Dutch Internet Standards Platform

DANE for SMTP

ir. Dennis Baaten CISSP

11-11-2021

Internet.nl
IS JOUW INTERNET UP-TO-DATE?

Who are we?

- **Dutch Internet Standards Platform** is the organization behind the test tool Internet.nl.
- Collaboration between parties from the Internet community and the Dutch government (public / private).
- Our goal is to stimulate the adoption of modern internet standards.

NLNETLABS

Ministry of Economic Affairs and Climate Policy

Internet Society

DNPA Dutch Hosting Provider Association

SIDN

RIPE NCC RIPE NETWORK COORDINATION CENTRE

nlnet FOUNDATION

Dutch Standardisation Forum

ECP

ISPConnect

VVR VERENIGING VAN REGISTRARS

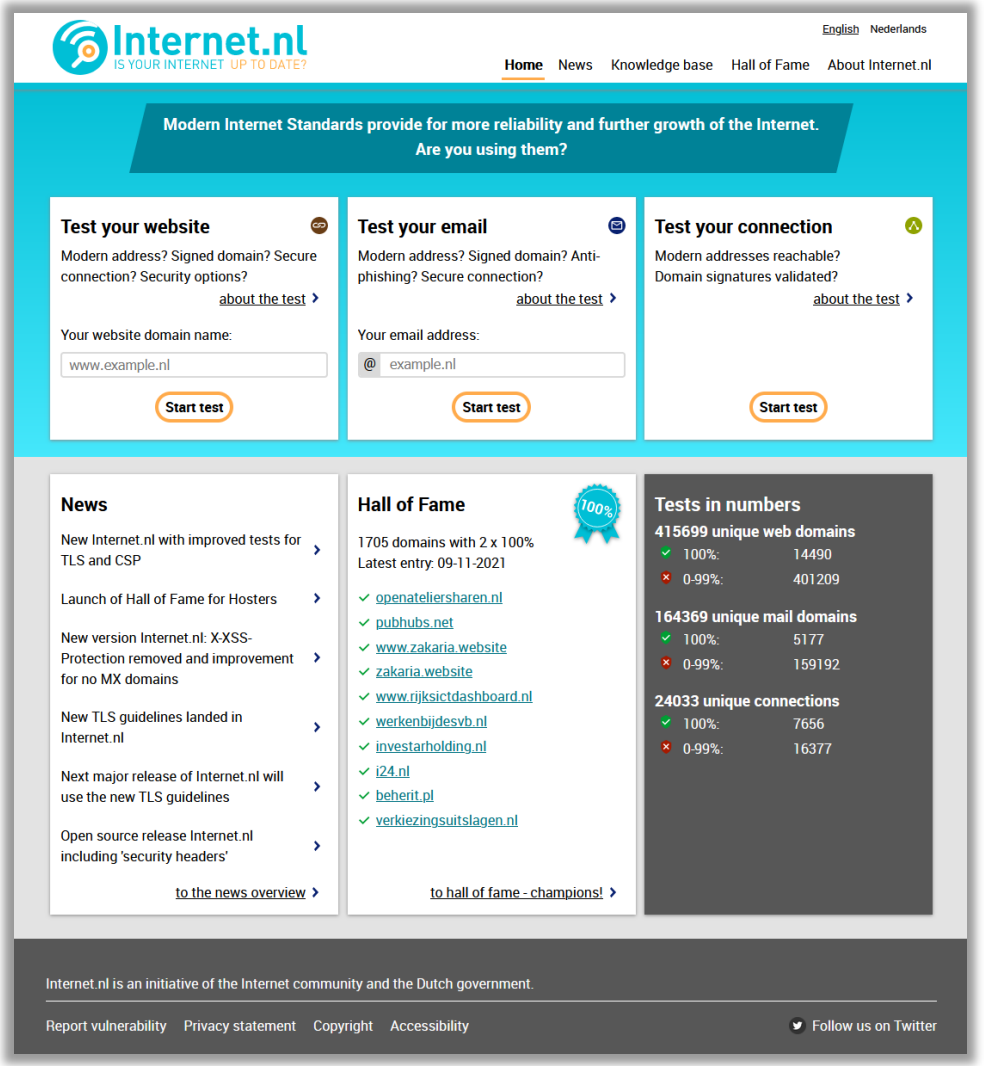
National Cyber Security Centre
Ministry of Justice and Security

SURF NET

Stimulating adoption of standards

We do a lot of things to stimulate the adoption of standards. The most important and visible things are:

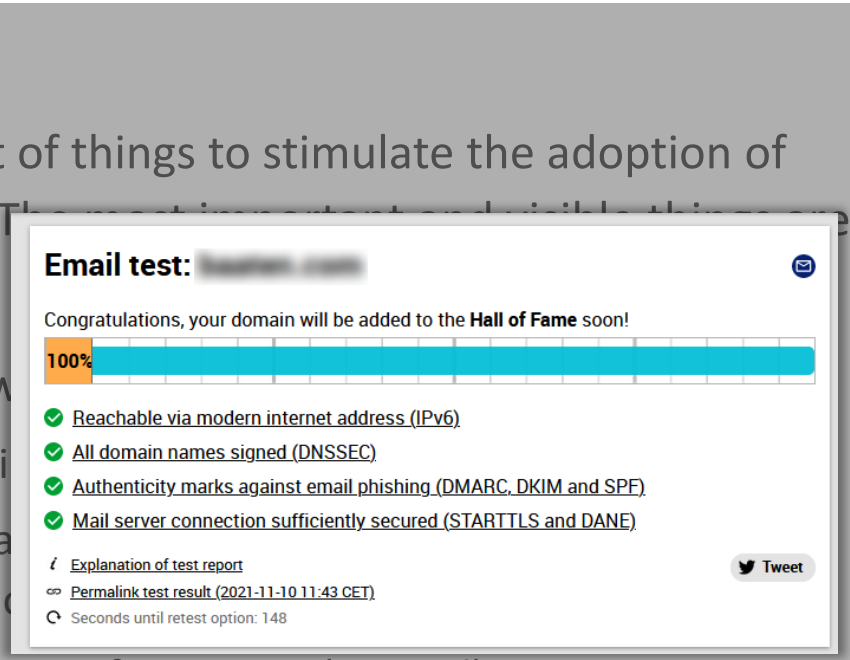
- It starts with our tool on https://internet.nl
  - Create insight and measure. Over 2000 tests per day.
  - Automatically generated Hall of Fame (web, email, champions).
  - Hall of Fame for Hosters (manual).
  - Our test norm is based on:
    - Internet Standards on the 'comply-or-explain' list of the Dutch Standardisation Forum.
    - Security advices of the Dutch NCSC.
    - Relevant RFC's of IETF.

