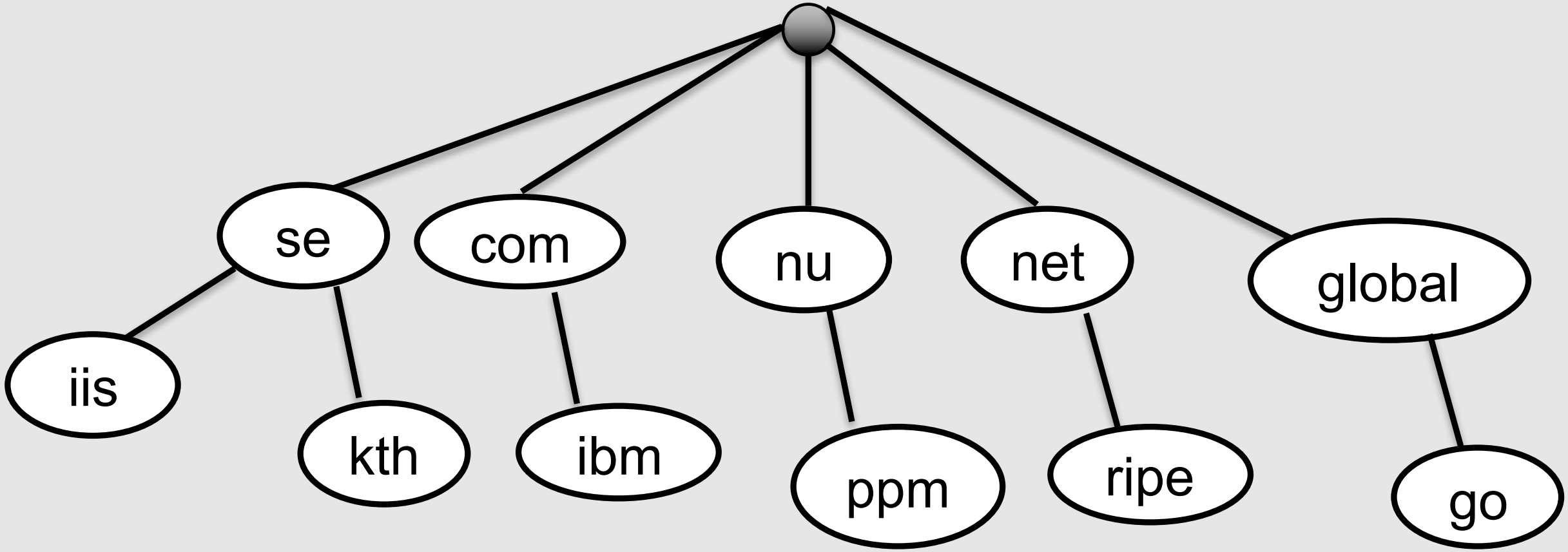

se
com
nu
net
global
iis
kth
ibm
ppm
ripe
go

INTERNET STIFTELSEN

Internets domännamnssystem*

Föreläsning FL11, HT 2023

Mats Dufberg

* Se “Internets domännamnssystem”

Innehåll

▶Tillgänglighet

[Innehåll]

Tillgänglighet

Tillgänglighet till DNS-datat är avgörande för DNS.

- Svarstiden för en DNS-fråga kan tyckas försumbar:
  - En fråga till resolvern kan leda till många frågor för resolvern. Jfr med frågorna för att få fram A-posten för www.dn.se.
  - En webbsidan leder ofta till många anrop och många DNS-frågor.

Här ska vi skilja mellan tillgänglighet till resolver och tillgänglighet till DNS-hosting (zondatat).

▶Resolvertillgänglighet

[Innehåll]

Tillgänglighet till resolver

Om resolvern har dålig tillgänglighet så kommer all DNS-uppslagning att drabbas.

Problem med resolvern kan av användarna uppfattas som nätproblem eller att tjänsterna är dåliga.

Resolverproblem

- Dålig eller lång förbindelse till resolvern
- Dålig kapacitet i resolvern
- Instabil resolver (går ner ibland)
- Resolvern har dålig eller lång förbindelse till allmänna Internet
- Primära resolvern svarar inte (mer nedan)

Cache snabbar upp resolvern

Om svaret redan finns i *chachen* så kan resolvern svara direkt utan lägga tid på att hämta svaret.

- Tillräckligt med minne för cachen är viktigt.
- Google och andra stora operatörer av öppna resolvrar lär ha en mekanism för att förnya cachen för viss data (*prefetching*).

Reservresolver

De flesta OS tillåter att en extra resolver konfigureras i /etc/resolv.conf eller motsvarande.

- Löser det akuta problemet ifall den primära resolvern är nere.
- Ger oftast långa uppslagningstider eftersom datorn först ska göra *timeout* på den primära resolvern.
- Kan vara krångligt att konfigurera *lokalt* ifall datorn får automatiskt tilldelad resolvern tillsamman med nätinställningar (DHCP).

Primär resolver är nere

DHCP kan dela ut flera resolvrar, eller servrar kan konfigureras flera resolvrar i resolv.conf. Den första blir oftast den primära.

Det är inte ett ovanligt fel är att *DHCP* delar ut dubbla resolveradresser för att ge en i reserv, men den primära svarar inte längre. Kanske har den bytt IP-adress eller så är den avvecklad, men konfigureringen har inte uppdaterats.

- Detta kommer att uppfattas som nätproblem eller att tjänsterna är tröga.
- Vid nätproblem, testa att göra DNS-uppslagning mot adresserna i resolv.conf.

▶Tillgänglighet till zondatat

[Innehåll]

Tillgänglighet till zondatat

Dålig tillgänglighet till en viss domän eller grupp av domäner påverkar framförallt dessa, men kan indirekt påverka andra domäner genom beroenden t.ex. genom CNAME.

Dålig tillgänglighet till rotzonen kan ge allmänna problem med DNS.

Det går att ladda en lokal kopia av rotzonen via resolvern för att ge snabbare tillgång till rotzonen, se RFC 8806 (överkurs).

https://datatracker.ietf.org/doc/html/rfc8806

Tillgänglighet till zondatat

Det finns flera faktorer som avgör hur tillgänglig DNS-datat för en viss domän (i en viss zon) är.

- Serverkapcitet
- TTL
- Serverredundans
- Nättillgänglighet
- Nätredundans

Serverkapacitet

För stora zoner med mycket trafik (många frågor) eller namnserver med många zoner med sammanlagt mycket trafik så kommer serverkapaciteten att vara avgörande för snabba svar.

För en server med en liten zon med få frågor så är serverkapaciteten av mindre betydelse. DNS är inte tungt, men mängden kan tynga.

TTL

Om en namnserver är hårt lastad av DNS-frågor så kan det i vissa fall avhjälpas genom översyn av vald TTL.

Om TTL är kort och TTL fördubblas för alla populär DNS-poster så kan lasten halveras.

- Slentrianmässigt låg TTL kan leda till onödig last.
- Längre TTL kan ge bättre användarupplevelse.

Sätt låg TTL i samband med uppdateringar, men annars högre TTL.

Serverredundans

Ju fler namnservrar en zon hanteras av, desto rubustare blir dess hosting.

- Varje NS-post ska vara en unik server.
  - IPv4 resp IPv6 kan servas av olika eller samma servrar.
  - En IPv4-adress och/eller en IPv6-adress per NS.
- Två NS ska absolut inte gå till samma IP-adress.

Håll delegeringen korrekt

Delegeringen är startpunkten för att nå domänen. Håll den uppdaterad och korrekt.

- Se över delegeringen varje gång som det görs ändringar på namnservrarna för zonen.
- Samma NS i delegering som i zonen. Det är framförallt delegeringen som styr. Uppdatera alltid delegeringen så att den stämmer med zonen.
  - NS som bara finns i zonen används om den kommer med i *authority section*. Det beror på namnserverns konfiguration.
- Se till att glue-posterna i delegeringen har korrekta IP-adresser.
  - Om namnserverna har IPv6-adress så ska den finnas med som glue i delegeringen.

Serverredundans

- Servrar som inte svarar förlänger svarstiden, t.ex. ouppdaterad glue-post.
- Överlastade servrar förlänger svarstiden.
- Slavar som inte uppdateras ger inkonsistent data.

Om antalet NS är tillräckligt så är det bättre att plocka bort dåliga slavar än att behålla en slav som alltid har uppdateringsproblem.

Gör gärna mastern dold för att den primärt ska mata slavarna med uppdaterad zon och inte lastas med DNS-frågor.

Serverredundans

- 2 NS är minimum. 1 NS kan vara OK för en testdomän.
  - Gäller per IPv4/IPv6.
- Om en namnserver har IPv6 så ska minst 2 ha IPv6 för att ge redundans. Helst alla NS.
- 3-5 NS är ett bra riktmål beroende på frågelast och hur viktig domänen är.
  - Vissa TLD:er har krav på minst två NS i delegeringen (vanligare förr).

