



lacnic24
lacnog

28/9 - 2/10
bogotá, colombia

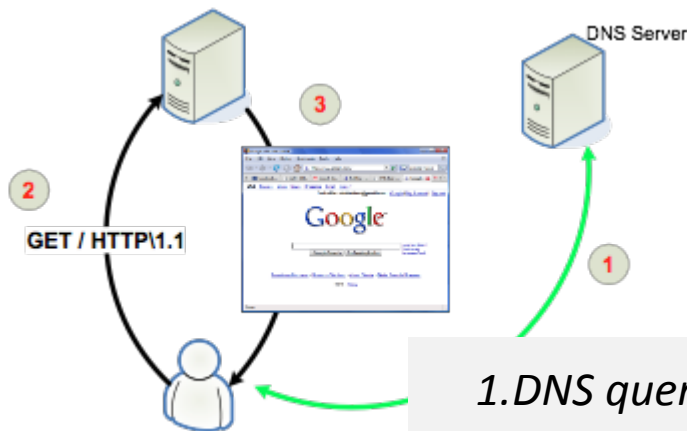
A Tutorial Introduction to DANE

Jan Zorz / ISOC

Carlos Martinez / LACNIC

Mechanics of Web Browsing

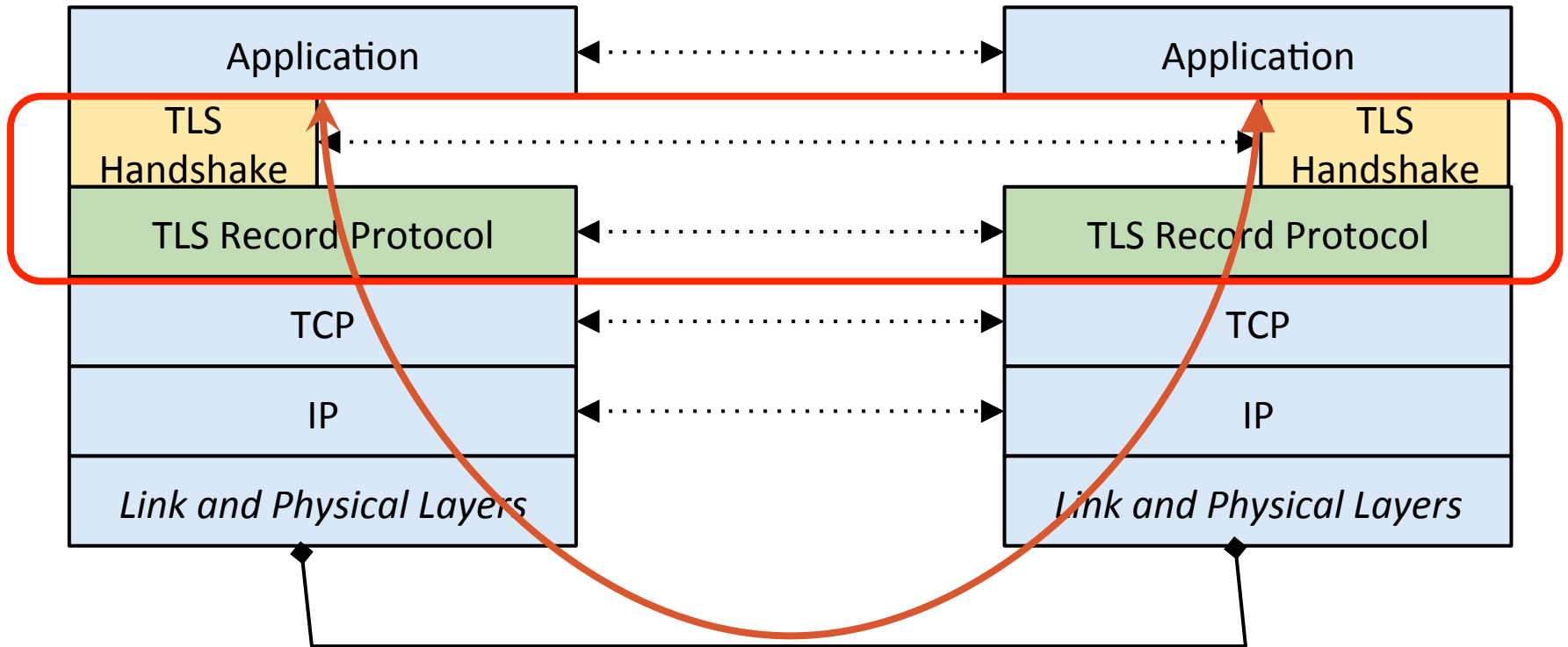
- Security?



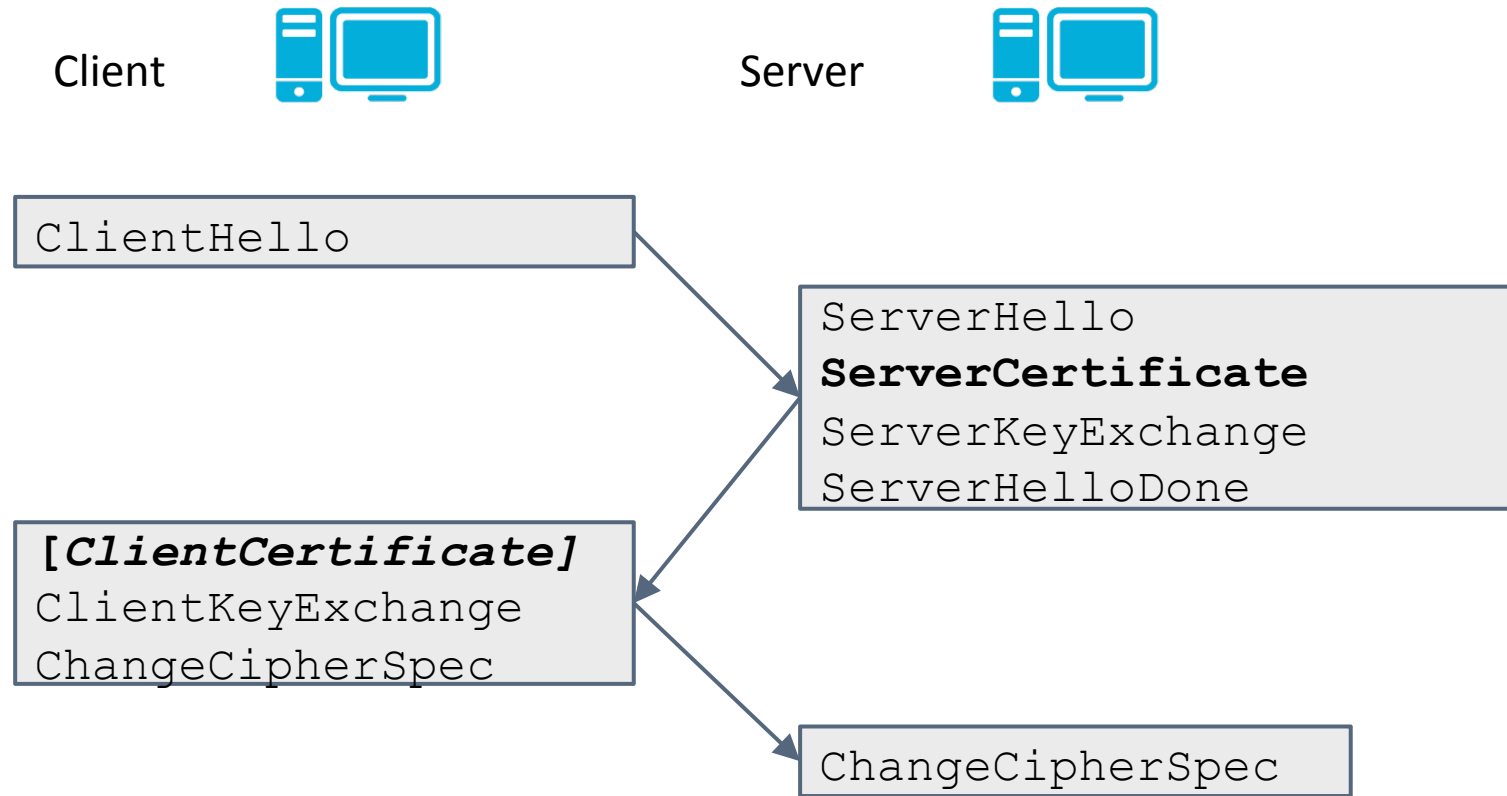
*Data flowing in **plain text**? That can be solved by **encrypting** the connection, right ?
Enter **TLS, Transport Layer Security***