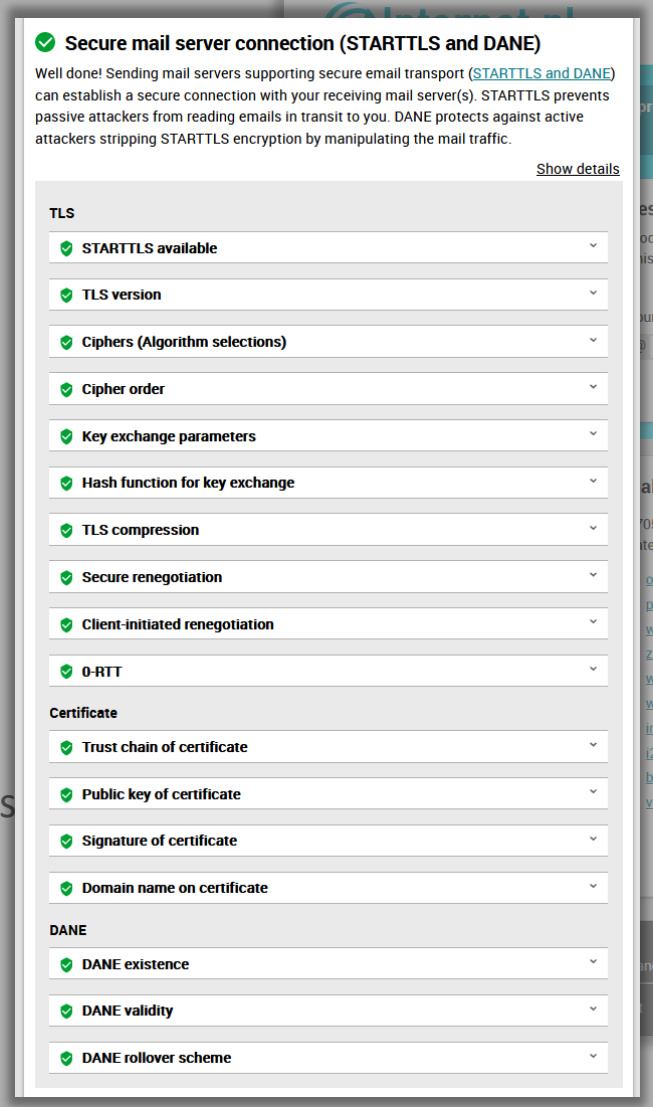


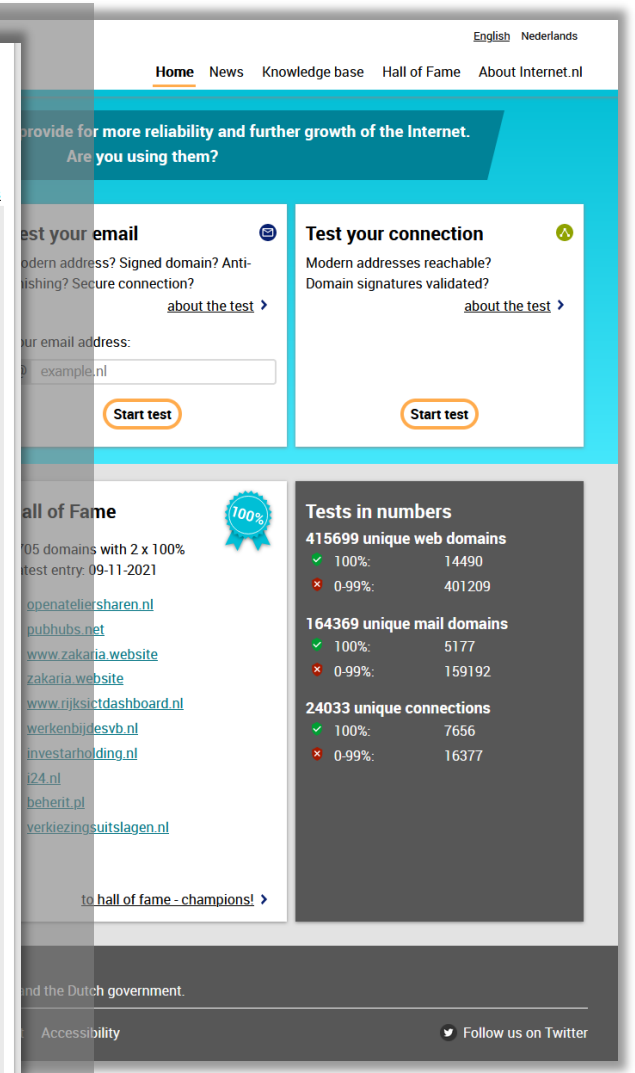
Stimulating adoption of standards

We do a lot of things to stimulate the adoption of standards. T[illegible]:

- It starts w[illegible]
  - Create i[illegible]
  - Automa[illegible] champi[illegible]
  - Hall of Fame for Hosters (manual).
  - Our test norm is based on:
    - Internet Standards on the 'comply-or-explain' lis[illegible] the Dutch Standardisation Forum.
    - Security advices of the Dutch NCSC.
    - Relevant RFC's of IETF.



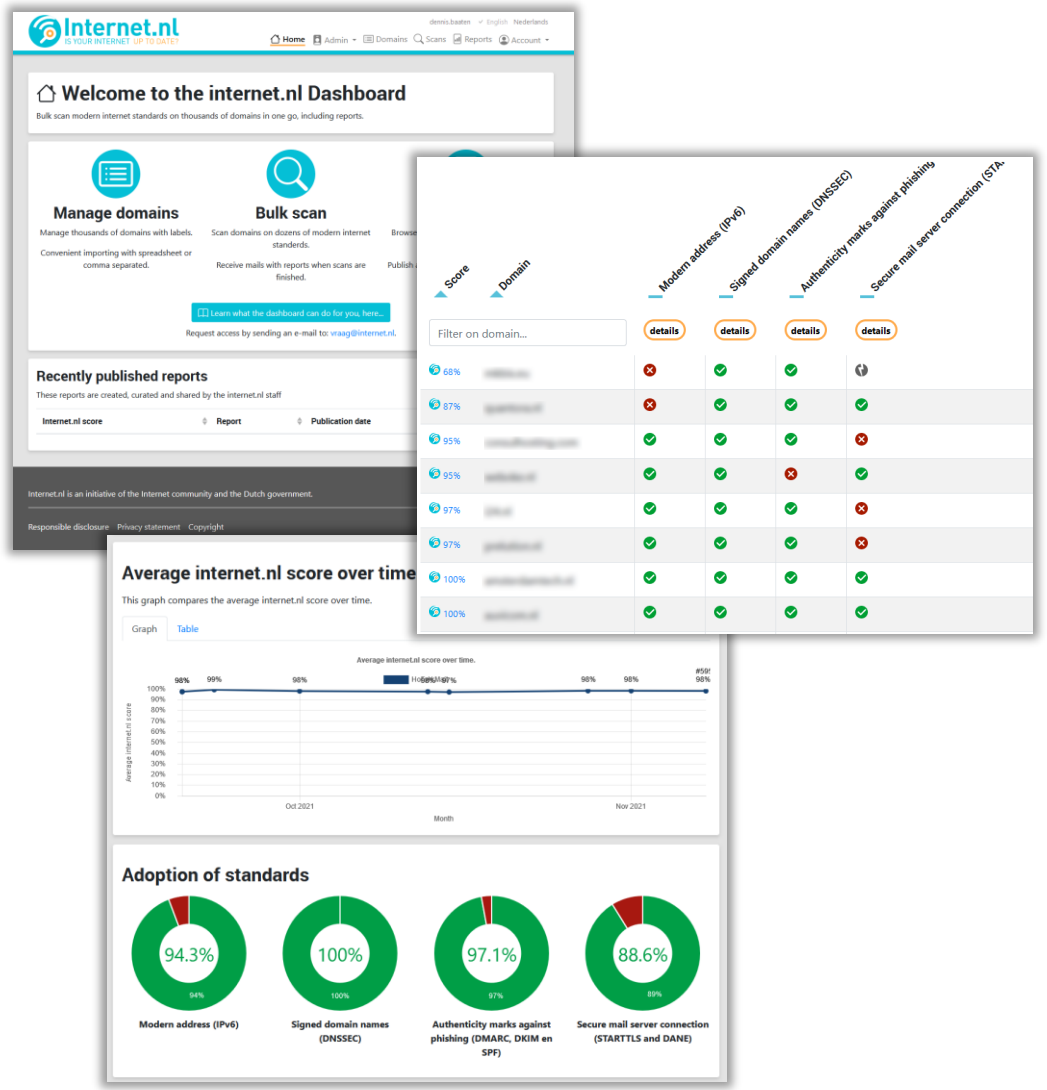




Stimulating adoption of standards

We do a lot of things to stimulate the adoption of standards. The most important and visible things are:

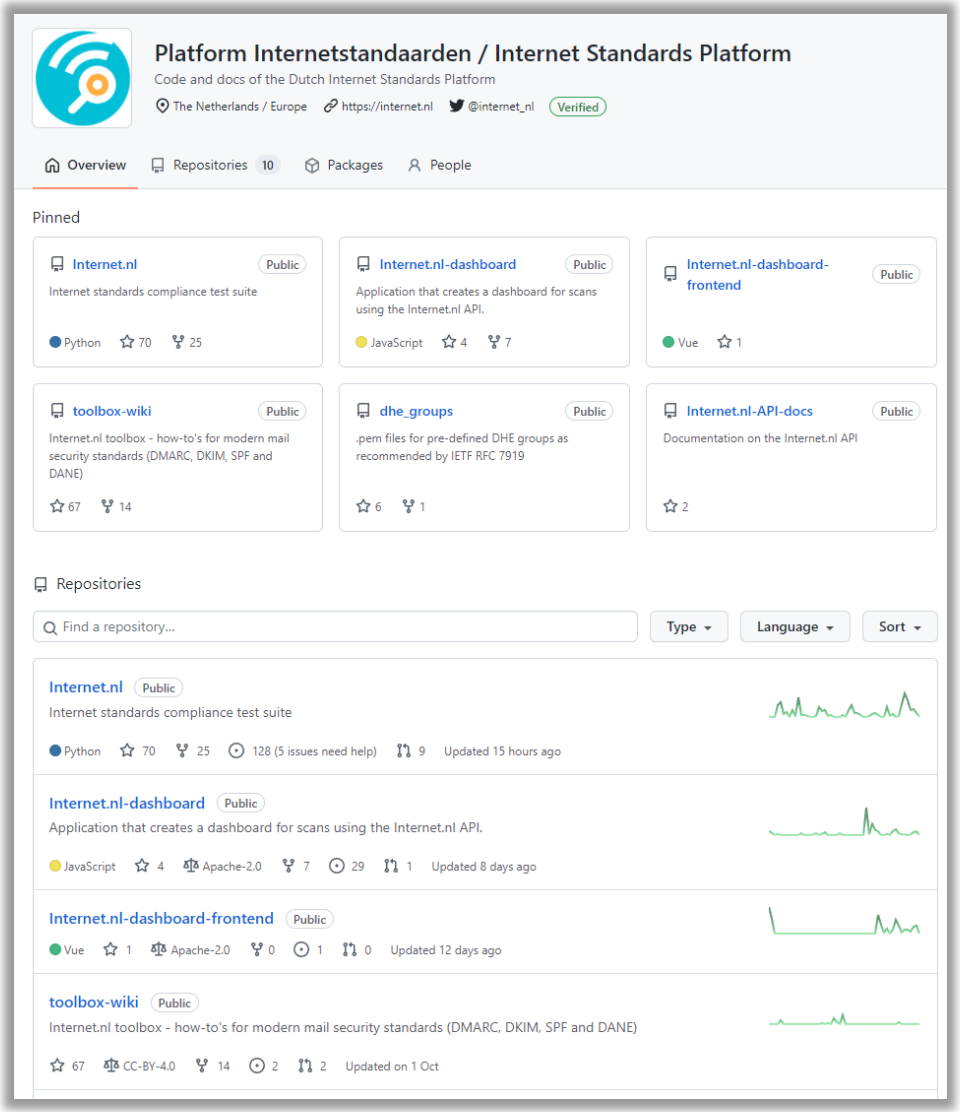
- API for bulk testing
  - JSON based REST-like API
  - OpenAPI specification: http://redocly.github.io/redoc/?url=https://batch.internet.nl/api/batch/openapi.yaml
  - Available for:
    - Platform members.
    - Government- and not-for-profit organizations.
    - Members of the Dutch Cloud Community (https://dutchcloudcommunity.nl/).
    - Members of VvR (https://www.verenigingvanregistrars.nl/).
- Dashboard (a GUI for the API).
  - Tracking changes over time.
  - Adoption statistics.



Stimulating adoption of standards

We do a lot of things to stimulate the adoption of standards. The most important and visible things are:

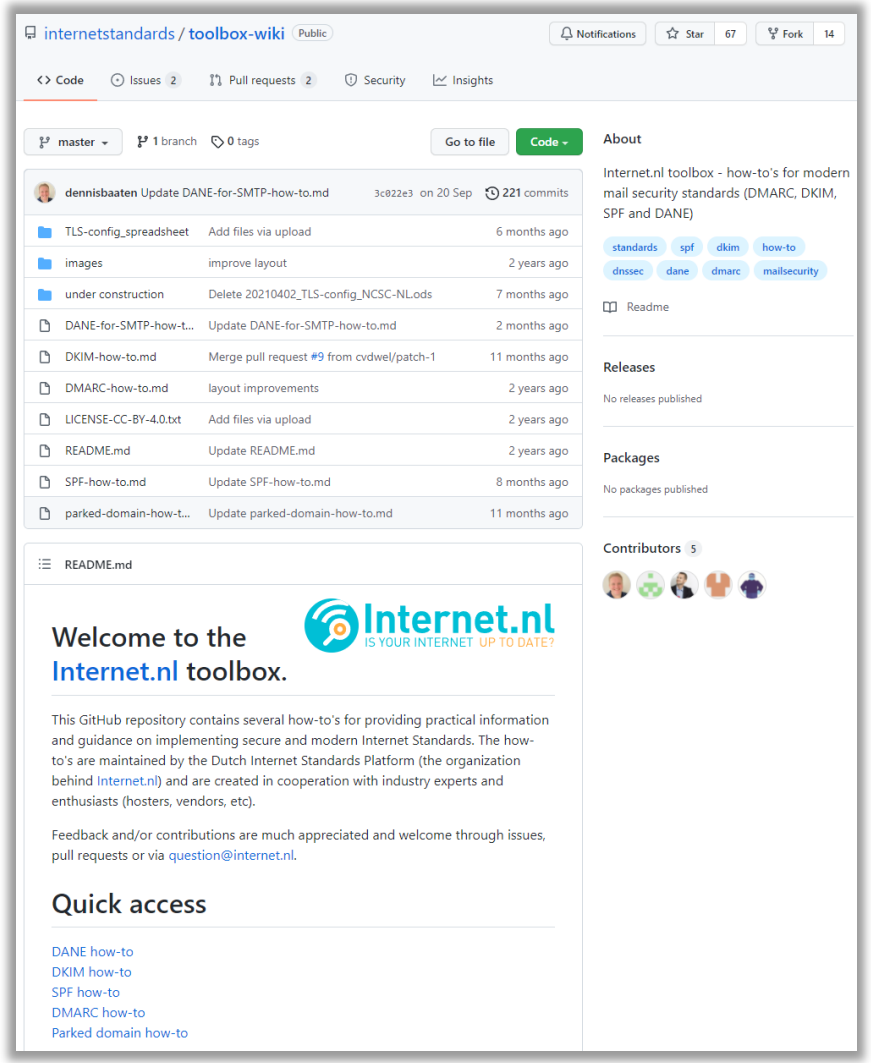
- All tools are open source (GitHub)
  - https://github.com/internetstandards
  - Used by several other countries, like the Danish: https://sikkerpånettet.dk/.
- Active on social media
  - https://mobile.twitter.com/internet_nl.
  - https://www.linkedin.com/company/internet-nl/.



Stimulating adoption of standards

We do a lot of things to stimulate the adoption of standards. The most important and visible things are:

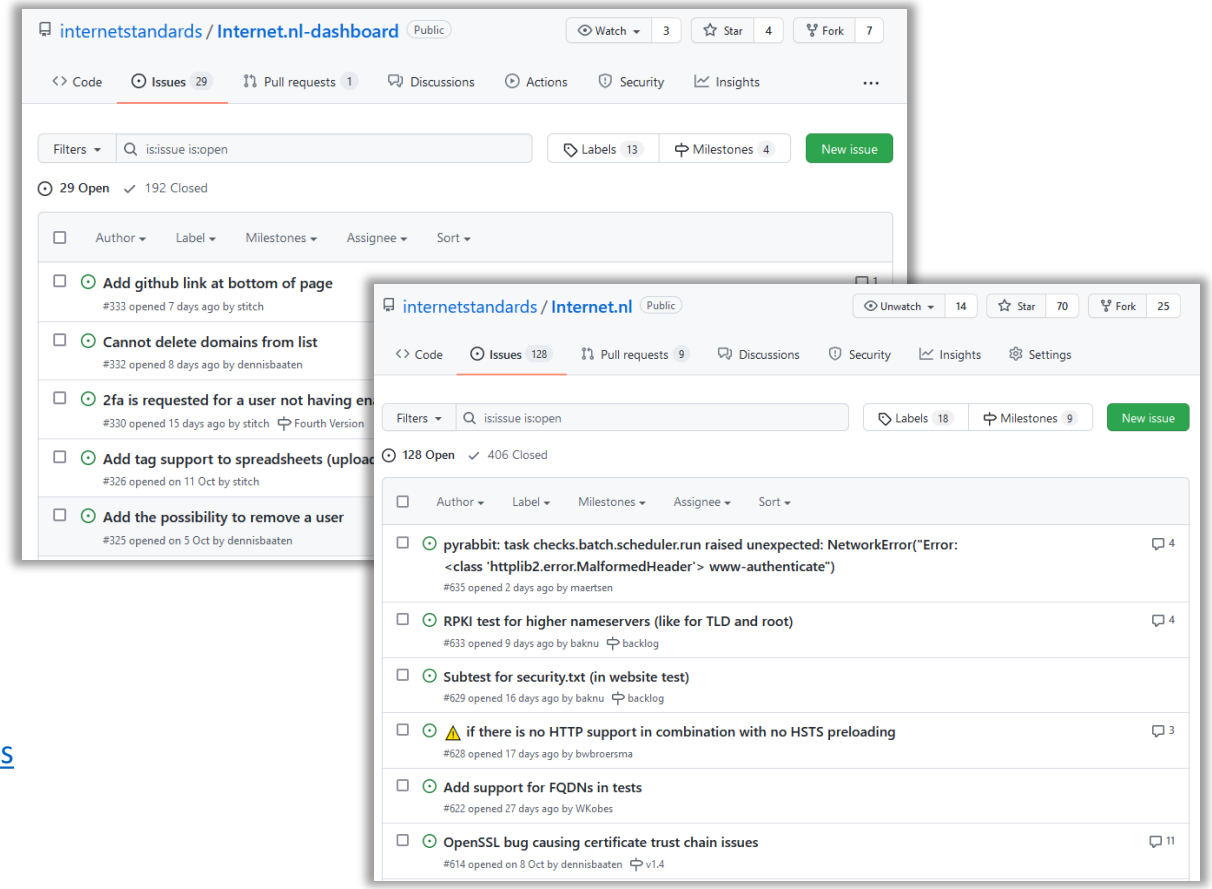
- Create how-to's for making implementation of standards as easy a possible
  - https://toolbox.internet.nl (redirect to GitHub repo)
  - A continuous process: how-to's are 'living documents' and never finished.
  - Lesson learned: difficult to get people to contribute.
  - Feel free to contribute!