Nättillgänglighet

Resolvrar kommer i viss utsträckning att använda "bästa" namnservern, d.v.s. den som svarar snabbast.

- Namnservrarna ska tillhöra olika subnät och sitta i olika datahallar.
- Närhet i nätet ger bättre tillgänglighet. Utspridning som motsvarar användarna ger snabbare svar för användarna.
- Geografisk närhet är inte alltid nätmässig närhet.
- Det är bra med spridning mellan olika operatörer.

Nätredundans

IP-adresser tillhör olika AS(*) som är en routingenhet. Om alla namnservrar för en domän tillhör samma AS så ökar risken för utslagning.

- Placera namnservrarna på olika AS, om möjligt.
- Olika operatörer ger normalt olika AS.

*) https://en.wikipedia.org/wiki/Autonomous_system_(Internet)

Anycast

Anycast (som diskuterades i samband med rotzonen) är ett avancerat sätt att öka redundansen för alla typer av zoner.

Hur bra är denna delegering?

```
; <<>> DiG 9.10.6 <<>> @a.ns.se fk.se soa +nocl +nottl

(...)

;; AUTHORITY SECTION:
fk.se.          NS  ns4.forsakringskassan.se.
fk.se.          NS  ns3.forsakringskassan.se.
fk.se.          NS  sadbnsy1.sfa.se.

;; ADDITIONAL SECTION:
sadbnsy1.sfa.se.           A  194.71.64.2
ns4.forsakringskassan.se.  A  194.71.64.30
ns3.forsakringskassan.se.  A  194.71.64.29

;; Query time: 88 msec
;; SERVER: 192.36.144.107#53(192.36.144.107)
;; WHEN: Fri Feb 17 11:46:12 CET 2023
;; MSG SIZE  rcvd: 167
```

Delegeringen från .se

Alla namnservrar sitter i samma /24 och därmed i samma AS.

Datum när det gällde. Kan ha uppdaterats.

Baklängeszoner

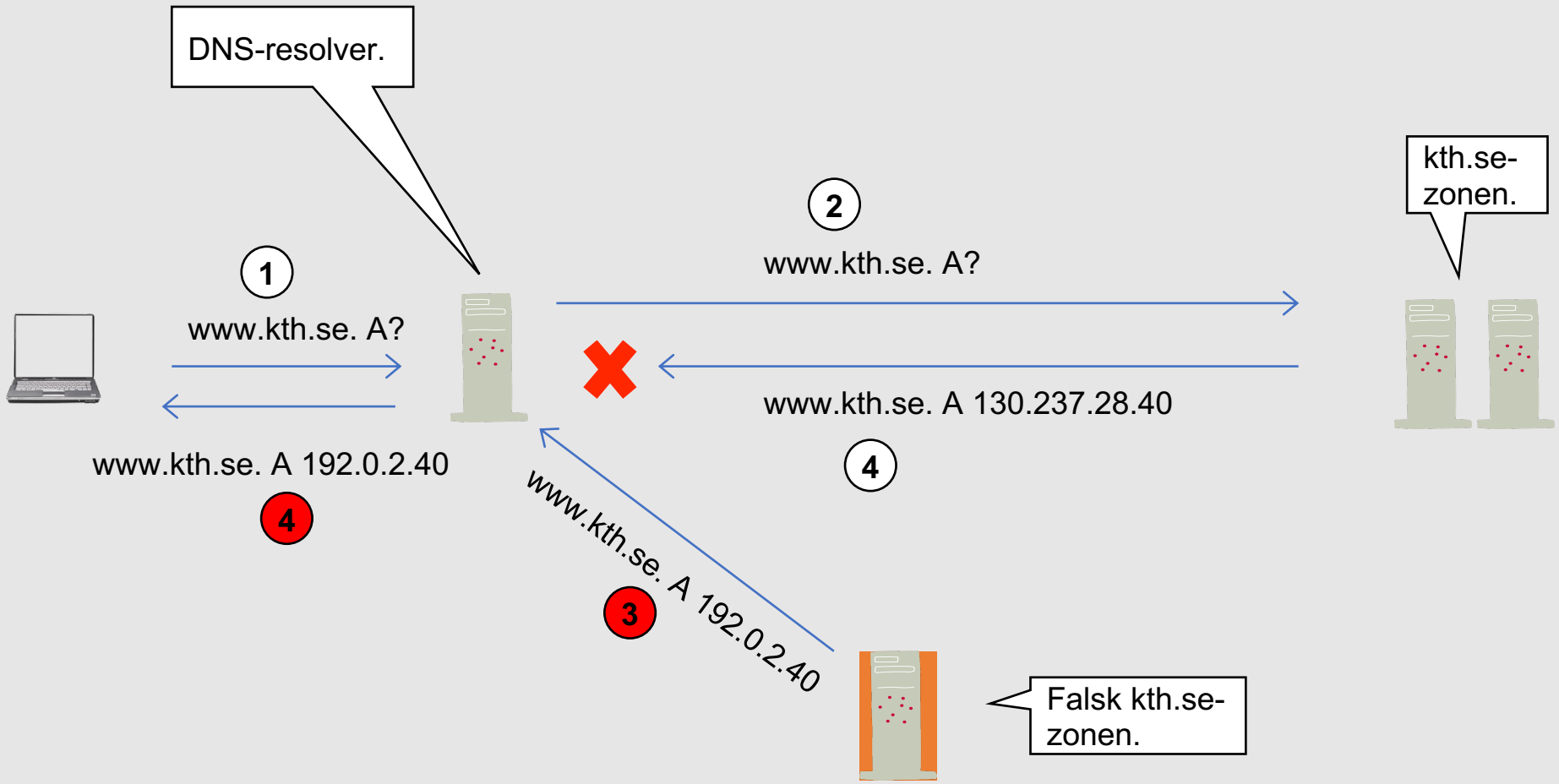
- Ställ samma krav på baklängeszonerna som framlängeszonerna.
- Lägg baklängeszonerna på samma servrar som framlängeszonerna.

Om det inte är viktigt med revers så är det bättre att ta bort reverszonerna och delegeringen till dem än att riskera långa svarstiden på baklängesuppslagning. Om det inte finns så blir det ett snabbt NXDOMAIN.

▶DNS och säkerhet

[Innehåll]

Uppslagning med ”man-in-the-middle”