1. DNS query for www.google.com
2. TCP connection to IP obtained in (1)
3. Data flows **in plain text**

Securing and Authenticating Endpoints



TLS Handshake



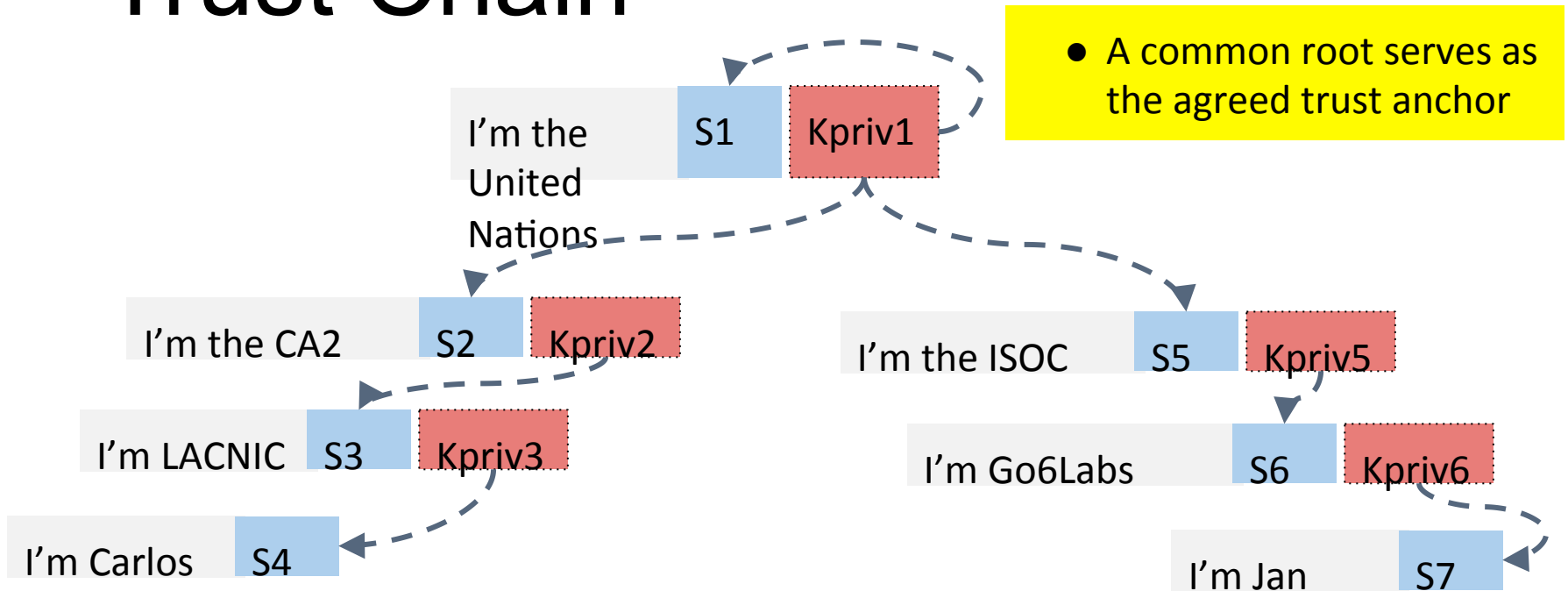
Digital Certificates

- A Public Key Certificate is a digital document that binds a set of information (fields) with a public key and is digitally signed.
- Signatures can be either be performed by a third party or by the issuer itself (self-signed certificates)
- Validation
 - Observers can verify the digital signature of the certificate
- Trust
 - Certificate Authority model
 - Signature verification is followed up a chain until reaching a commonly agreed trust anchor

Digital Certificates (2)

- Fields and flags in a certificate define how, where and when the certificate can be used and define is valid to be used
 - Valid-from, Valid-until times
 - Express constraints on usage
- Extensions
 - Lists of [type-value-critical_flag]
- Examples
 - Key usage
 - Extended key usage (clientAuth, serverAuth, emailProtection, ...)
 - RFC 3779 (Internet number resources)

Trust Chain



- Certs are **public**
- Private keys are **not published**, but held by their owners and used for signing when needed

Drawbacks of the CA-Based Chain

- Trust anchors can (and have been) successfully attacked
 - DigiNotar, GlobalSign, DigiCert Malaysia are just some examples
- The process that CAs use to validate information provided by customers can be subverted
- CAs are slow to react when a certificate is compromised
- The revocation process can be slow and is based on the concept of CRLs that have to be downloaded and are re-created every few hours
- [Check <https://tools.ietf.org/html/draft-housley-web-pki-problems-00>]

The DigiNotar Debacle

- [<https://www.enisa.europa.eu/media/news-items/operation-black-tulip>]



Operation Black Tulip: Certificate authorities lose authentication
DigiNotar, a digital certificate authority (CA), recently suffered a cyber-attack. In the attack false certificates were created for hundreds of websites, including the incident was made public, the Dutch government and browser vendors took part of the attack. But Fox-IT suggests in their [investigation report](#) that the cyber-attack occurred in mid-June and that for almost two months false certificates were used to eavesdrop on browsing in Iran. We see three major issues:

KIM ZETTER SECURITY 09.20.11 3:05 PM

DIGINOTAR FILES FOR BANKRUPTCY IN WAKE OF DEVASTATING HACK

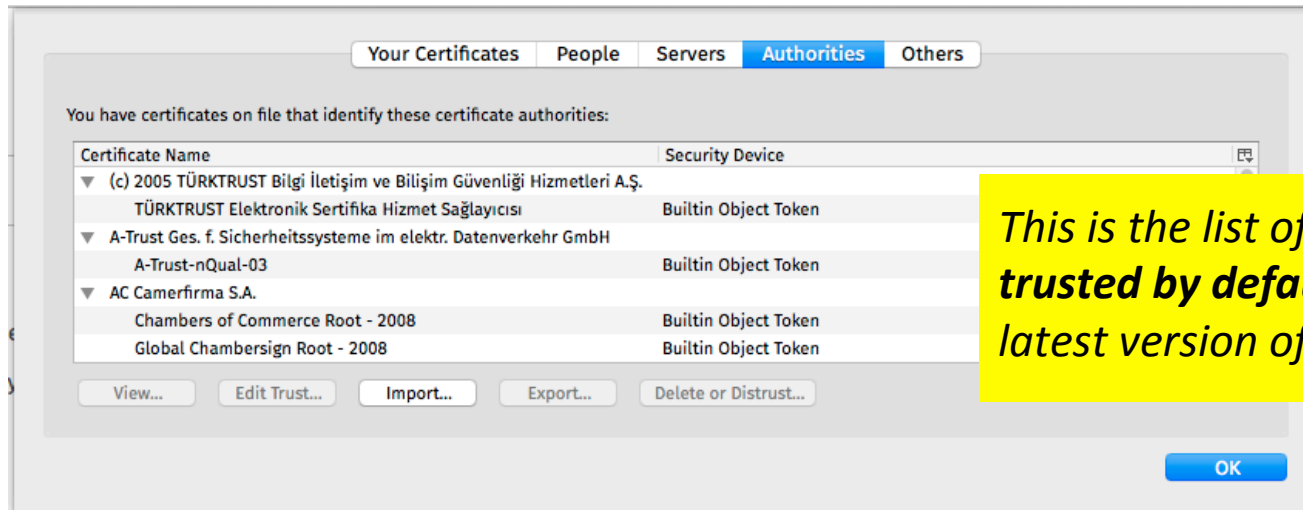
A Dutch certificate authority that suffered a major hack attack this summer has been unable to recover from the blow and filed for bankruptcy this week.

DigiNotar, which is owned by Illinois-based Vasco Data Security and was the primary provider of digital security certificates for domains



Shortcomings of the Traditional CA Model

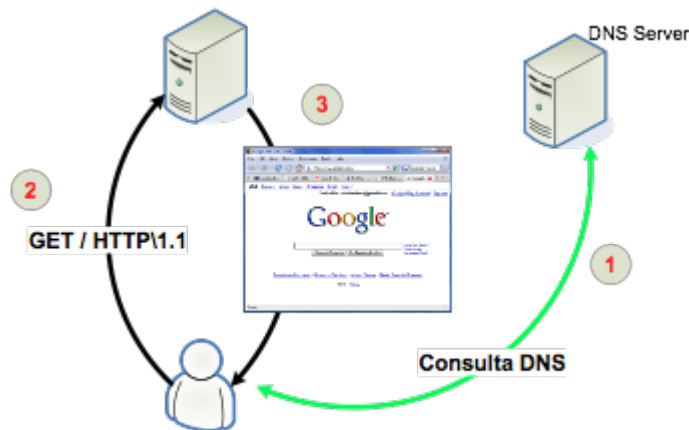
- The attack surface is huge and growing!
- A CA can sign for ANY domain, and for the browser it's enough to find one CA vouching for a given combination of domain and IP



This is the list of TAs trusted by default by the latest version of Firefox.

And there is one hole more...

- *Any web browsing starts with a DNS query*



*Even if all the certificates and SSL servers are configured perfectly, there is still **at least one insecure DNS query***

*Enabling **DNSSEC** for the server domain secures the query.*

*Without **DNSSEC** no connection is fully secured even if all certificates look fine.*

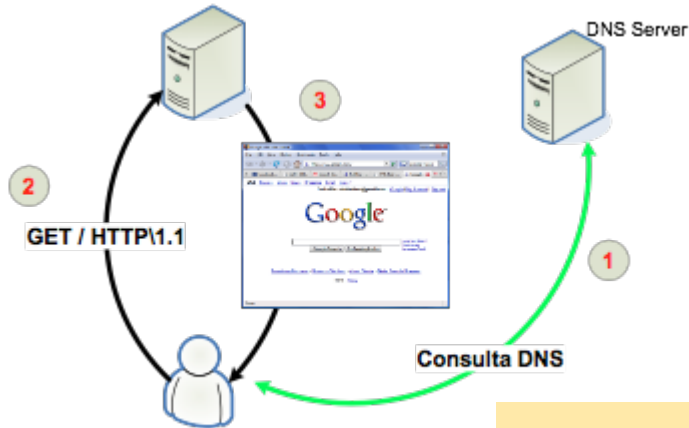
1. *DNS query for `www.google.com`*
2. *TCP connection to IP obtained in (1)*
3. *Hopefully, SSL handshake*
4. *Data flows*

Enter DANE

To Keep in Mind

- TLS secures communications, prevents eavesdropping, allows server identification
- When a client (C) connects to a TLS-protected server (S):
 - S presents C with a X.509 certificate
- C must check whether:
 - Does the certificate contain the correct server name?
 - Does the certificate contain the correct IP address?
 - Is the server certificate signed by a CA I trust ?

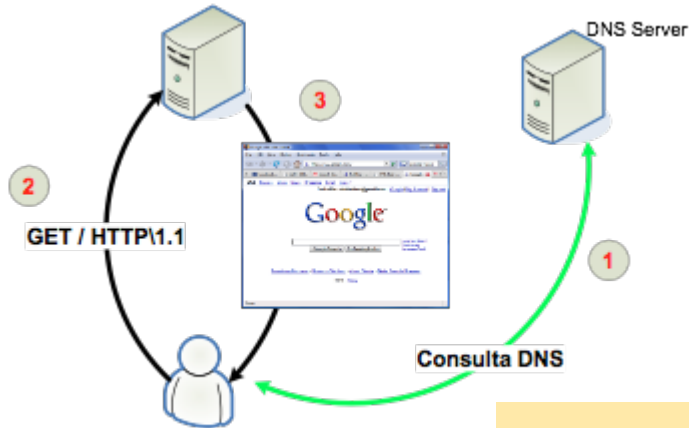
DANE – The TLSA DNS Record



From this point we assume all DNS zones are DNSSEC-signed.

What if... I could publish my digital certificates in the DNS itself ?

