNSSEC validating (even if you didn't ask)

```
# dig ucdavis.edu NS

;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 6

;; ANSWER SECTION:
ucdavis.edu.              9985    IN NS dns-three.ucdavis.edu.
ucdavis.edu.              9985    IN NS dns-one.ucdavis.edu.
ucdavis.edu.              9985    IN NS dns-two.ucdavis.edu.

;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN A       128.120.252.10
dns-one.ucdavis.edu.    9985    IN A       128.120.252.9
dns-three.ucdavis.edu.  9985    IN A       169.237.243.171
dns-two.ucdavis.edu.    9986    IN AAAA    2607:f810:3f0:2::2
dns-three.ucdavis.edu.  9987    IN AAAA    2607:f810:ce0:10::2
```

Real World Views of DNSSEC

Your resolver is DNSSEC validating (even if you didn't ask)

Seconds left in the resolver's cache before it asks again

```
# dig ucdavis.edu NS

;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 6

;; ANSWER SECTION:
ucdavis.edu.          9985    IN NS dns-three.ucdavis.edu.
ucdavis.edu.          9985    IN NS dns-one.ucdavis.edu.
ucdavis.edu.          9985    IN NS dns-two.ucdavis.edu.

;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986   IN A      128.120.252.10
dns-one.ucdavis.edu.    9985   IN A      128.120.252.9
dns-three.ucdavis.edu.  9985   IN A      169.237.243.171
dns-two.ucdavis.edu.    9986   IN AAAA   2607:f810:3f0:2::2
dns-three.ucdavis.edu.  9987   IN AAAA   2607:f810:ce0:10::2
```

88

Real World Views of DNSSEC

```
# dig +dnssec ucdavis.edu NS

;; ANSWER SECTION:
ucdavis.edu.        9936    IN NS dns-three.ucdavis.edu.
ucdavis.edu.        9936    IN NS dns-one.ucdavis.edu.
ucdavis.edu.        9936    IN NS dns-two.ucdavis.edu.
ucdavis.edu.        9936    IN RRSIG NS 8 2 14400 20230106061807 20220106051807
                            40986 ucdavis.edu. KitFCqJ28ppQjZB............hK S+4=


;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN A        128.120.252.10
dns-two.ucdavis.edu.    8409    IN RRSIG A 8 3 14400 20230106061807 20220106051807
                            40986 ucdavis.edu. FOqxEMGl4h..............+to lh8=
```

(other names/AAAA records truncated for brevity)

Real World Views of DNSSEC

A signature on the NS record **SET**

```
# dig +dnssec ucdavis.edu NS

;; ANSWER SECTION:
ucdavis.edu.      9936    IN NS dns-three.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-one.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-two.ucdavis.edu.
ucdavis.edu.      9936    IN RRSIG NS 8 2 14400 20230106061807 20220106051807
                          40986 ucdavis.edu. KitFCqJ28ppQjZB...........hK S+4=


;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN A        128.120.252.10
dns-two.ucdavis.edu.    8409    IN RRSIG A 8 3 14400 20230106061807 20220106051807
                          40986 ucdavis.edu. FOqxEMGl4h............+to lh8=


(other names/AAAA records truncated for brevity)
```

Real World Views of DNSSEC

A signature on the NS record **SET**

```
# dig +dnssec ucdavis.edu NS

;; ANSWER SECTION:
ucdavis.edu.      9936    IN NS dns-three.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-one.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-two.ucdavis.edu.
ucdavis.edu.      9936    IN RRSIG NS 8 2 14400 20230106061807 20220106051807
                          40986 ucdavis.edu. KitFCqJ28ppQjZB............hK S+4=
```

Key lookup hint

```
;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN A        128.120.252.10
dns-two.ucdavis.edu.    8409    IN RRSIG A 8 3 14400 20230106061807 20220106051807
                                40986 ucdavis.edu. FOqxEMGl4h..............+to lh8=
```

(other names/AAAA records truncated for brevity)