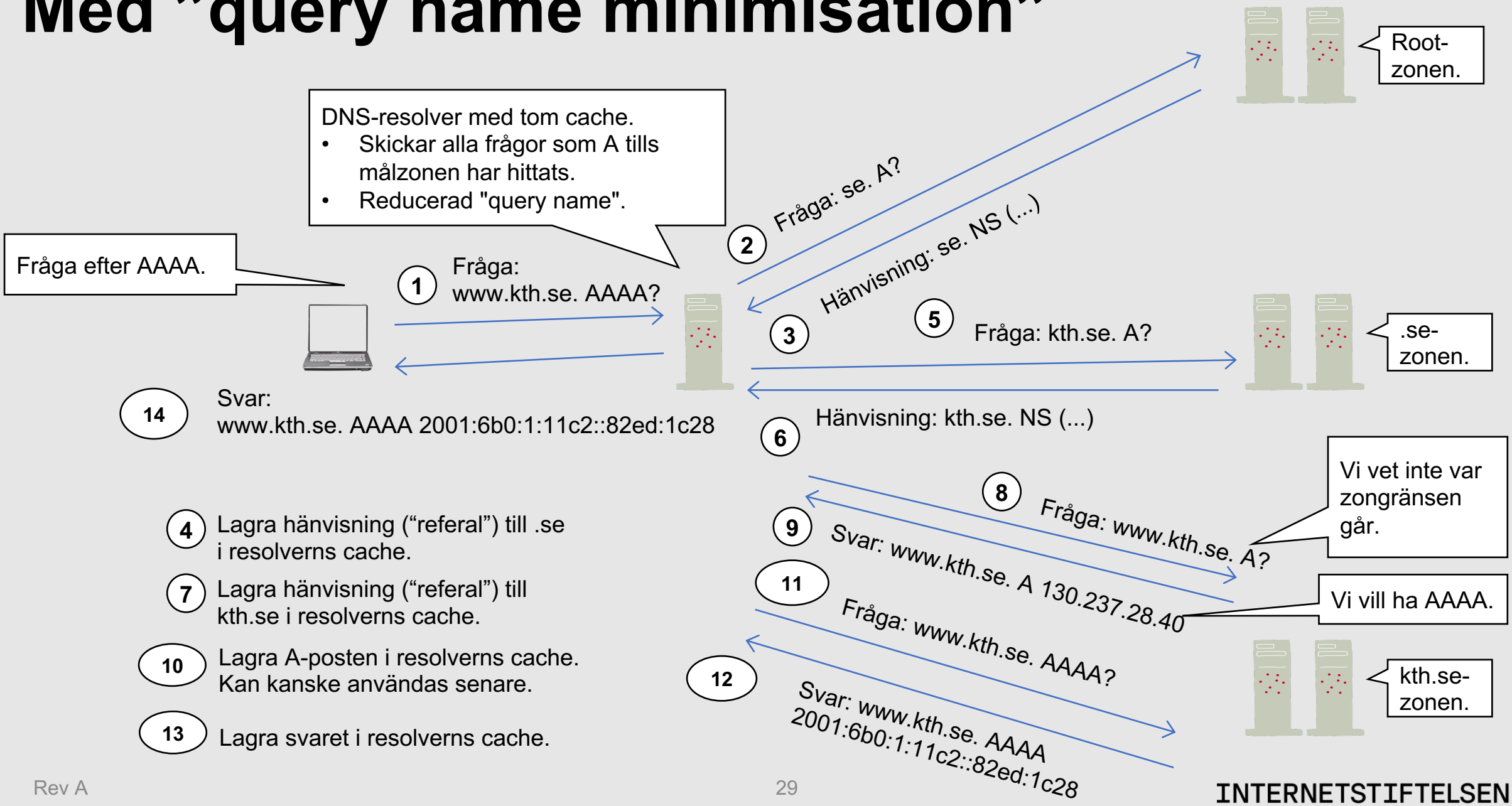


DNSSEC löser vissa problem

DNSSEC löser problemet med falsk data genom att introducera kryptografiska signaturer.

DNS skickas fortfarande i klartext.

Med ”query name minimisation”

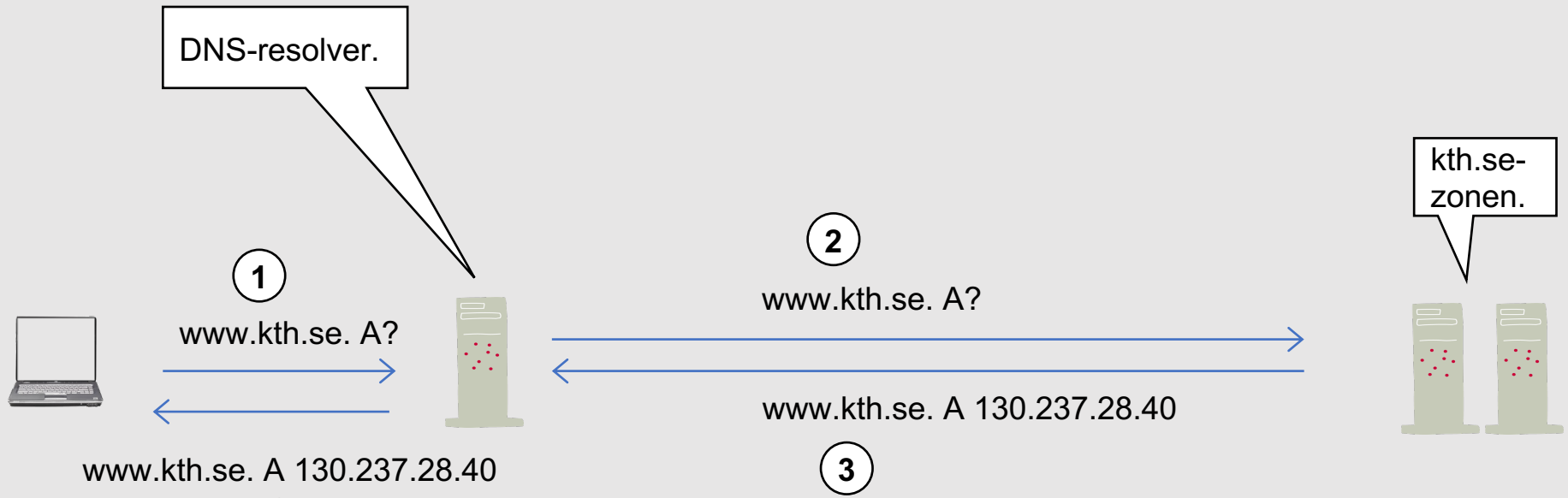


”QNAME minimisation” minskar spridningen

Med “query name minimisation” så minskar spridningen av frågan, vilket minskar exponeringen.

(Se tidigare föreläsning för mera detaljer om "query name minimisation".)

Frågor och svar



Trafiken mellan klient och resolver går i klartext och kan avlyssnas av den som sitter på rätt ställe. Och här kan det aldrig bli någon ”query name minimisation”. Ju längre bort resolvern sitter, ju fler platser kan avlyssna.

Skyddar DNSSEC svaren från resolvern?

Om resolvern validerar DNSSEC – och domänen i fråga är signerad – så kommer resolverns svar att vara skyddat från förvanskning när den lämnar resolvern.

- Om svaret skickas oskyddat i klartext till klienten – det normala – så måste klienten göra om valideringen för att vara säker.

I teorin så skulle TSIG kunna användas som skydd, men det skalar inte. Fortfarande så går trafiken i klartext även med TSIG.

▶Krypterad DNS

[Innehåll]

Kryptering på Internet

Det finns en allmän ambition när det gäller standardiseringen för Internet att kommunikationen ska vara, eller ska kunna vara, krypterad.

Det gäller t.ex. http och mail.

Det säkerhetslager som har fått mest utbredning är TLS, “Transport Layer Security”, som fungerar som ett lager mellan TCP och applicationen. Föregångaren till TLS hette SSL.

https://en.wikipedia.org/wiki/Transport_Layer_Security

DNS går normalt över UDP

DNS går normalt över UDP. När DNS ska krypteras så visar det sig bli komplicerat. Försöken med krypterad DNS över UDP har inte varit framgångsrika och har stannat på experimentstadiet.

En komplexitet har varit att alla implementationer också måste har stöd för TCP, vilket betyder att två lösningar måste tas fram.

Som det ser ut nu så kan vi bortse från TLS över UDP för DNS.

▶Krypterad DNS – DoT

[Innehåll]

DNS över TLS, DoT

DNS över TLS (över TCP) fungerar som vanlig DNS över TCP, men med ett TLS-lager mellan. Det är en utökning av standarden för DNS.

Vanliga port 53 ska inte användas, utan standarden har fastställt att port 853 ska användas.

Målet för DoT är primärt att göra det möjligt att få kommunikationen mellan klient och resolver att bli krypterad och därmed skyddad från insyn.

Resolver till auktoritativ server

Kommunikationen mellan resolver och auktoritativ namnserver (master eller slav) är mindre känslig och kan komma i ett senare skede.

Utmaningar för DoT

DoT är tänkt att ersätta vanlig DNS-kommunikation, men det krävs att båda klient och server är uppdaterad.

DNS på klientsidan sitter huvudsakligen som bibliotek i OS:et. Det kan göra det komplext att ändra på det. Dock har nyare versioner av Android stöd för DoT.

Även på serversidan så måste stödet läggas till, men där är det tekniskt enklare.

Utmaningar för DoT

Så länge användarna är få blir trycket på resolveroperatörerna inte så stort att införa, vilket gör att färre användare kan använda det

Stora öppna resolvrar som 1.1.1.1, 8.8.8.8 och 9.9.9.9 har stöd för DoT.

Brandväggar kan dock ställa till det om de inte släpper ut trafik till en okänd port, 853.

Utmaningar för DoT

All TLS bygger på servern som kontakts också har en identitet, ett domännamn, och att servern kan leverera ett certifikat för det namnet.

Certifikatet ska också kunna valideras. Om klienten bara har en IP-adress och inte ett namn på servern så kan den inte säkerställa att certifikatet är det som förväntas.

Utmaningar för DoT

Om klienten bara accepterar kända certifikat så finns det risk att klienten i vissa lägen måste vara utan DNS.

Om klienten accepterar det den får, så kan klienten inte vara säker på att den får den integritet som den efterfrågar.

DoT-stöd i dig

Nyare version av “dig” har stöd för DoT.

- Versionen på dig ska vara 9.18 eller högre (“dig -v”).
- Optionen “+tls” gör att frågan skickas över TLS över TCP.
- Porten blir då 853 istället för 53.
- Det krävs att servern har stöd för DoT.

```
$ dig –v
DiG 9.18.1-1ubuntu1.3-Ubuntu
```

DoT med dig

”+tls” gör att ”dig” frågar över DoT

Google har stöd för DoT, men det gäller även 1.1.1.1 och 9.9.9.9.

```
$ dig +tls www.sunet.se @8.8.8.8

; <<>> DiG 9.18.1-1ubuntu1.3-Ubuntu <<>> +tls www.sunet.se @dns.google
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27203
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.sunet.se. IN A

;; ANSWER SECTION:
www.sunet.se. 21063 IN A 37.156.192.50
www.sunet.se. 21063 IN A 37.156.192.51

;; Query time: 24 msec
;; SERVER: 8.8.8.8#853(8.8.8.8) (TLS)
;; WHEN: Fri Feb 17 16:07:36 UTC 2023
;; MSG SIZE  rcvd: 73
```

”dig” version 9.18 har stöd för DoT, men inte äldre versioner av ”dig”.