DANE – The TLSA DNS Record



From this point we assume all DNS zones are DNSSEC-signed.

```
; Zone example.com - Signed with DNSSEC
```

```
example.com IN SOA (...)  
IN NS ...  
IN DNSKEY ...
```

```
www.example.com. IN A 10.0.0.1
```

```
_443._tcp.www.example.com. IN TLSA ...
```

TLSA Record Overview

- The TLSA DNS record is our friend!
- Contains information binding keys or certificates to domain names and DNS zones
- Four fields:
 - Certificate usage field
 - Selector field
 - Matching type field
 - DATA

```
_443._tcp.www.example.com IN TLSA  
3 1 1 DATA
```

"3" - Certificate usage field

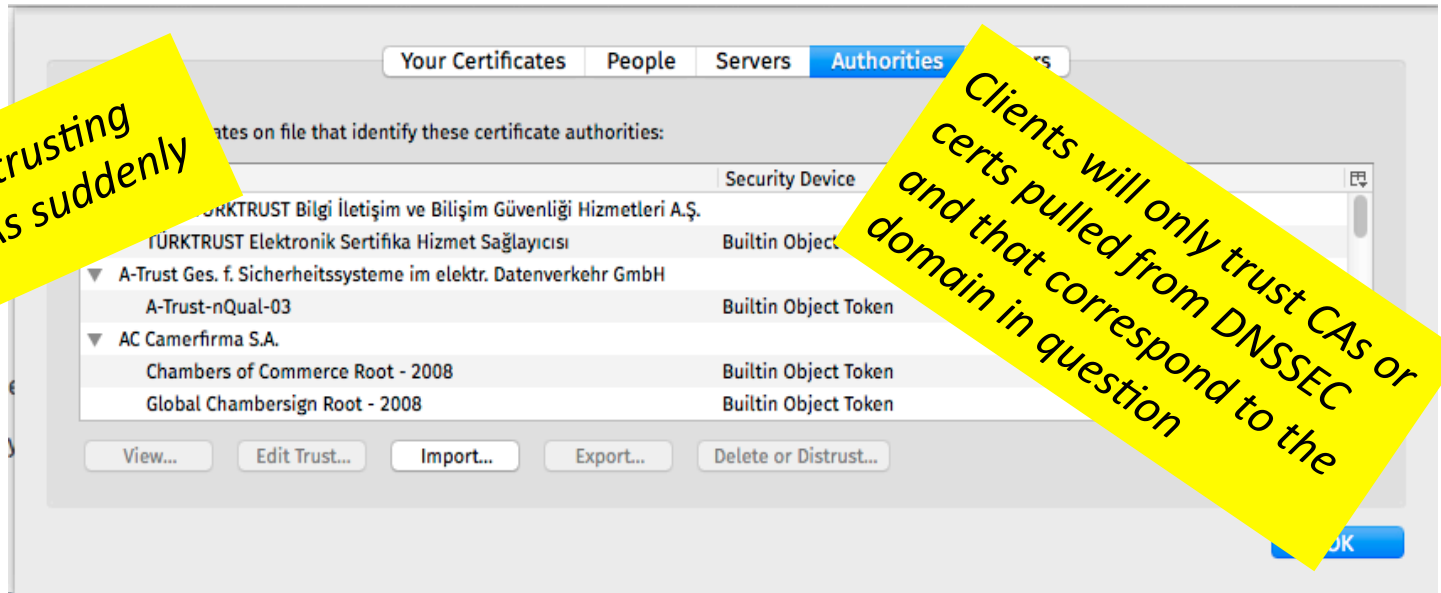
"1" - Selector field

"1" - Matching type field

DATA - *Depends on the values of the above*

DANE Use Cases

- Now the operator of a TLS-enabled server can:
 - publish a complete certificate on the DNS
 - refer in the DNS to a CA that can validate the certs within that domain

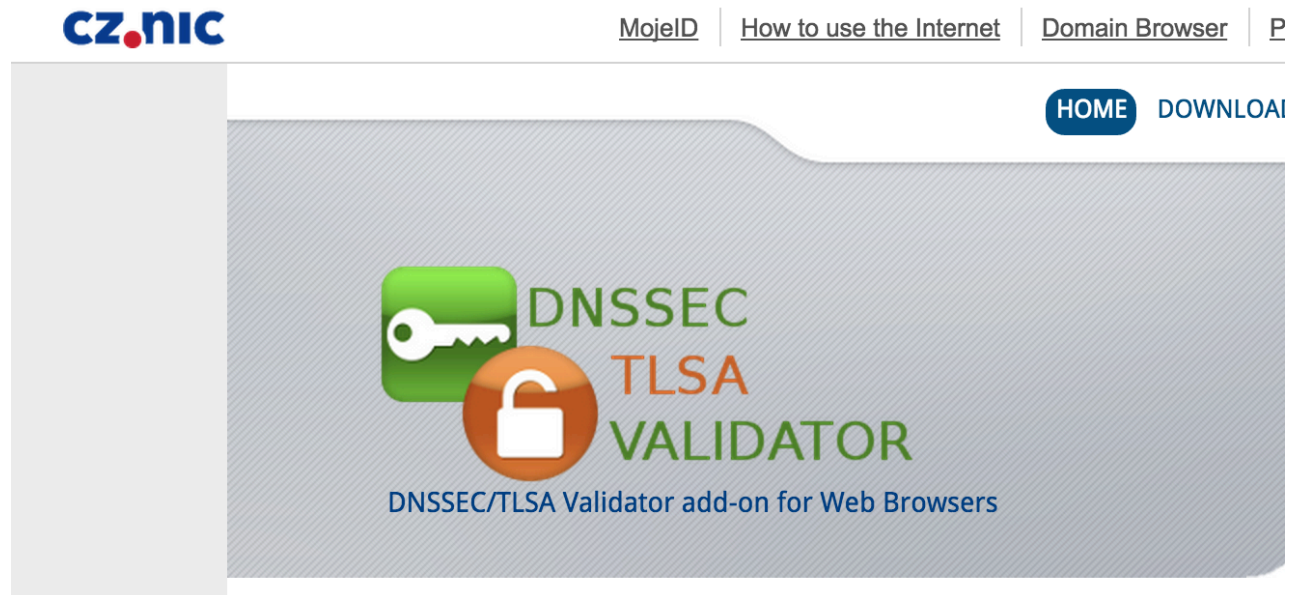


1-Slide DANE HOW-TO

- Sign your zone with DNSSEC
- Configure 'HTTPS' in your web server
 - Create a digital certificate yourself using OpenSSL
 - Configure Apache or your web server of choice
- Create TLSA records using Idns-dane
 - <http://www.nlnetlabs.nl/projects/ldns/>
 - There are other tools out there, I just found this one to be easy to use
- Add the TLSA records to your DNS zone and re-sign
- Wait for TTLs to expire.... et voilà!