91

Real World Views of DNSSEC

A signature on the NS record **SET**

```
# dig +dnssec ucdavis.edu NS

;; ANSWER SECTION:
ucdavis.edu.      9936    IN NS dns-three.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-one.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-two.ucdavis.edu.
ucdavis.edu.      9936    IN RRSIG NS 8 2 14400 20230106061807 20220106051807
                          40986 ucdavis.edu. KitFCqJ28ppQjZB...........hK S+4=
```

Key lookup <u>hint</u>

Signing algorithm

```
;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN A        128.120.252.10
dns-two.ucdavis.edu.    8409    IN RRSIG A 8 3 14400 20230106061807 20220106051807
                                40986 ucdavis.edu. FOqxEMGl4h..............+to lh8=

(other names/AAAA records truncated for brevity)
```

Real World Views of DNSSEC

A signature on the NS record **SET**

DNSSEC Signature End Date

```
# dig +dnssec ucdavis.edu NS

;; ANSWER SECTION:
ucdavis.edu.      9936    IN NS dns-three.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-one.ucdavis.edu.
ucdavis.edu.      9936    IN NS dns-two.ucdavis.edu.
ucdavis.edu.      9936    IN RRSIG NS 8 2 14400 20230106061807 20220106051807
                          40986 ucdavis.edu. KitFCqJ28ppQjZB...........hK S+4=
```

Key lookup hint

Signing algorithm

```
;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.   9986    IN A       128.120.252.10
dns-two.ucdavis.edu.   8409    IN RRSIG A 8 3 14400 20230106061807 20220106051807
                               40986 ucdavis.edu. FOqxEMGl4h..............+to lh8=

(other names/AAAA records truncated for brevity)
```

Real World Views of DNSSEC

A signature on the NS record **SET**

DNSSEC Signature End Date

DNSSEC Signature Begin Date

```
# dig +dnssec ucdavis.edu NS

;; ANSWER SECTION:
ucdavis.edu.       9936    IN NS dns-three.ucdavis.edu.
ucdavis.edu.       9936    IN NS dns-one.ucdavis.edu.
ucdavis.edu.       9936    IN NS dns-two.ucdavis.edu.
ucdavis.edu.       9936    IN RRSIG NS 8 2 14400 20230106061807 20220106051807
                          40986 ucdavis.edu. KitFCqJ28ppQjZB............hK S+4=
```

Key lookup hint

Signing algorithm

```
;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN A        128.120.252.10
dns-two.ucdavis.edu.    8409    IN RRSIG A 8 3 14400 20230106061807 20220106051807
                                40986 ucdavis.edu. FOqxEMGl4h.............+to lh8=
```

(other names/AAAA records truncated for brevity)

Real World Views of DNSSEC

A signature on the NS record **SET**

DNSSEC Signature End Date

DNSSEC Signature Begin Date

```
# dig +dnssec ucdavis.edu NS

;; ANSWER SECTION:
ucdavis.edu.      9936    IN  NS  dns-three.ucdavis.edu.
ucdavis.edu.      9936    IN  NS  dns-one.ucdavis.edu.
ucdavis.edu.      9936    IN  NS  dns-two.ucdavis.edu.
ucdavis.edu.      9936    IN  RRSIG  NS 8 2 14400 20230106061807 20220106051807
                  40986 ucdavis.edu. KitFCqJ28ppQjZB............hK S+4=
```

Key lookup hint

Signing algorithm

Cryptography Working For You

```
;; ADDITIONAL SECTION:
dns-two.ucdavis.edu.    9986    IN  A       128.120.252.10
dns-two.ucdavis.edu.    8409    IN  RRSIG  A 8 3 14400 20230106061807 20220106051807
                        40986 ucdavis.edu. FOqxEMGl4h...............+to lh8=

(other names/AAAA records truncated for brevity)
```

Real World Views of DNSSEC / DNSKEYs

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.      79818  IN DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.      79818  IN DNSKEY 256 3 8 AwEAAc07f……uEUD/7
```

```
ZSK (256): signs ALL records
           "zone signing key"
KSK (257): signs only the keys themselves
           "key signing key"
           DS records are created from this one to give to your parent
```

Real World Views of DNSSEC / DNSKEYs

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.      79818  IN DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.      79818  IN DNSKEY 256 3 8 AwEAAc07f……uEUD/7
```

Type of key

```
ZSK (256): signs ALL records
           "zone signing key"
KSK (257): signs only the keys themselves
           "key signing key"
           DS records are created from this one to give to your parent
```

Why does DNSSEC have 2 key types????