DoT använder port 853.

▶Krypterad DNS – DoH

[Innehåll]

DNS över HTTPS, DoH

DNS över HTTPS, DoH, kommer från webbläsarindustrin (Firefox m.fl.).

Även DoH är en utökning av DNS-standarden.

DNS över HTTPS, DoH

Istället för att ta vanlig DNS över en ny kanal för generell användning så valdes en annan vinkel:

- Lösningen är först tänkt för webbläsare.
- Trafiken går som vanlig HTTP över TLS (HTTPS) över vanliga port 443.
- DNS-meddelandena ligger inbäddad i vanlig HTTP, samma format som skickas över UDP, men Base64-kodat.
- DNS-frågorna kan tänkas gå till samma server som webbläsaren hämtar webbdata från.

Utmaningar för DoH

DoH bygger liksom DoT på att DNS körs över TLS över TCP.

Även DoH är tänkt att vara kanalen mellan klient och resolver, inte för kanalen mellan resolver och auktoritativ server.

DoH är tänkt att förkonfigureras i de klientapplikationer som ska använda DoH, snarare än att vara en generellt DNS-bibliotek. Och webbläsaren är den primära målgruppen.

Men detta kan komma att ändras så att DoH blir generellt.

DoH och brandväggar

Brandväggar tillåter ofta utgående HTTP och HTTPS. Det är svårt eller omöjligt att skilja mellan vanlig trafik och DoH.

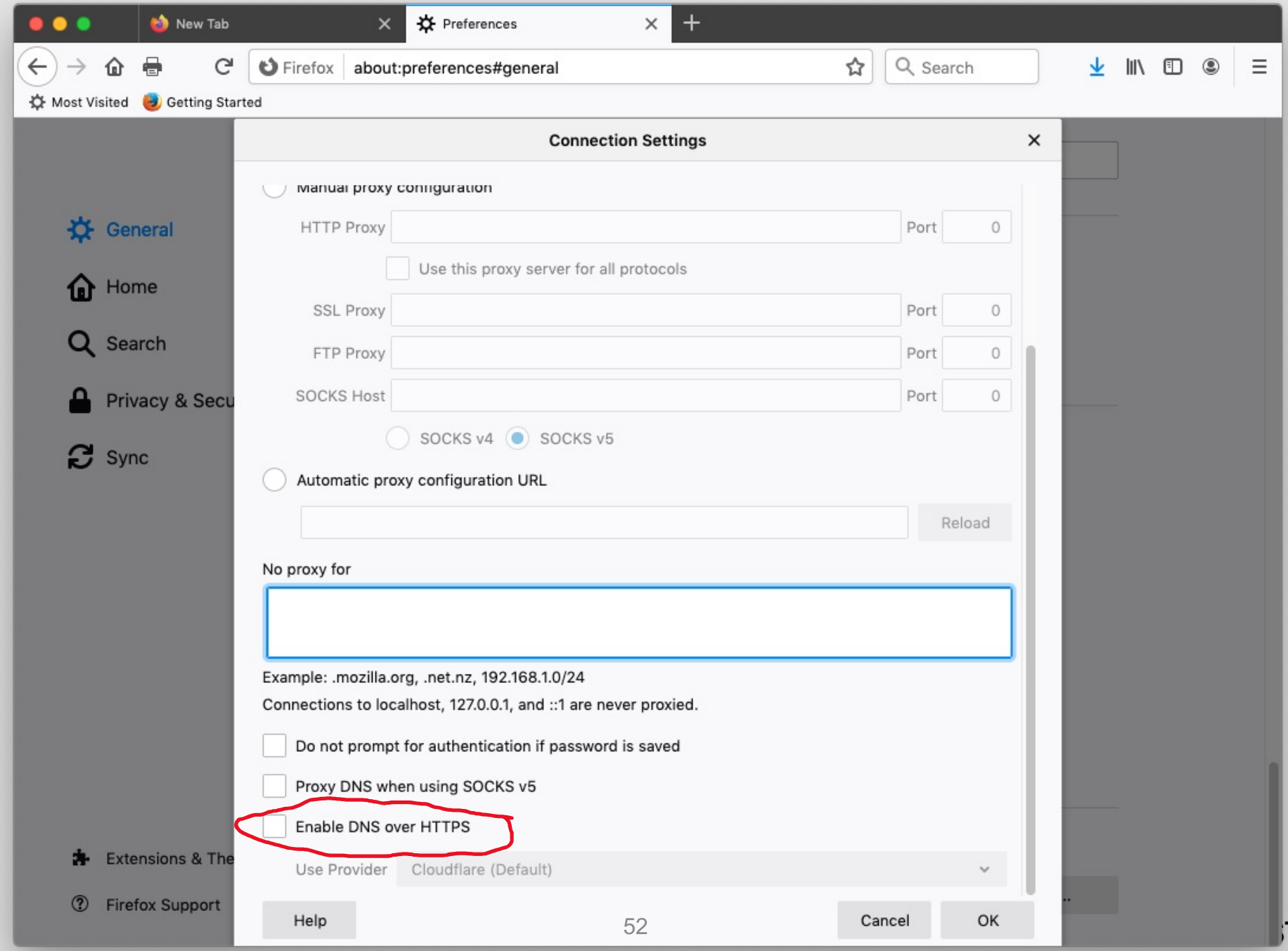
- Med DNS okrypterat över port 53 så kan brandväggar stoppa oönskade frågor.
- Med DoT så kan brandväggen helt stoppa det alternativet eller försöka fånga trafiken ifall klienten inte har full kontroll över servern och dess certifikat.
- Med DoH så blir utmaningen mycket större för den som vill kontrollera trafiken.

DoH hos resolveroperatörerna

Redan idag så finns det stöd för DoH hos Google (8.8.8.8), Quad9 (9.9.9.9) och Cloudflare (1.1.1.1).

DoH i webbläsaren

Firefox och Chrome har stöd för DoH. I Firefox så kan man enkelt att slå på DoH och börja använda Cloudflare som operatör.


New Tab
Preferences
Firefox
about:preferences#general
Search
Most Visited
Getting Started
General
Home
Search
Privacy & Secu
Sync
Extensions & The
Firefox Support
Connection Settings
Manual proxy configuration
HTTP Proxy
Port
0
Use this proxy server for all protocols
SSL Proxy
FTP Proxy
SOCKS Host
SOCKS v4
SOCKS v5
Automatic proxy configuration URL
Reload
No proxy for
Example: .mozilla.org, .net.nz, 192.168.1.0/24
Connections to localhost, 127.0.0.1, and ::1 are never proxied.
Do not prompt for authentication if password is saved
Proxy DNS when using SOCKS v5
Enable DNS over HTTPS
Use Provider
Cloudflare (Default)
Help
Cancel
OK

Firefox

Problem med DoH

Det finns problem med kopplingen mellan DoH och webbläsaren:

- Det finns en oro för att webbläsarindustrin ska ytterligare stärka sitt grepp om användarna. Användarna vinner kanske i att inte kontrolleras av lokal brandvägg, men istället blir det stora resolveroperatörer som sätter policy.
- Lokalt barnporrfilter med DNS blir kanske inte möjligt med DoH, eller så blir det mycket mer filter för att DoH-operatören tycker så.

Problem med DoH via webbläsaren

- Annan trafik, som inte bygger på webbläsaren använder inte DoH och kommer att få en annan uppslagning.
- Om den lokala resolvern tillhandahåller uppslagning av interna servrar bara internt, så kommer detta inte att finnas tillgängligt över DoH från en extern operatör.
- Det blir svårare att felsöka om DNS-trafiken går olika vägar. En väg för Firefox, annan för Chrome och en tredje för övrig trafik.

DoH-stöd i dig

Nyare version av “dig” har stöd för DoH.

- Versionen på dig ska vara 9.18 eller högre (“dig -v”).
- Optionen “+https” gör att frågan skickas över HTTP över TLS över TCP.
- Porten blir då 443 istället för 53.
- Det krävs att servern har stöd för DoH.

DoT med dig

```
$ dig +https www.sunet.se @8.8.8.8

; <<>> DiG 9.18.7 <<>> +https www.sunet.se A @8.8.8.8
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12495
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.sunet.se. IN A

;; ANSWER SECTION:
www.sunet.se. 20775 IN A 37.156.192.50
www.sunet.se. 20775 IN A 37.156.192.51

;; Query time: 5 msec
;; SERVER: 8.8.8.8#443(8.8.8.8) (HTTPS)
;; WHEN: Sun Feb 19 12:50:05 UTC 2023
;; MSG SIZE  rcvd: 73
```

”+https” gör att ”dig” frågar över DoH

Google har stöd för DoH, men det gäller även 1.1.1.1 och 9.9.9.9.