What we are working on

We are constantly seeking for ways to improve our platform and increase our value for our users.

- Current/recent:
  - Minor improvements and bugfixes.
  - Improving user feedback (test results).
  - Maintain / create how-to’s.
  - New version (version 4) of our Dashboard
    - Public reports
    - Working with tags for domain names (great for filtering results)
- New:
  - Looking into feasibility of an ‘accessibility test’.
  - Implementing RPKI check (increased BGP security).
  - Interactive mail test for our API environment.
  - See also: https://github.com/internetstandards/Internet.nl/milestones



One of the standards we are actively stimulating is **DANE**.

So… let’s look at DANE.

Why DANE?

- Our (public) email ecosystem is flawed by default.
- SMTP on its own cannot ensure integrity and confidentiality of email delivery.
- Several measures can and have been taken to improve, but are still not enough.
- Regarding email transport, DANE is the next and final step.

What is DANE?

DNS-based **A**uthentication of **N**amed **E**ntities

RFC 6698 & RFC 7671

> [*DNS-Based Authentication of Named Entities (DANE) offers the option to use the DNSSEC infrastructure to store and sign keys and certificates that are used by TLS.*]

In other words:

- Verifying certificates with information stored in a DNS record.
- No Certificate Authority (CA) needed.
- Using DNSSEC for authenticity and integrity.

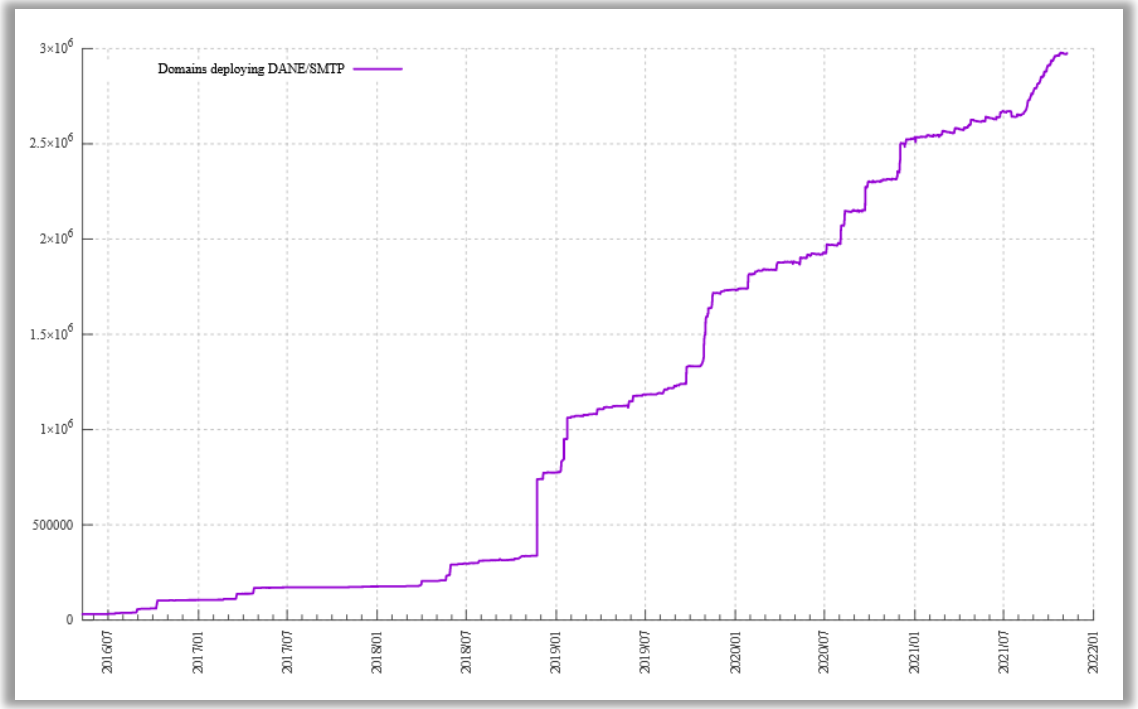
What is DANE?

DANE for web

- Not used due to lack of browser support.

DANE for SMTP

- Usages keeps increasing.
- Oct 2021: almost 3 million domains.