ZSK (256): signs ALL records
         "zone signing key"

         Designed to be rotated frequently if desired
         Requires no parent communication to roll them
         Current software can **entirely automate this**
         If rotated frequently, can generally be shorter in size

KSK (257): signs only the keys themselves
         "key signing key"

         DS records are created from KSKs to give to your parent
         Harder to rotate
         Often longer in size
         Rolling only requires special considerations

Note: you can have just a ZSK — the DS in the parent should point to the ZSK 98

Real World Views of DNSSEC / DNSKEYs → DS at a parent

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.      79818  IN DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.      79818  IN DNSKEY 256 3 8 AwEAAc07f……uEUD/7



# dig @k.edu-servers.net. ucdavis.edu DS
ucdavis.edu.      86400  IN DS 31642 8 2 2F9EC3E5………48CB394
ucdavis.edu.      86400  IN DS 31642 8 1 71F9C0…6E8FCB



SHA256 (2):     the algorithm you really MUST have
SHA1   (1):     stop using this at this point — no longer needed and less secure
```

Real World Views of DNSSEC / DNSKEYs → DS at a parent

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.      79818  IN  DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.      79818  IN  DNSKEY 256 3 8 AwEAAc07f……uEUD/7
```

Key lookup hint

```
# dig @k.edu-servers.net. ucdavis.edu DS
ucdavis.edu.      86400  IN  DS  31642  8 2 2F9EC3E5………48CB394
ucdavis.edu.      86400  IN  DS  31642  8 1 71F9C0…6E8FCB
```

```
SHA256 (2):    the algorithm you really MUST have
SHA1   (1):    stop using this at this point – no longer needed and less secure
```

Real World Views of DNSSEC / DNSKEYs → DS at a parent

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.      79818  IN DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.      79818  IN DNSKEY 256 3 8 AwEAAc07f……uEUD/7
```

Key lookup <u>hint</u>

```
# dig @k.edu-servers.net. ucdavis.edu DS
ucdavis.edu.      86400  IN DS 31642 8 2 2F9EC3E5………48CB394
ucdavis.edu.      86400  IN DS 31642 8 1 71F9C0…6E8FCB
```

Hashing type

```
SHA256 (2):    the algorithm you really MUST have
SHA1   (1):    stop using this at this point — no longer needed and less secure
```

One DS → DNSKEY match is always REQUIRED

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.       79818   IN DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.       79818   IN DNSKEY 256 3 8 AwEAAc07f……uEUD/7


# dig @k.edu-servers.net. ucdavis.edu DS
ucdavis.edu.       86400   IN DS 31642 8 2 2F9EC3E5………48CB394
```

One DS → DNSKEY match is always REQUIRED

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.     79818  IN DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.     79818  IN DNSKEY 256 3 8 AwEAAc07f……uEUD/7


# dig @k.edu-servers.net. ucdavis.edu DS
ucdavis.edu.     86400  IN DS 31642 8 2 2F9EC3E5………48CB394
```

CRITICAL for validation to succeed:
1. **One DS record in your parent must match one DNSKEY in your zone at all times**

One DS → DNSKEY match is always REQUIRED

```
# dig ucdavis.edu DNSKEY

;; ANSWER SECTION:
ucdavis.edu.      79818  IN  DNSKEY 257 3 8 AwEAAeGSj……rXsHE=
ucdavis.edu.      79818  IN  DNSKEY 256 3 8 AwEAAc07f……uEUD/7