”dig” version 9.18 har stöd för DoH, men inte äldre versioner av ”dig”.

DoH använder port 443.

▶Krypterad DNS – DoQ

[Innehåll]

DNS över Quic, DoQ

QUIC är framtaget för att vara ett bättre transportprotokoll för webb. Det är HTTP över TLS över UDP. Det är ett nytt protokoll.

DoQ (“DNS over QUIC”) är som DoH, men använder QUIC istället.

DoQ är helt nytt tillägg till DNS-protokollet och har ännu inte fått någon större spridning. Det återstår om det lyckas bättre än DoT och DoH.

DNS över Quic, DoQ

Port för DoQ är 853, samma som för DoT, men här över UDP. Kanske kommer 443 att användas i vissa sammanhang eftersom DoH använder den porten och den porten går lättare igenom brandväggar.

Klarar stora svar, precis som TCP.

“dig” har f.n. inte stöd för DoQ, men det kommer säkert.

▶DNS över olika transportsätt

[Innehåll]

UDP, TCP, TLS, HTTPS eller QUIC

	UDP	TCP	DoT	DoH	DoQ
Transport	UDP	TCP	TLS över TCP	HTTP över TLS över TCP	HTTP över TLS över UDP
DNS-paketformat	UDP	TCP	TCP	Base64-kodat UDP	Base64-kodat UDP
Klarar stora paket	Ja med EDNS och ev. fragmentering	Ja	Ja	Ja	Ja
Port (lyssnande)	53	53	853	443	853
Krypterat	nej	nej	ja	ja	ja
Standardiserat för hosting	obligatorisk	obligatorisk	nej	nej	nej
Standardiserat för resolvning	obligatorisk	obligatorisk	ja	ja	ja

Spela in trafik

Vi ska jämföra DNS-trafiken beroende på om det är över UDP, TCP, TLS eller HTTPS. (Kunde inte spela in över QUIC.)

Trafiken spelades in genom att köra “dig” för varje transport i ett fönster och “tcpdump” i ett annat fönster till samma dator.

Trafiken spelades in till filer som sedan öppnades med “wireshark”.

tcpdump i fönster 1

```
$ tcpdump -n host 8.8.8.8 -w dig-query-udp-tcp-tls-https-2023-02-19.pcap
```

Spela in all trafik till och från 8.8.8.8 och spara i en fil.

dig i fönster 2

```
$ dig www.sunet.se A @8.8.8.8

; <<>> DiG 9.18.7 <<>> www.sunet.se A @8.8.8.8
(...)
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Sun Feb 19 10:55:28 UTC 2023
;; MSG SIZE  rcvd: 73

$ dig www.sunet.se A @8.8.8.8 +tcp

; <<>> DiG 9.18.7 <<>> www.sunet.se A @8.8.8.8 +tcp
(...)
;; SERVER: 8.8.8.8#53(8.8.8.8) (TCP)
;; WHEN: Sun Feb 19 10:56:28 UTC 2023
;; MSG SIZE  rcvd: 73

$ dig www.sunet.se A @8.8.8.8 +tls

; <<>> DiG 9.18.7 <<>> www.sunet.se A @8.8.8.8 +tls
(...)
;; SERVER: 8.8.8.8#853(8.8.8.8) (TLS)
;; WHEN: Sun Feb 19 10:57:28 UTC 2023
;; MSG SIZE  rcvd: 73

$ dig www.sunet.se A @8.8.8.8 +https

; <<>> DiG 9.18.7 <<>> www.sunet.se A @8.8.8.8 +https
(...)
;; SERVER: 8.8.8.8#443(8.8.8.8) (HTTPS)
;; WHEN: Sun Feb 19 10:58:28 UTC 2023
;; MSG SIZE  rcvd: 73
```

DoQ är inte inkluderat.

Öppna filen i wireshark

Den inspelade filen flyttades till lokal laptop och öppnades i wireshark som är ett grafiskt program för att titta på nätverkstrafik

Sedan gjordes skärmdumpar av valda delar av fönstret.

UDP

Ett paket för frågan, ett för svaret. Fråga och svar i klartext. 2 IP-paket. 3 ms mellan första och andra paketet.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000	172.31.10.109	8.8.8.8	DNS	95	Standard query 0x7714 A www.sunet.se OPT
2	0.003	8.8.8.8	172.31.10.109	DNS	115	Standard query response 0x7714 A www.sunet.se A 37.156.192.51 A 37.156.192.50 OPT

TCP

Fråga och svar svar i en TCP-session. Fråga och svar i klartext. 9 IP-paket.12 ms mellan första och sista.

3	60.047	172.31.10.109	8.8.8.8	TCP	74	44983 → 53 [SYN] Seq=0 Win=65535 Len=0 MSS=8961 WS=64 SACK_PERM TSval=108777257 TSecr=0
4	60.050	8.8.8.8	172.31.10.109	TCP	74	53 → 44983 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM TSval=2029173481 …
5	60.050	172.31.10.109	8.8.8.8	TCP	66	44983 → 53 [ACK] Seq=1 Ack=1 Win=66368 Len=0 TSval=108777257 TSecr=2029173481
6	60.050	172.31.10.109	8.8.8.8	DNS	121	Standard query 0x3900 A www.sunet.se OPT
7	60.054	8.8.8.8	172.31.10.109	TCP	66	53 → 44983 [ACK] Seq=1 Ack=56 Win=65536 Len=0 TSval=2029173484 TSecr=108777257
8	60.054	8.8.8.8	172.31.10.109	DNS	141	Standard query response 0x3900 A www.sunet.se A 37.156.192.51 A 37.156.192.50 OPT
9	60.055	172.31.10.109	8.8.8.8	TCP	66	44983 → 53 [FIN, ACK] Seq=56 Ack=76 Win=66368 Len=0 TSval=108777267 TSecr=2029173484
10	60.059	8.8.8.8	172.31.10.109	TCP	66	53 → 44983 [FIN, ACK] Seq=76 Ack=57 Win=65536 Len=0 TSval=2029173489 TSecr=108777267
11	60.059	172.31.10.109	8.8.8.8	TCP	66	44983 → 53 [ACK] Seq=57 Ack=77 Win=66304 Len=0 TSval=108777267 TSecr=2029173489

DoT (TLS)

DNS-trafiken är dold i en TLS-session. 19 IP-paket. 35 ms mellan första och sista.

12	120.146	172.31.10.109	8.8.8.8	TCP	74	25679 → 853 [SYN] Seq=0 Win=65535 Len=0 MSS=8961 WS=64 SACK_PERM TSval=917248238 TSecr…
13	120.149	8.8.8.8	172.31.10.109	TCP	74	853 → 25679 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM TSval=3764790808…
14	120.149	172.31.10.109	8.8.8.8	TCP	66	25679 → 853 [ACK] Seq=1 Ack=1 Win=66368 Len=0 TSval=917248238 TSecr=3764790808
15	120.150	172.31.10.109	8.8.8.8	TLSv1.3	365	Client Hello
16	120.153	8.8.8.8	172.31.10.109	TCP	66	853 → 25679 [ACK] Seq=1 Ack=300 Win=66816 Len=0 TSval=3764790812 TSecr=917248238
17	120.164	8.8.8.8	172.31.10.109	TLSv1.3	1466	Server Hello, Change Cipher Spec
18	120.164	8.8.8.8	172.31.10.109	TCP	1466	853 → 25679 [ACK] Seq=1401 Ack=300 Win=66816 Len=1400 TSval=3764790823 TSecr=917248238…
19	120.164	172.31.10.109	8.8.8.8	TCP	66	25679 → 853 [ACK] Seq=300 Ack=2801 Win=63616 Len=0 TSval=917248258 TSecr=3764790823
20	120.164	8.8.8.8	172.31.10.109	TCP	1466	853 → 25679 [ACK] Seq=2801 Ack=300 Win=66816 Len=1400 TSval=3764790823 TSecr=917248238…
21	120.164	8.8.8.8	172.31.10.109	TLSv1.3	692	Application Data
22	120.164	172.31.10.109	8.8.8.8	TCP	66	25679 → 853 [ACK] Seq=300 Ack=4827 Win=61632 Len=0 TSval=917248258 TSecr=3764790823
23	120.165	172.31.10.109	8.8.8.8	TLSv1.3	146	Change Cipher Spec, Application Data
24	120.173	8.8.8.8	172.31.10.109	TCP	66	853 → 25679 [ACK] Seq=4827 Ack=380 Win=66816 Len=0 TSval=3764790832 TSecr=917248258
25	120.173	172.31.10.109	8.8.8.8	TLSv1.3	143	Application Data
26	120.176	8.8.8.8	172.31.10.109	TCP	66	853 → 25679 [ACK] Seq=4827 Ack=457 Win=66816 Len=0 TSval=3764790835 TSecr=917248258
27	120.177	8.8.8.8	172.31.10.109	TLSv1.3	707	Application Data, Application Data
28	120.178	172.31.10.109	8.8.8.8	TCP	66	25679 → 853 [FIN, ACK] Seq=457 Ack=5468 Win=66368 Len=0 TSval=917248271 TSecr=37647908…
29	120.181	8.8.8.8	172.31.10.109	TCP	66	853 → 25679 [FIN, ACK] Seq=5468 Ack=458 Win=66816 Len=0 TSval=3764790840 TSecr=9172482…
30	120.181	172.31.10.109	8.8.8.8	TCP	66	25679 → 853 [ACK] Seq=458 Ack=5469 Win=66304 Len=0 TSval=917248271 TSecr=3764790840