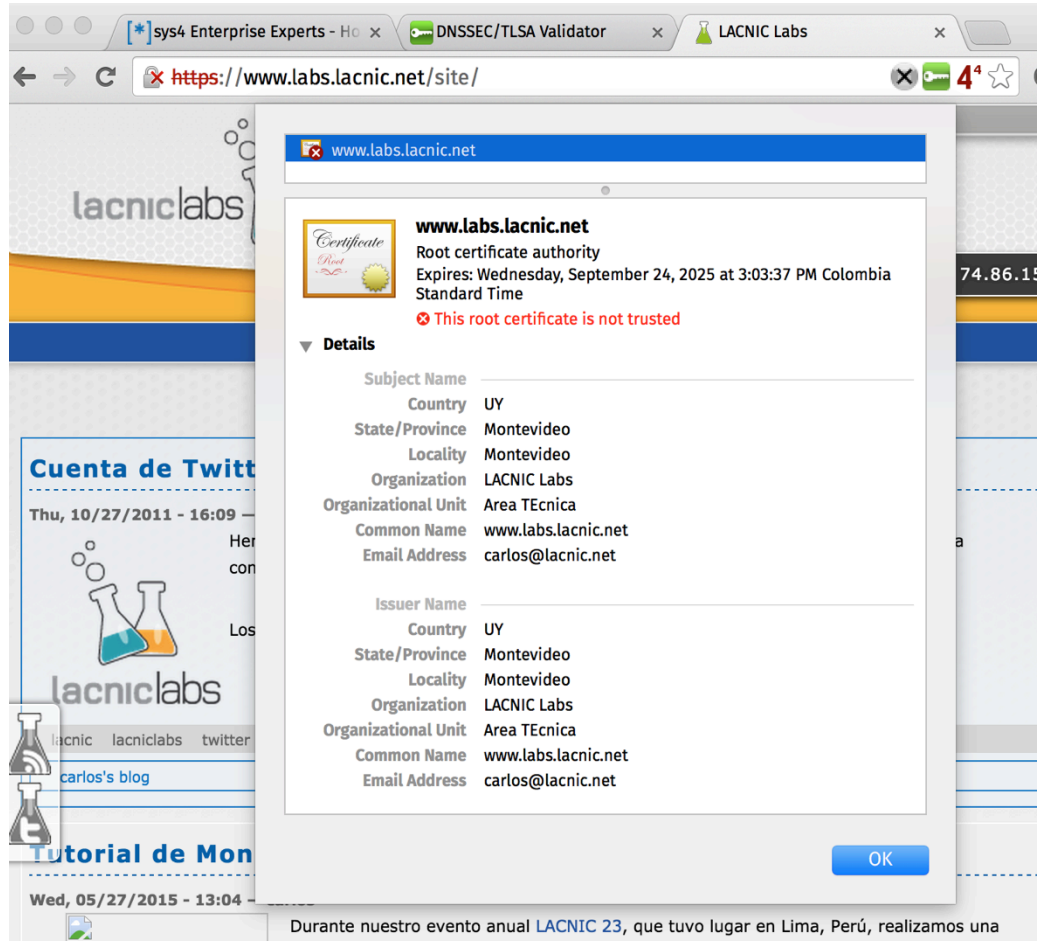
Browser Support Via Plugins

- CZ.NIC has implemented a nice set of plugins for validating https connections with DANE and for validating DNSSEC



LACNIC Labs Site Before DANE

- Certificate is not trusted
- It's not signed by any known CA



The screenshot shows a web browser window with the address bar displaying <https://www.labs.lacnic.net/site/>. A security warning dialog box is open, showing the following information:

www.labs.lacnic.net
Root certificate authority
Expires: Wednesday, September 24, 2025 at 3:03:37 PM Colombia Standard Time
✗ This root certificate is not trusted

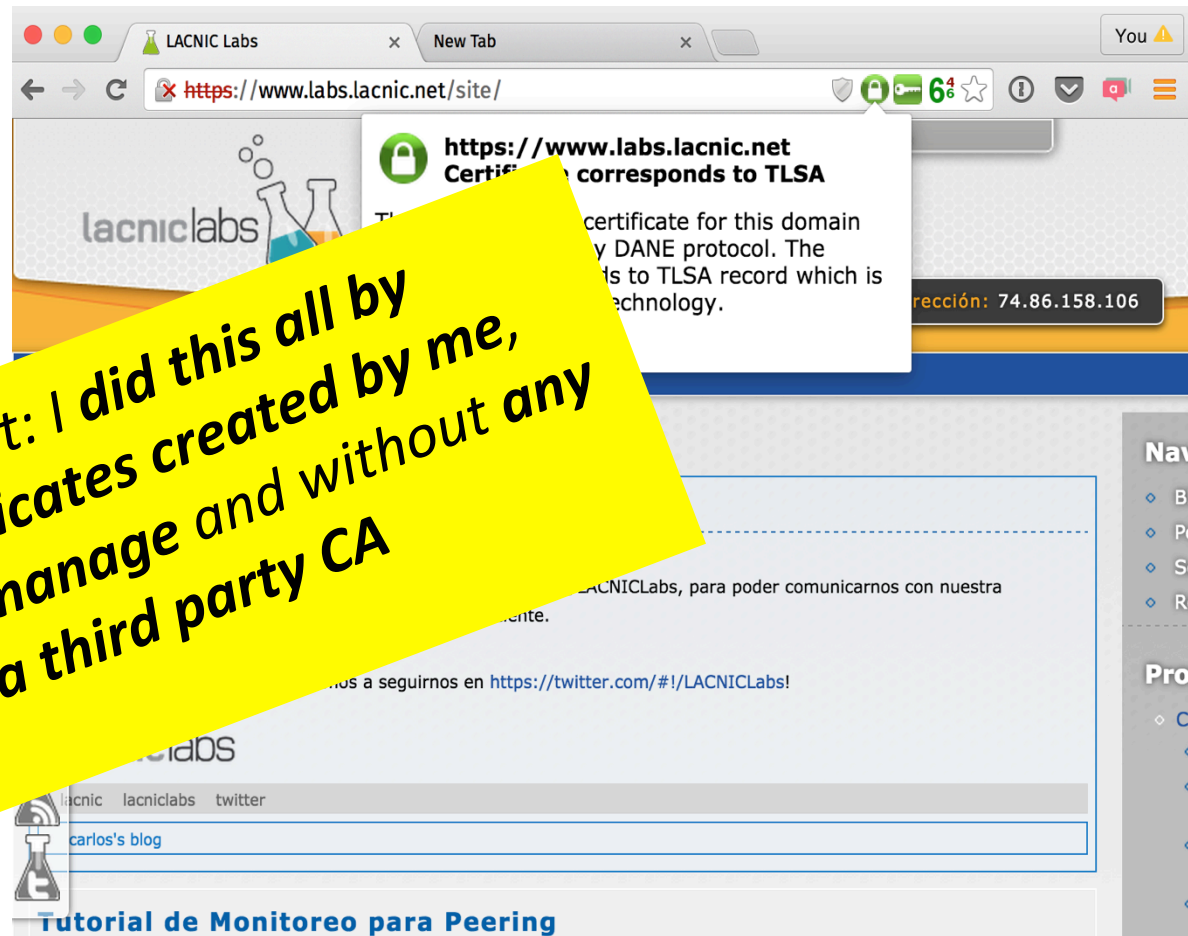
Details

Subject Name	
Country	UY
State/Province	Montevideo
Locality	Montevideo
Organization	LACNIC Labs
Organizational Unit	Area Técnica
Common Name	www.labs.lacnic.net
Email Address	carlos@lacnic.net
Issuer Name	
Country	UY
State/Province	Montevideo
Locality	Montevideo
Organization	LACNIC Labs
Organizational Unit	Area Técnica
Common Name	www.labs.lacnic.net
Email Address	carlos@lacnic.net