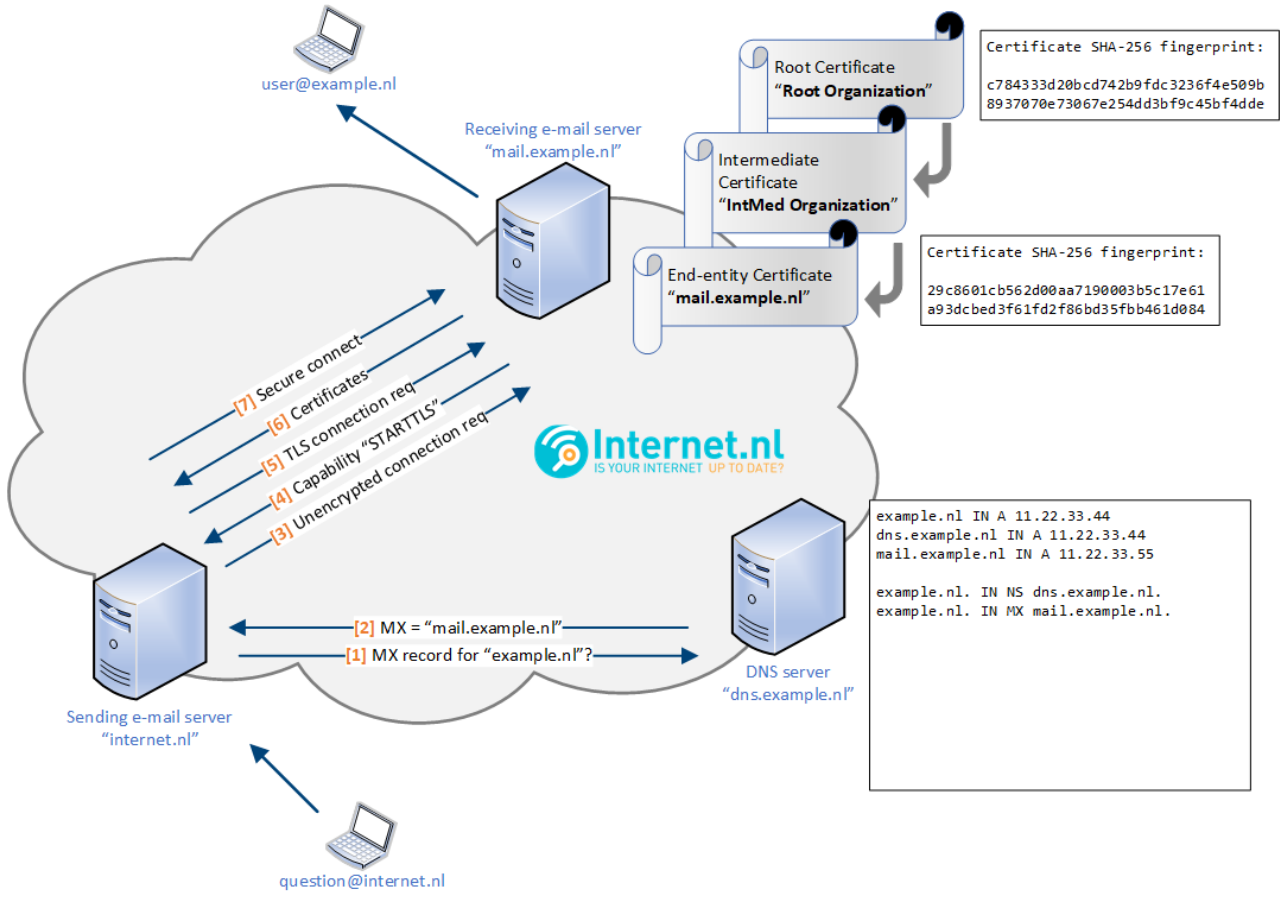


Risks mitigated by DANE

Using an encrypted SMTP connection based on STARTTLS still leaves email transport at risk:

- STARTTLS is **opportunistic**, which means that encryption is only used after being negotiated over an unencrypted connection.

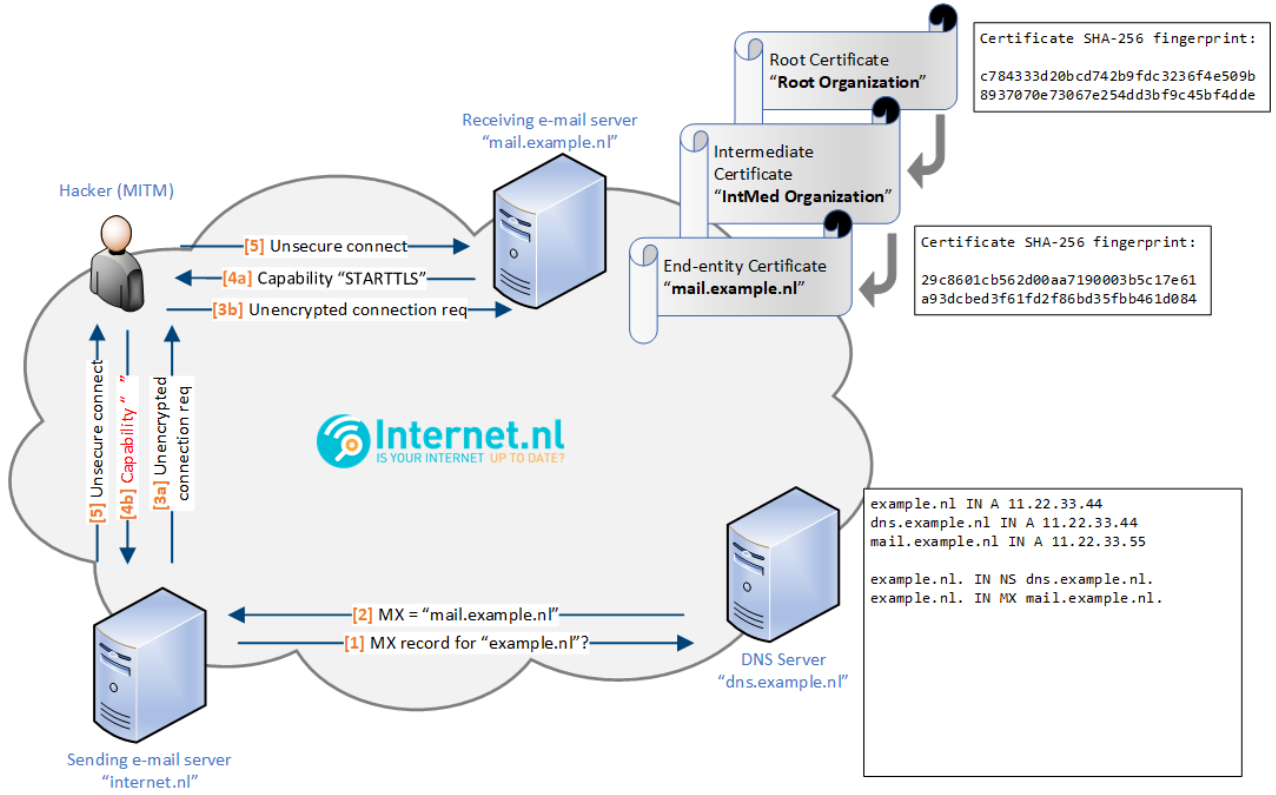
- At the same time SMTP servers, by design, **do not validate the authenticity of another mail server's certificate**; any random certificate is accepted.