# dig @k.edu-servers.net. ucdavis.edu DS
ucdavis.edu.      86400  IN  DS 31642 8 2 2F9EC3E5………48CB394
```

CRITICAL for validation to succeed:
1. One DS record in your parent must match one DNSKEY in your zone at all times
2. This includes all records potentially in resolver caches – TTLs really really matter

OK, but how does one actually roll keys? *(and what is rolling???)*
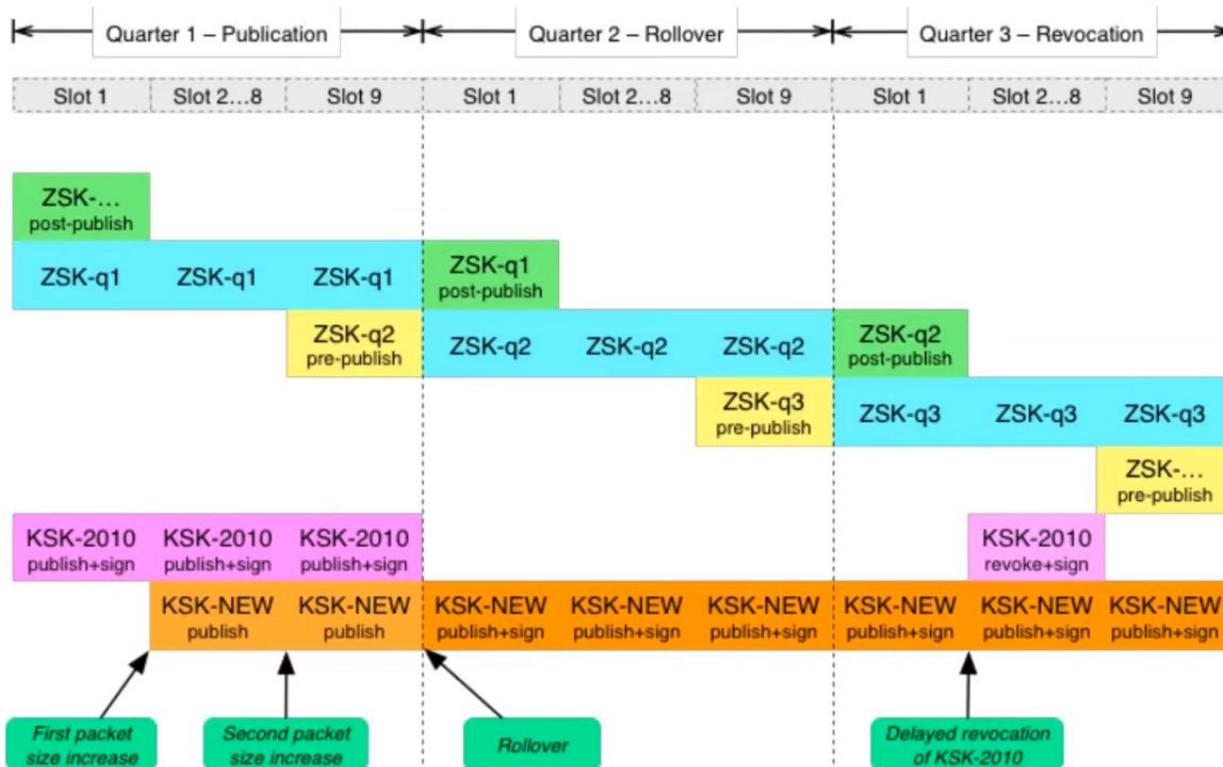

Figure 1. Rollover Scheduling

ICANN's multi-step process for rolling the root zone's KSK

DNSSEC Deployment Recommendations

- ## Use modern tools
  - "back in my day" we had to self-manage keys, signing, rollovers, etc
  - These days excellent tools do all the hard work: infoblox, registratars (google), knot, bind, …

- ## Use modern algorithms
  - DS records: SHA256 only in your parent
  - Use elliptic curve algorithms (eg, algorithm 13)
  - Don't use NSEC3

- ## Monitor monitor monitor
  - Do all your parent/child records match?
  - Are your signatures still good far enough out?

DoT, DoH: Path Security at Work

Zeek (A useful tool)

Zeek — put a pcap in and get a bunch of log files out.

Installation: sudo apt install zeek

Running zeek on a pcap gives a list of files, e.g.,
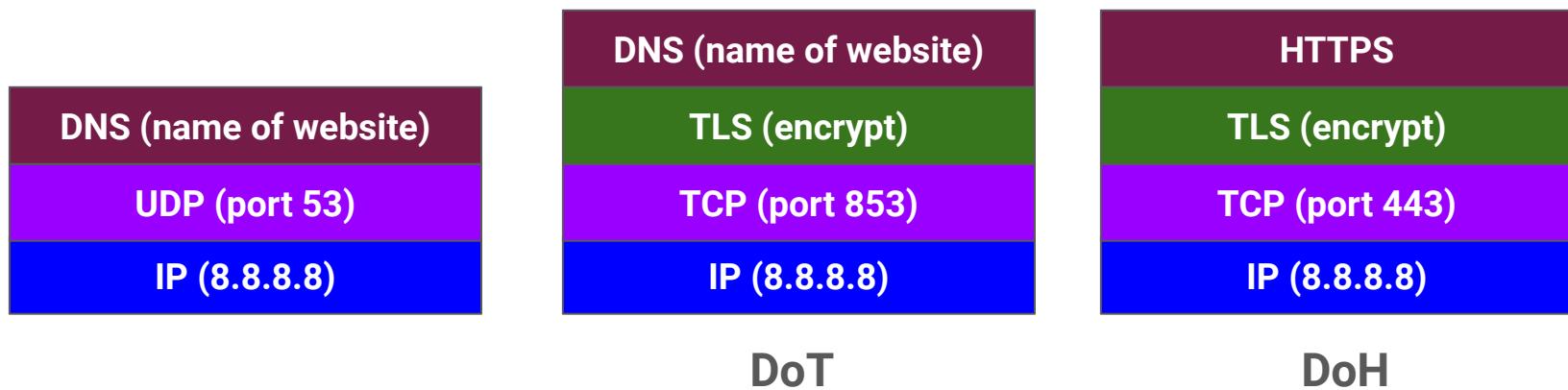
conn.log → contains src/dst ip, src/dst port,

dns.log → contains dns requests

less -S conn.log (displays results in an organized column format)

Exercise: Run zeek on a pcap and analyze the details.

DNS over TLS (DoT) and DNS over HTTP (DoH)

DNS (name of website)
UDP (port 53)
IP (8.8.8.8)

DNS (name of website)
TLS (encrypt)
TCP (port 853)
IP (8.8.8.8)

DoT

HTTPS
TLS (encrypt)
TCP (port 443)
IP (8.8.8.8)

DoH

DNS over HTTP (DoH)

Exercise: Decrypt DNS traffic using DoH.

1. Generate DNS traffic and see that in wireshark.
2. Turn DoH on in Firefox and see how the traffic changes.
3. How to decrypt this traffic?

DoT can be observed in wireshark using the same process with some tweaks.

TLSv1.2 vs TLSv1.3

Difference?