The background of the browser shows the LACNIC Labs website header with the logo and a Twitter account post for 'Cuenta de Twitter' dated 'Thu, 10/27/2011 - 16:09'. The bottom of the browser window shows a footer with the text: 'Durante nuestro evento anual LACNIC 23, que tuvo lugar en Lima, Perú, realizamos una'.

LACNIC Labs After DANE

- Validated!



Drawbacks ? Sure...

- There is a bit of a learning curve
- Browser support, still in its infancy
- Application support in general
- Dependent on DNSSEC adoption

Thanks and over to
Jan!

DANE/DNSSEC/TLS Testing in the Go6lab

Jan Žorž, ISOC/Go6 Institute, Slovenia

jan@go6.si

zorz@isoc.org

Acknowledgement

I would like to thank Internet Society to let me spend some of my ISOC working time in go6lab and test all this new and exciting protocols and mechanisms that makes Internet a bit better and more secure place...

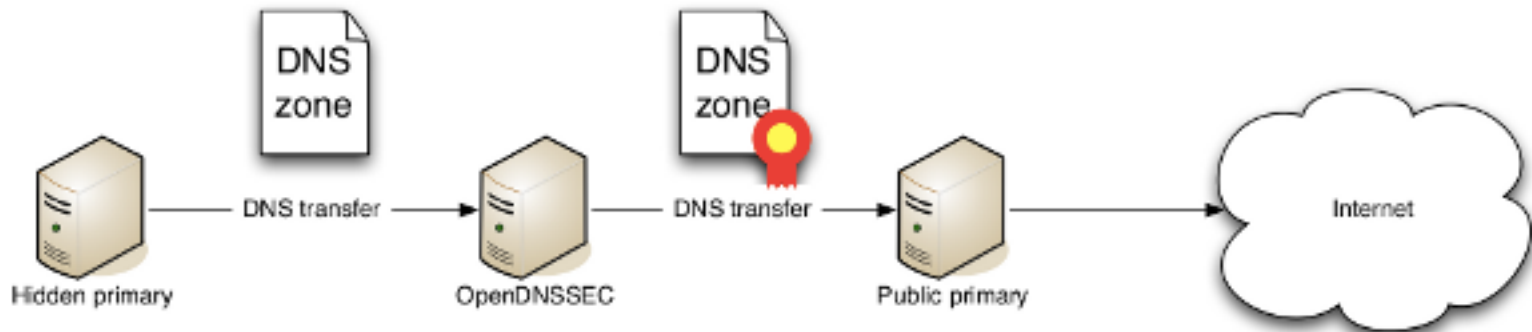


DNSSEC implementation in go6lab

- Powerdns server (used as primary for non-signed domains) as “hidden” primary DNS server
- OpenDNSSEC platform for signing domains
- BIND9 DNS servers as secondaries to OpenDNSSEC to serve signed zones
- Virtualization used: PROXMOX 3.4
- OS templates: fedora-20, Centos6/7

DNSSEC implementation in go6lab

- “Bump in a wire”
- Two public “primary” servers
- Concept:



DNSSEC in go6lab

- That was fairly easy and it works very well.
- Implementation document used from Matthijs Mekking:

<http://go6.si/docs/opendnssec-start-guide-draft.pdf>

DANE experiment

- When DNSSEC was set up and functioning we started to experiment with DANE (DNS Authenticated Name Entities).
- Requirements:
 - DNSSEC signed domains
 - Postfix server with TLS support > 2.11
- We decided on Postfix 3.0.1

DANE

- TLSA record for mx.go6lab.si

```
_25. tcp.mx.go6lab.si. IN  TLSA  3 0 1  
B4B7A46F9F0DFEA0151C2E07A5AD7908F4C8B0050E7CC  
25908DA05E2 A84748ED
```

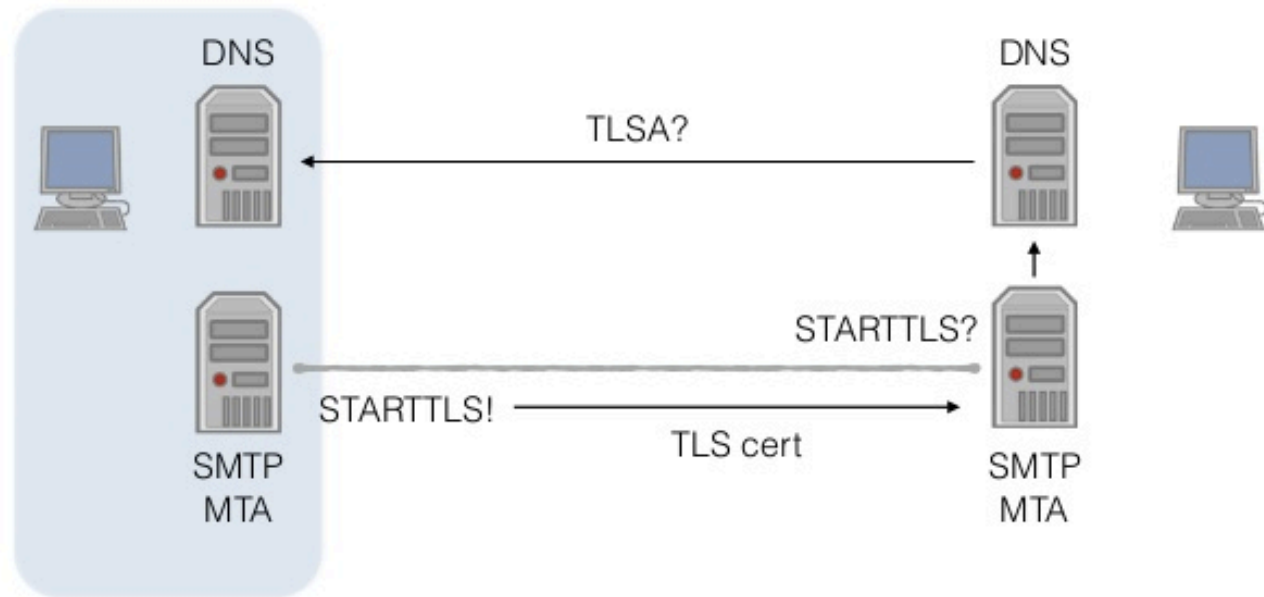
It's basically a hash of TLS certificate on mx.go6lab.si