DoH (HTTPS)

DNS-trafiken dold i en TLS-session. 29 IP-paket. 32 ms mellan första och sista.

```
   31 180.251    172.31.10.109  8.8.8.8        TCP        74 40541 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=8961 WS=64 SACK_PERM TSval=664811430 TSecr…
   32 180.256    8.8.8.8        172.31.10.109  TCP        74 443 → 40541 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM TSval=3977132060…
   33 180.256    172.31.10.109  8.8.8.8        TCP        66 40541 → 443 [ACK] Seq=1 Ack=1 Win=66368 Len=0 TSval=664811442 TSecr=3977132060
   34 180.256    172.31.10.109  8.8.8.8        TLSv1.3   368 Client Hello
   35 180.260    8.8.8.8        172.31.10.109  TCP        66 443 → 40541 [ACK] Seq=1 Ack=303 Win=66816 Len=0 TSval=3977132065 TSecr=664811442
   36 180.270    8.8.8.8        172.31.10.109  TLSv1.3  1466 Server Hello, Change Cipher Spec
   37 180.270    8.8.8.8        172.31.10.109  TCP      1466 443 → 40541 [ACK] Seq=1401 Ack=303 Win=66816 Len=1400 TSval=3977132074 TSecr=664811442…
   38 180.270    172.31.10.109  8.8.8.8        TCP        66 40541 → 443 [ACK] Seq=303 Ack=2801 Win=63616 Len=0 TSval=664811456 TSecr=3977132074
   39 180.270    8.8.8.8        172.31.10.109  TCP      1466 443 → 40541 [ACK] Seq=2801 Ack=303 Win=66816 Len=1400 TSval=3977132074 TSecr=664811442…
   40 180.270    8.8.8.8        172.31.10.109  TLSv1.3   701 Application Data
   41 180.270    172.31.10.109  8.8.8.8        TCP        66 40541 → 443 [ACK] Seq=303 Ack=4836 Win=61632 Len=0 TSval=664811456 TSecr=3977132074
   42 180.271    172.31.10.109  8.8.8.8        TLSv1.3   146 Change Cipher Spec, Application Data
   43 180.271    172.31.10.109  8.8.8.8        TLSv1.3   127 Application Data
   44 180.271    172.31.10.109  8.8.8.8        TLSv1.3   250 Application Data
   45 180.276    8.8.8.8        172.31.10.109  TLSv1.3   714 Application Data, Application Data
   46 180.276    172.31.10.109  8.8.8.8        TLSv1.3    97 Application Data
   47 180.276    8.8.8.8        172.31.10.109  TLSv1.3    97 Application Data
   48 180.277    8.8.8.8        172.31.10.109  TLSv1.3   314 Application Data
   49 180.277    172.31.10.109  8.8.8.8        TCP        66 40541 → 443 [ACK] Seq=659 Ack=5763 Win=66176 Len=0 TSval=664811462 TSecr=3977132080
   50 180.277    8.8.8.8        172.31.10.109  TLSv1.3   170 Application Data
   51 180.277    8.8.8.8        172.31.10.109  TLSv1.3    97 Application Data
   52 180.277    172.31.10.109  8.8.8.8        TCP        66 40541 → 443 [ACK] Seq=659 Ack=5898 Win=66368 Len=0 TSval=664811462 TSecr=3977132081
   53 180.277    8.8.8.8        172.31.10.109  TLSv1.3   105 Application Data
   54 180.277    172.31.10.109  8.8.8.8        TLSv1.3    90 Application Data
   55 180.277    172.31.10.109  8.8.8.8        TCP        66 40541 → 443 [FIN, ACK] Seq=683 Ack=5937 Win=66368 Len=0 TSval=664811462 TSecr=39771320…
   56 180.282    8.8.8.8        172.31.10.109  TCP        66 443 → 40541 [ACK] Seq=5937 Ack=683 Win=67840 Len=0 TSval=3977132086 TSecr=664811462
   57 180.282    8.8.8.8        172.31.10.109  TCP        66 443 → 40541 [FIN, ACK] Seq=5937 Ack=683 Win=67840 Len=0 TSval=3977132086 TSecr=6648114…
   58 180.282    172.31.10.109  8.8.8.8        TCP        66 [TCP Retransmission] 40541 → 443 [FIN, ACK] Seq=683 Ack=5938 Win=66368 Len=0 TSval=664…
   59 180.282    8.8.8.8        172.31.10.109  TCP        66 443 → 40541 [ACK] Seq=5938 Ack=684 Win=67840 Len=0 TSval=3977132086 TSecr=664811462
```

▶TXT-poster för mail

[Innehåll]

TXT-poster för mail

Vi har redan sett hur MX används för att slå upp mailserver.

Nu ska vi titta på exempel på hur speciella TXT-poster används av funktioner för mailsystemet: SPF, DKIM och DMARC.

Bakgrunden är problemet med spam och falska mail. Detta är ett exempel på hur DNS används som en allmänt tillgänglig databas för annan information än IP-adresser.

Posttyp TXT

SPF, DKIM och DMARC är alltså ***inte*** nya posttyper, utan funktioner som använder posttyp **TXT** enligt samma specifikation som tidigare.

▶TXT-post för mail – SPF

[Innehåll]

SPF

Sender Policy Framework (SPF) specificerar från vilken mailserver som mailet får skickas från.

Mailservern som tar emot mailet kan slå upp informationen i DNS enligt ett standardiserat format (om informationen finns). Mottagande mailserver använder den så kallade ”kuvertavsändaren” (”envelop from”).

https://en.wikipedia.org/wiki/Sender_Policy_Framework

SPF-post i DNS

Låt oss anta att kuvertavsändaren är ”dufberg@kth.se”

SPF-post hittar man i DNS som ”<MAILDOMÄN> TXT”, d.v.s.:

”kth.se. TXT” i detta fall.

SPF-post för kth.se

```
$ dig kth.se txt
(...)
;; ANSWER SECTION:
kth.se. 1800 IN TXT "google-site-verification=6B_ijtuPLs8QEQOrY6CA9KNR6x_Tlq9h87fvzbQcTGk"
kth.se. 1800 IN TXT "v=spf1 include:_spf.kth.se include:customers.clickdimensions.com ~all"
kth.se. 1800 IN TXT "MS=ms86914267"
kth.se. 1800 IN TXT "atlassian-domain-
verification=bdViuC6aa8uovu2aPgsZjA4zglkgbkqx7eCtfJtIXi6uWFglmI96V2EdENETiV1k"
kth.se. 1800 IN TXT "adobe-idp-site-
verification=9cb3a0e2a6197322d0dd6d29de0e1ac1ce9f990e3d5bd864aa4c19007f78ef91"
kth.se. 1800 IN TXT "include:_spf.eu.messagegears.net"
kth.se. 1800 IN TXT "apple-domain-verification=yiK5a1btuIncRk6Q"
kth.se. 1800 IN TXT "KTH Royal Institute of Technology, SWEDEN"
kth.se. 1800 IN TXT "facebook-domain-verification=ybxcywd9y5omfyzv9a4hi1122pxhys"
(...)
```

Plocka ut den TXT-post som börjar på ”v=spf1” (får bara finnas en) och använd RDATA.

SPF-post för kth.se

SPF-posten (TXT-posten) innehåller policy för kth.se.

```
kth.se. TXT "v=spf1 include:_spf.kth.se
include:customers.clickdimensions.com ~all"
```

(TXT-posten visas här på två rader för att få plats.)

Slå upp ev. ”include”.

SPF för kth.se

Uppslagning av ”include” som ger yttterligare ”include”.

```
_spf.kth.se. TXT "v=spf1 ip4:130.237.32.0/24 ip4:130.237.48.0/24
ip6:2001:6b0:1:1200::/64 ip6:2001:6b0:1:1300::/64 ~all"

customers.clickdimensions.com. TXT "v=spf1 include:_spf.messagegears.net
include:_spf.eu.messagegears.net ~all”
```

(TXT-posten visas här med radbrytningar för att få plats.)