Encrypted Server Name Indication (ESNI)

Issue: people can tell where you are going even if you're using TLS.

Exercise: Find all the SNI values within a capture?

What is Server Name Indication?

Server Name Indication (SNI) is an extension to the Transport Layer Security (TLS) computer networking protocol by which a client indicates which hostname it is attempting to connect to at the start of the handshaking process.

Turn to wireshark to show SNI.

Continued…..
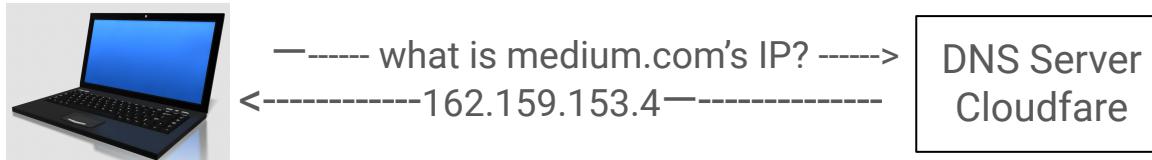
Encrypted Server Name Indication (ESNI) is an extension to TLS 1.3 which prevents eavesdroppers from knowing the domain name of the website network users are connecting to. When combined with encrypted DNS, it is not possible to know which websites a user is visiting.

DoH or DoT should be in place before ESNI.
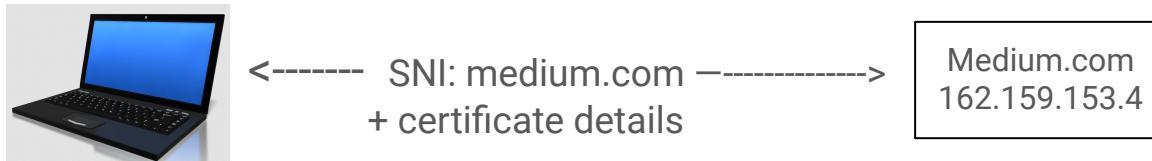
ESNI to Protect TLS

To visit a TLS website:

1. Get IP from DNS server

⸺------ what is medium.com's IP? ------> | DNS Server Cloudfare
<------------162.159.153.4⸺-------------

2. Connect to medium's IP using TLS

<------- SNI: medium.com ⸺-------------> | Medium.com 162.159.153.4
+ certificate details

ESNI to Protect TLS

To visit a TLS website:

When DoH is enabled, we will only see that someone is connecting to the Cloudfare DNS server? But, we don't know what they are asking….

1. Get IP from DNS server

—----- what is ????????????? IP? -----> | DNS Server Cloudfare |
<------------ ????????????? —------------

2. Connect to medium's IP using TLS

<------- SNI: medium.com —-------------> | Medium.com 162.159.153.4 |
+ certificate details
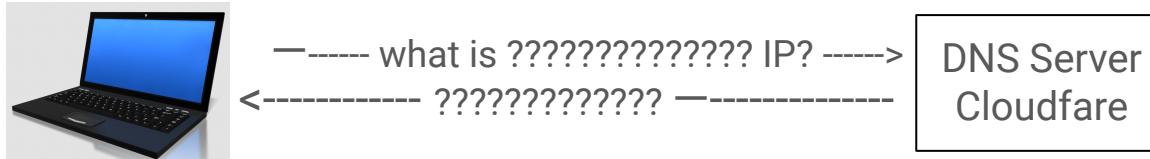
In TLSv1.2, we would see SNI and certificate details with DoH enabled.

ESNI to Protect TLS

To visit a TLS website:

When DoH is enabled, we will only see that someone is connecting to the Cloudfare DNS server? But, we don't know what they are asking....

1. Get IP from DNS server

—------ what is ????????????? IP? ------> DNS Server Cloudfare