More about DANE:

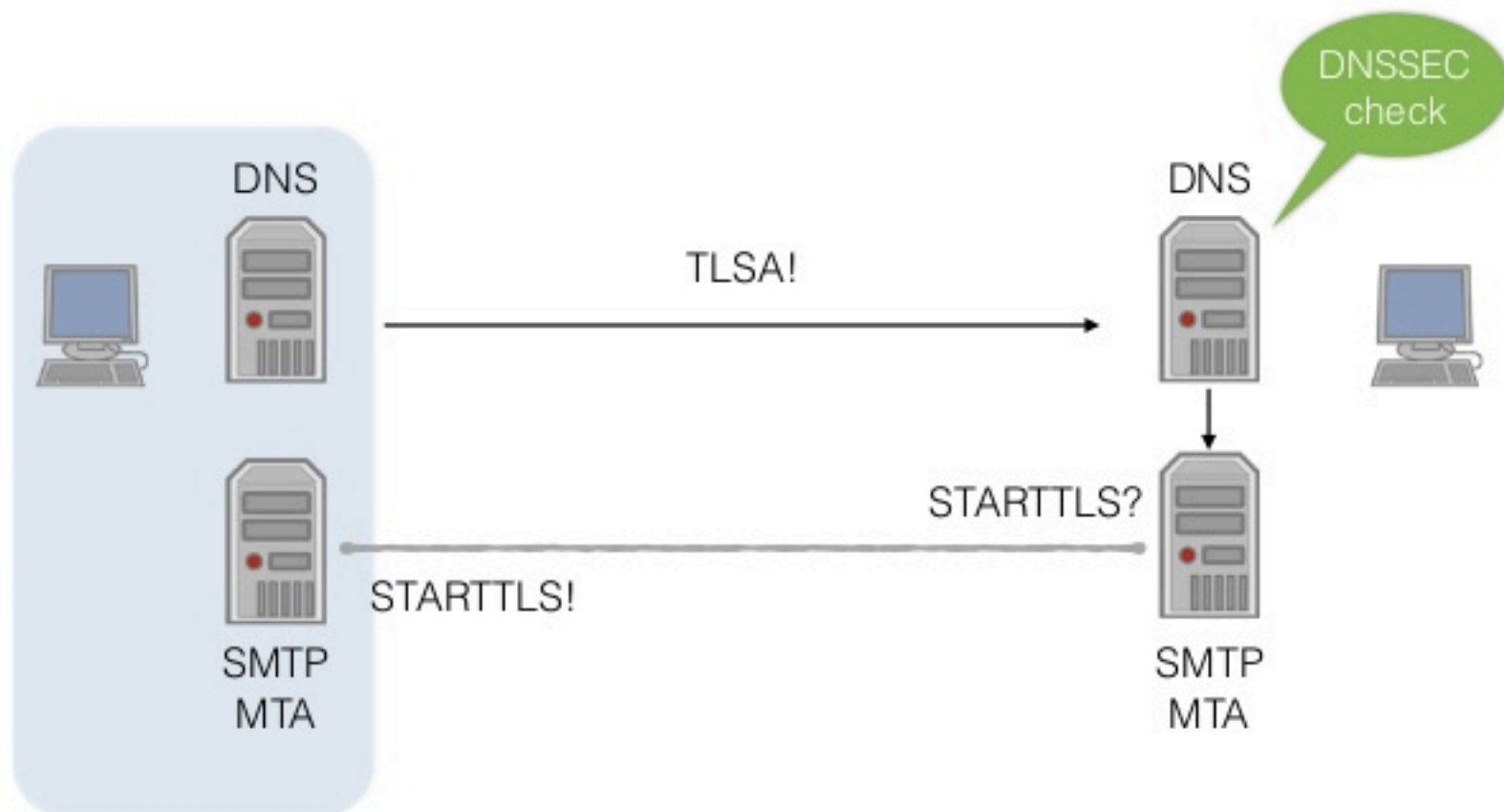
<http://www.internetsociety.org/deploy360/resources/dane/>