SPF för kth.se

Uppslagning av ytterligare ”include”.

```
_spf.messagegears.net. TXT "v=spf1 ip4:135.84.216.0/22
ip4:66.240.227.0/24 ip4:63.143.59.128/25 ip4:63.143.57.128/25
ip4:216.98.158.0/24 ip4:74.63.212.0/24 ip4:52.200.59.0/24
ip4:34.245.210.0/24 -all"

_spf.eu.messagegears.net. TXT "v=spf1 ip4:34.245.210.0/24 -all"
```

(TXT-posten visas här med radbrytningar för att få plats.)

SPF för kth.se

Alla delar tillsammans ger SPF-policy.

"v=spf1 include:_spf.kth.se include:customers.clickdimensions.com ~all"

"v=spf1 ip4:130.237.32.0/24 ip4:130.237.48.0/24 ip6:2001:6b0:1:1200::/64 ip6:2001:6b0:1:1300::/64 ~all"

"v=spf1 include:_spf.messagegears.net include:_spf.eu.messagegears.net ~all"

"v=spf1 ip4:135.84.216.0/22 ip4:66.240.227.0/24 ip4:63.143.59.128/25 ip4:63.143.57.128/25 ip4:216.98.158.0/24 ip4:74.63.212.0/24 ip4:52.200.59.0/24 ip4:34.245.210.0/24 -all"

"v=spf1 ip4:34.245.210.0/24 -all"

▶TXT-post för mail – DKIM

[Innehåll]

DKIM

DomainKeys Identified Mail (**DKIM**) gör det möjligt för avsändande mailserver att signera delar av mailet för att motverka spoofing. Det är mottagande mailserver som gör kontrollen.

https://en.wikipedia.org/wiki/DomainKeys_Identified_Mail

DKIM-post i mailheader

I ett mail från någon på ”@kth.se” finns bl.a. följande i mailheadern (syns normalt inte, men mailklienten kan visa den):

Parametrarma ”d” och ”s” behövs för att kunna slå upp DKIM-posten i DNS.

```
(...)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; d=kth.se; s=default;
    t=1697627661; bh=XLzEuIa/5TZyxaK+GGTdxbus0ZVFR9SwzAGROPgolGw=;
    h=From:To:Subject:Date:References:In-Reply-To:From;
    b=ZgGqQZbwVmREYFp+WQSrboOOikPuvfD/iZG6xhcVYhfZM4gW+/mXE1IX11LyaZ+/u
    BdVf/HsdmPFBNSKrMAuqHfv2UVHM2uQaYR3ILdya7bWICX8MThHWbs3MaQEymEIYUv
    IUpLjoof1ARGz04NSnt++/KKNZAvJhiJFCSsieM8=
(...)
```

DKIM-post i DNS

För att slå upp DKIM-posten behöver vi följande från ”DKIM-signature” från mailheadern: ”d=kth.se; s=default;”. Det är ”domain” resp. ”selector”:

<SELECTOR>._domainkey.<DOMAIN>. TXT

I detta fall:

default._domainkey.kth.se. TXT

DKIM-post i DNS

```
dig default._domainkey.kth.se. TXT

(...)
;; ANSWER SECTION:
default._domainkey.kth.se. 7200  IN TXT  "v=DKIM1; k=rsa; "
"p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCkWYVVrcCLYg845ds+Mm
2mLNAMzRFf4PJ9F1LhpzSO3toZQzKBTY+0T27WyJu5R/g86tZHZCIvBAdeUWv
siw9LityPGHZts7qoRPGAk6DPXWEWGJhmdcGLcxRJX7kNn1JvXpVLywnzoLRL
/YYs42emR8LK1uqWZ22m2CPtNuoJ2QIDAQAB"
```

Parametern ”p” är den publika nyckeln till den privata nyckeln som har signerat valda delar av mailet.

▶TXT-post för mail – DMARC

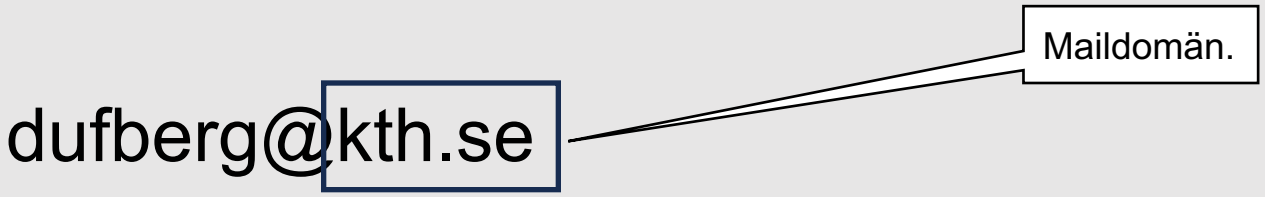
[Innehåll]

DMARC

Domain-based Message Authentication, Reporting and Conformance (**DMARC**) är också en mekanism för att autenticera skickandet av mail. Det är en utökning av SPF och DKIM. Det tillför också en mekanism för innehavaren av en maildomän att få rapporter om hur mottagande servrar har verifierat – eller inte – mail från maildomänen.

https://en.wikipedia.org/wiki/DMARC

DMARC-post i DNS



DMARC-post hittar man i DNS som ”_dmarc.<MAILDOMÄN> TXT”, d.v.s.:

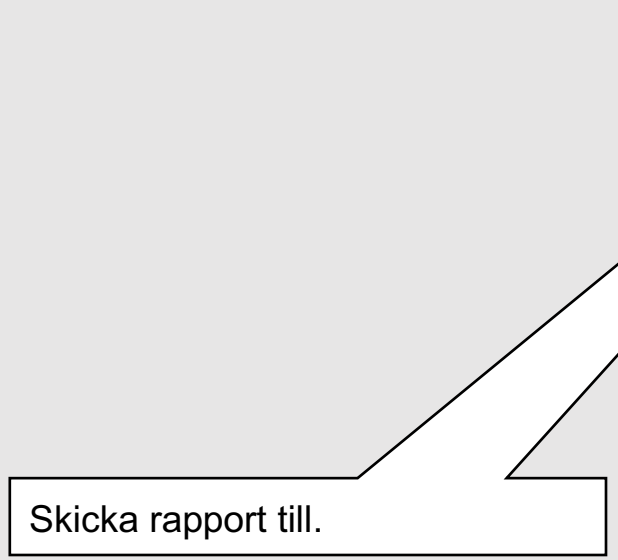
”_dmarc.kth.se. TXT” i detta fall.

DMARC-post för kth.se

```
$ dig _dmarc.kth.se txt

(...)

;; ANSWER SECTION:
_dmarc.kth.se. 7200 IN TXT "v=DMARC1;p=none;pct=100;rua=mailto:dmarc-admin@kth.se"
```



▶Posttyp SRV

[Innehåll]

Posttyp SRV

SRV returnerar server för en viss tjänst. Det är som en generaliserad MX.

Det används t.ex. för sip (“session initiation protocol”) för IP-telefoni och för att hitta AD (Microsoft Active Directory). Exempel:

```
_ldap._tcp.namn.se. SRV 0 100 389 ad1.namn.se.
_ldap._tcp.namn.se. SRV 0 100 389 ad2.namn.se.
```

(Notera domännamn med “_”.)

Posttyp SRV och SIP

En SIP-adress för IP-telefoni kan se ut så här:

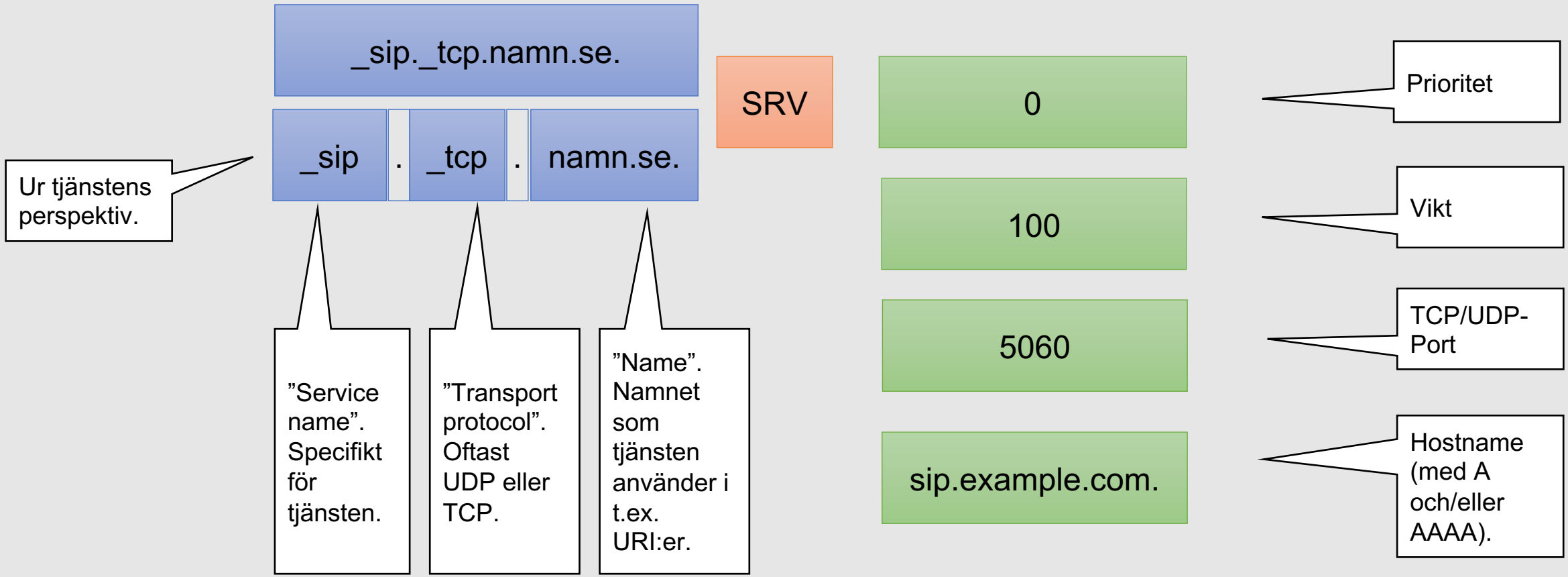
sip:mats@namn.se

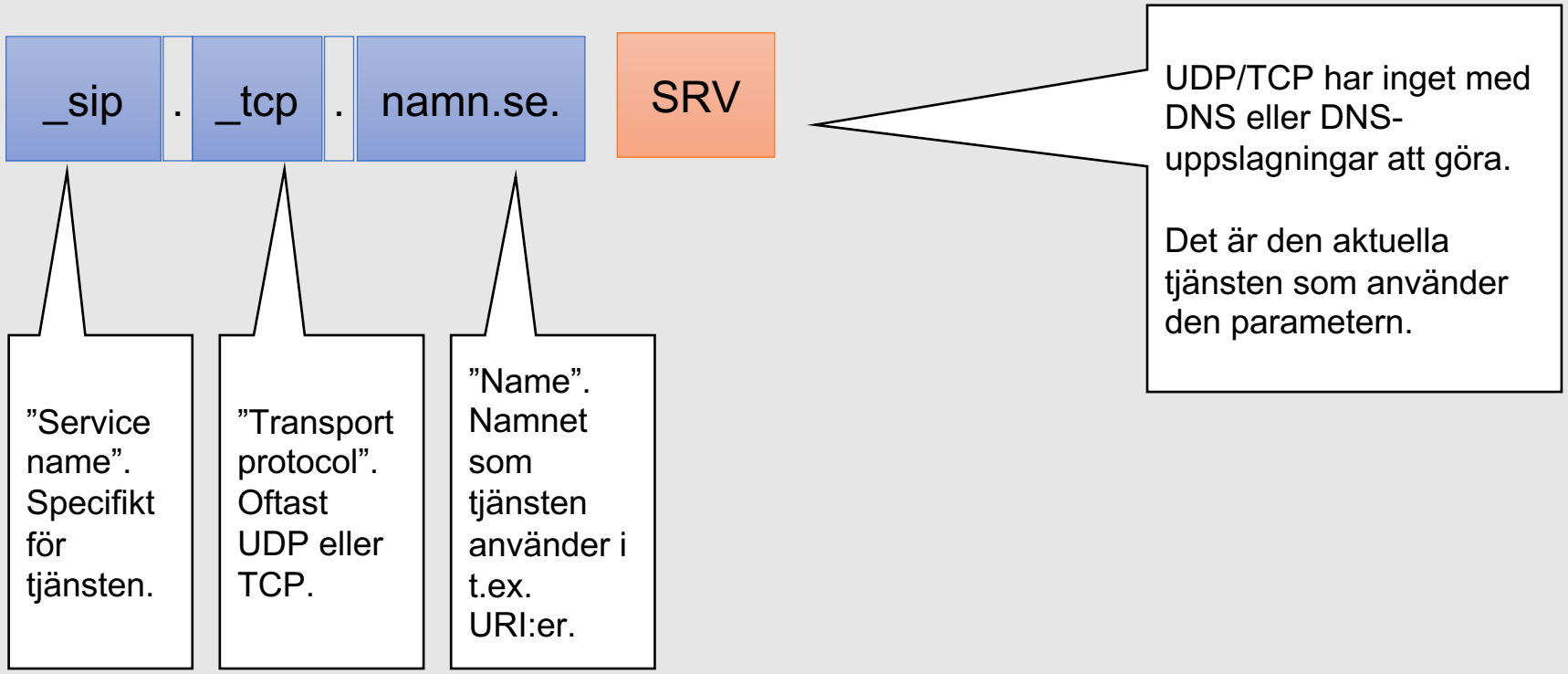
IP-telefoniklienten ska då finna den sip-server som har ett konto för denna adress. Den använder då en SRV-fråga där domänen “namn.se” omvandlas till rätt ”owner name”. Klienten väljer här att den vill kommunicera över TCP (men kan också fråga om UDP), och det är ju SIP det gäller.

namn.se → _sip._tcp.namn.se

Ur DNS-perspeiktiv:

- ”Owner name”, d.v.s. platsen i domännamnsträdet.
- Det är detta namn som DNS-uppslagningen kommer att gälla.




_sip
.
_tcp
.
namn.se.
SRV
”Service name”. Specifikt för tjänsten.
”Transport protocol”. Oftast UDP eller TCP.
”Name”. Namnet som tjänsten använder i t.ex. URI:er.
UDP/TCP har inget med DNS eller DNS-uppslagningar att göra.
Det är den aktuella tjänsten som använder den parametern.

Tecken i SRV “labels”

Notera att de två “labels” som läggs på för “service” resp. “transport protocol” startar på understreck “_” som annars inte används.

Det är inget “hostname” och det finns aldrig någon A eller AAAA på det namnet.

```
_sip._tcp.namn.se.
```

▶DANE och posttyp TLSA

[Innehåll]

Lagra information om certifikat för TLS

I DNS kan man tänka sig att lagra olika slags information. Med DNSSEC så har det blivit möjligt att lagra information som kräver att vi kan säkerställa att informationen är korrekt.

Specifikt har det blivit möjligt att binda TLS-certifikat till en DNSSEC-signerad domän.

DANE

DANE kan användas för att binda rätt certifikat till mailservrar som använder protokollet SMTP över TLS. SMTP är det vanligaste protokollet för mailstransport och det är önskvärt att transporten går över TLS.

Här ett exempel på hur DNS kan användas för att underlätta SMTP över TLS.

För DANE används TLSA-poster.

Mail till Internetstiftelsen

MX styr vart mail till @internetstifltelsen.se ska skickas:

```
internetstiftelsen.se.   MX  5 mx1.iis.se.
internetstiftelsen.se.   MX  5 mx2.iis.se.
```

Vilket certifikat ska accepteras när man anropar mailservrarna?

- Det kan TLSA-posterna berätta.

```
_25._tcp.mx1.iis.se. TLSA   3 1 1 (
            3EC9DC5D031807738EF1CCF91D5990AA9BC110D9F250
            2DBCC44FCA8D80E18426 )
_25._tcp.mx2.iis.se. TLSA   3 1 1 (
            F199402AB4F095F1477ECDF3B0922CC23F49CA7BA1F8
            7B014D74ED014D5FCA86 )
```

Ur DNS-perspeiktiv:

- ”Owner name”, d.v.s. platsen i domännamnsträdet.
- Det är detta namn som DNS-uppslagningen kommer att gälla.

_25._tcp.mx1.iis.se.

Ur tjänstens perspektiv.

_25 . _tcp . mx1.iis.se.

”Port number”. Specifikt för tjänsten. Notera "_".

”Transport protocol”. Oftast UDP eller TCP. Notera "_".

”Name”. Namnet som tjänsten utgår från.

TLSA

3

1

1

3EC9DC5D031807738EF1CCF91D5990AA9BC110D9F2502DBCC44FCA8D80E18426

Fyra fält som binder servern till rätt certifikat.

Verifiering av certifikat

En TLS-server skickar ett certifikat till klienten.

En SMTP-klient (i detta fall en annan SMTP-server) kopplar upp sig för att leverera mail till t.ex. mx1.iis.se. Om SMTP-klienten har stöd för DANE så kan den hämta TLSA-posten och verifiera TLSA-posten mot DNSSEC.

Till slut så kan man jämföra certifikatet som mx1.iis.se presenterar med informationen i TLSA-posten.

▶Om presentationen

[Innehåll]

Internets domännamnssystem

Denna presentation är framtagen 2019–2023 av Mats Dufberg (mats.dufberg@internetstiftelsen.se) på Internetstiftelsen (https://internetstiftelsen.se/). Den är en del av undervisningsmaterialet för kursen ”Internets domännamnssystem” vid Kungliga tekniska högskolan, KTH (kurskod HI1037) resp. Karlstads universitet, KAU (kurskod DVGC28).

Licens

Dokumenthistorik

- Rev A: Ursprunglig version HT 2023

Slut.