<------------ ????????????? —--------------

2. Connect to medium's IP using TLS

<------- SNI: medium.com —--------------> Medium.com 162.159.153.4

In TLSv1.3, the certificate details are encrypted.

116

ESNI to Protect TLS

To visit a TLS website:

When DoH is enabled, we will only see that someone is connecting to the Cloudfare DNS server? But, we don't know what they are asking....

1. Get IP from DNS server

—------ what is ????????????? IP? ------>   DNS Server Cloudfare
<------------ ????????????? —-------------

2. Connect to medium's IP using TLS

<------- ??????????????????? —------------->   Medium.com 162.159.153.4

In TLSv1.3, the ESNI encrypts the SNI.

The websites that support ESNI will have their public key in DNS records so that the clients can use that to encrypt their traffic.

ESNI to date is not heavily supported.

We are going towards intercepting the TLS connection traffic to get information about what someone is connecting to.

Shortcomings of ESNI

Incomplete protection.

For example, during session resumption, the Pre-Shared Key extension could, legally, contain a cleartext copy of exactly the same server name that is encrypted by ESNI.

Real-world use of ESNI has exposed interoperability and deployment challenges that prevented it from being enabled at a wider scale.

https://blog.mozilla.org/security/2021/01/07/encrypted-client-hello-the-future-of-esni-in-firefox/

Encrypted Client Hello (ECH)

Firefox decided to kill ESNI.

But there is a replacement for encrypting the destination of TLS traffic.

The standard that replaced ESNI is called ECH.

Useful Resources:

- 12 days of Defense series (https://www.youtube.com/watch?v=9H8wAXg2vrs)
- Zeek (https://zeek.org/get-zeek/)
- Malware Traffic Analysis.net (https://www.malware-traffic-analysis.net/index.html)
- Wes Hardaker lecture on DNS security (https://www.linkedin.com/posts/whardaker_dns-security-overview-activity-6901910613824565250-XGmL?trk=public_profile_share_view)