What is DANE and how does it work

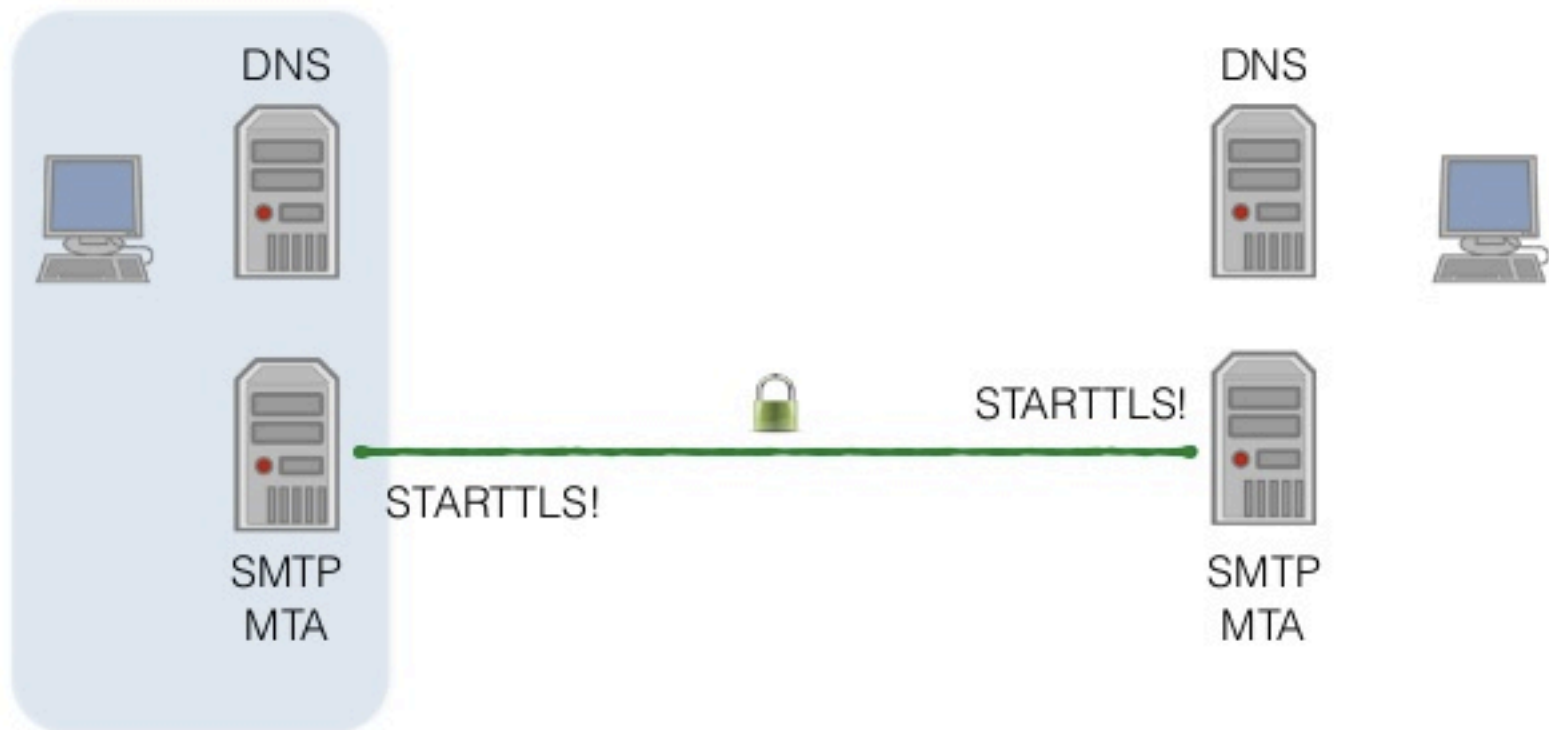
TLS and SMTP



TLS and SMTP



TLS and SMTP



DANE verification

- Mx.go6lab.si was able to verify TLS cert to T-2 mail server and nlnet-labs and some others...

mx postfix/smtp[31332]: Verified TLS connection established to smtp-good-in-2.t-2.si[2a01:260:1:4::24]:25: TLSv1 with cipher DHE-RSA-AES256-SHA (256/256 bits)

dicht postfix/smtp[29540]: Verified TLS connection established to mx.go6lab.si[2001:67c:27e4::23]:25: TLSv1.2 with cipher ECDHE-RSA-AES256-GCM-SHA384 (256/256 bits)

Postfix config

```
smtpd_use_tls = yes
smtpd_tls_security_level = may
smtpd_tls_key_file = /etc/postfix/ssl/server.pem
smtpd_tls_cert_file = /etc/postfix/ssl/server.pem
smtpd_tls_auth_only = no
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
smtp_tls_security_level = dane
smtp_use_tls = yes
smtp_tls_note_starttls_offer = yes
smtp_tls_loglevel = 1
tls_random_exchange_name = /var/run/prng_exch
tls_random_source = dev:/dev/urandom
tls_smtp_use_tls = yes
```

Malformed TLSA record

- We created a TLSA record with a bad hash (one character changed)
- Postfix failed to verify it and refused to send a message

```
mx postfix/smtp[1765]: Untrusted TLS connection  
established to
```

```
mail-bad.go6lab.si[2001:67c:27e4::beee]:25: TLSv1.2  
with
```

```
cipher ECDHE-RSA-AES256-GCM-SHA384 (256/256 bits)
```

```
mx postfix/smtp[1765]: 3A4BE8EE5C: Server certificate  
not trusted
```

1M top Alexa domains and DANE

- We fetched top 1 million Alexa domains and created a script that sent an email to each of them (test-dnssec-dane@[domain])
- After some tweaking of the script we got some good results
- Then we built a script that parsed mail log file and here are the results:

Results

- Out of 1 million domains, 992,232 of them had MX record and mail server.
- Nearly 70% (687,897) of all attempted SMTP sessions to Alexa top 1 million domains MX records were encrypted with TLS
- Majority of TLS connections (60%) were established with trusted certificate
- 1,382 connections where remote mail server announced TLS capability failed with "Cannot start TLS: handshake failure"

More results

TLS established connections ratios are:

Anonymous: 109.753

Untrusted: 167.063

Trusted: 410.953

Verified: 128

Quick guide: Anonymous (opportunistic TLS with no signature), Untrusted (peer certificate not signed by trusted CA), Trusted (peer certificate signed by trusted CA) and Verified (verified with TLSA by DANE).

DANE Verified

Verified: 128 !!!

Mail distribution

Mail Servers	# Domains Handled	TLS State
google.com	125,422	Trusted
secureserver.net	35,759	Some Trusted, some no TLS at all
qq.com	11,254	No TLS
Yandex.ru	9,268	Trusted
Ovh.net	8,531	Most Trusted, with redirect servers having no TLS at all

Mail distribution

Mail Servers	# Domains Handled	TLS State
Emailsrvr.com	8,262	Trusted
Zohomail.com	2,981	Trusted
Lolipop.jp	1,685	No TLS
Kundenserver.de	2,834	Trusted
Gandi.net	2,200	Anonymous

DNSSEC? DANE?

None of these “big” mail servers (and their domains) are DNSSEC signed (that means no DANE for them possible).

When do DANE things fail?

- Of course, with wrong certificate hash in TLSA record (refuses to send mail)
- If domain where MX record resides is not DNSSEC signed (can't trust the data in MX, so no verification)
- If TLSA record published in non-DNSSEC zone (can't trust the data in TLSA, so no verification)

When do things fail? (example)

- go6lab.si zone is signed, so is mx.go6lab.si
- there is TLSA for mx.go6lab.si, also signed
- Domain signed.si is signed and MX points to mx.go6lab.si
- Domain not-signed.si is not signed and MX points to mx.go6lab.si
- We send email to jan@signed.si and jan@not-signed.si (signed.si and not-signed.si are used just as examples)

When do things fail? (example)

When I send email to jan@signed.si (signed domain):

Verified TLS connection established to
mx.go6lab.si[2001:67c:27e4::23]:25:

When I send email to jan@not-signed.si (not signed domain):

Anonymous TLS connection established to
mx.go6lab.si[2001:67c:27e4::23]:25:

When do DANE verification also fail?

- Let's try to point MX record from signed domain to A/AAAA record in not-signed domain with TLSA that is also not signed (obviously) – mail.not-signed.si

Send mail to jan@signed.si when MX for [signed.si](#) points to [mail.not-signed.si](#) – DANE verification is not even started as chain of trust is broken

When do DANE verification also fail?

- Let's try to point MX record from signed domain to A/AAAA record in not-signed domain with **malformed** TLSA that is also not signed (obviously)
– mail.not-signed.si

Send mail to jan@signed.si when MX for [signed.si](#) points to [mail-bad.not-signed.si](#) – DANE verification is not even started as chain of trust is broken and even if there is TLSA record with a hash that does not match the offered TLS cert hash – mail is sent anyway.

Conclusions

- 70% of email can be encrypted in some way, you just need to enable TLS on your server
- Low number of DNSSEC signed domains/servers
- Even lower number of DANE/TLSA verified servers/connections
- It's easy, go and do it – it's not the end of the world and it helps with verifying who are you sending emails to – and vice versa ;)

Conclusions II.

- DANE verification fails (or is aborted) if DNSSEC chain of trust is not fully established and complete along the whole way.
- TLSA in not-signed DNS zones would not help you much preventing your correspondents sending emails to server-in-the-middle
- DNSSEC/DANE is easy, but please understand what are you doing before implementing it in production...

Q&A

Questions? Protests?
Suggestions? Complaints?

jan@go6.si

zorz@isoc.org