Risk 1: STRIPTLS / downgrade attack

The opportunistic character of SMTP makes it relatively easy for cybercriminals to circumvent the usage of encryption and force transfer of emails over an unencrypted connection.

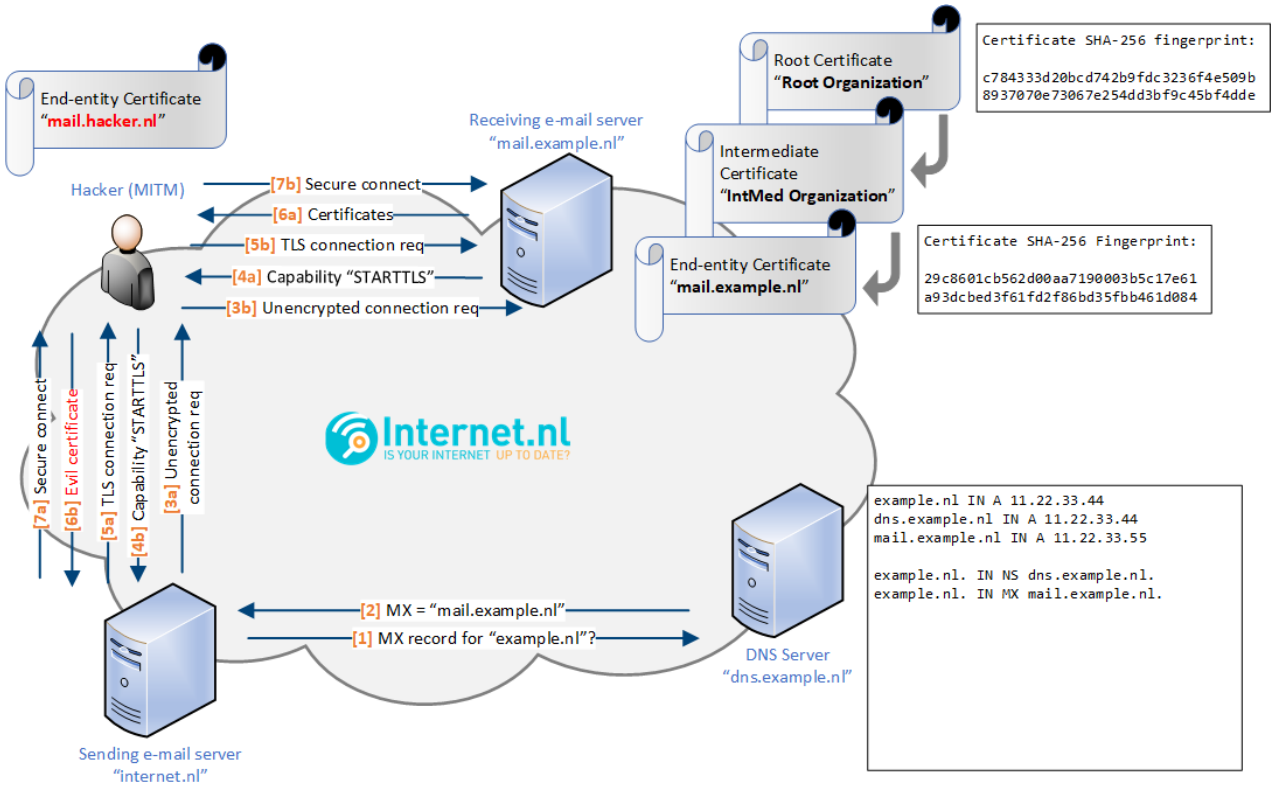
Without the STARTTLS capability, the sending server will proceed to transport the e-mail unencrypted.



Risk 2: Divert mail traffic

Not validating the authenticity of another mail server's certificate allows for any random certificate to be accepted. This again makes it relatively easy for cybercriminals to manipulate email transport.

The certificate of the hacker is used for encryption of mail transport. The hacker can now unencrypt traffic and manipulate email transport.



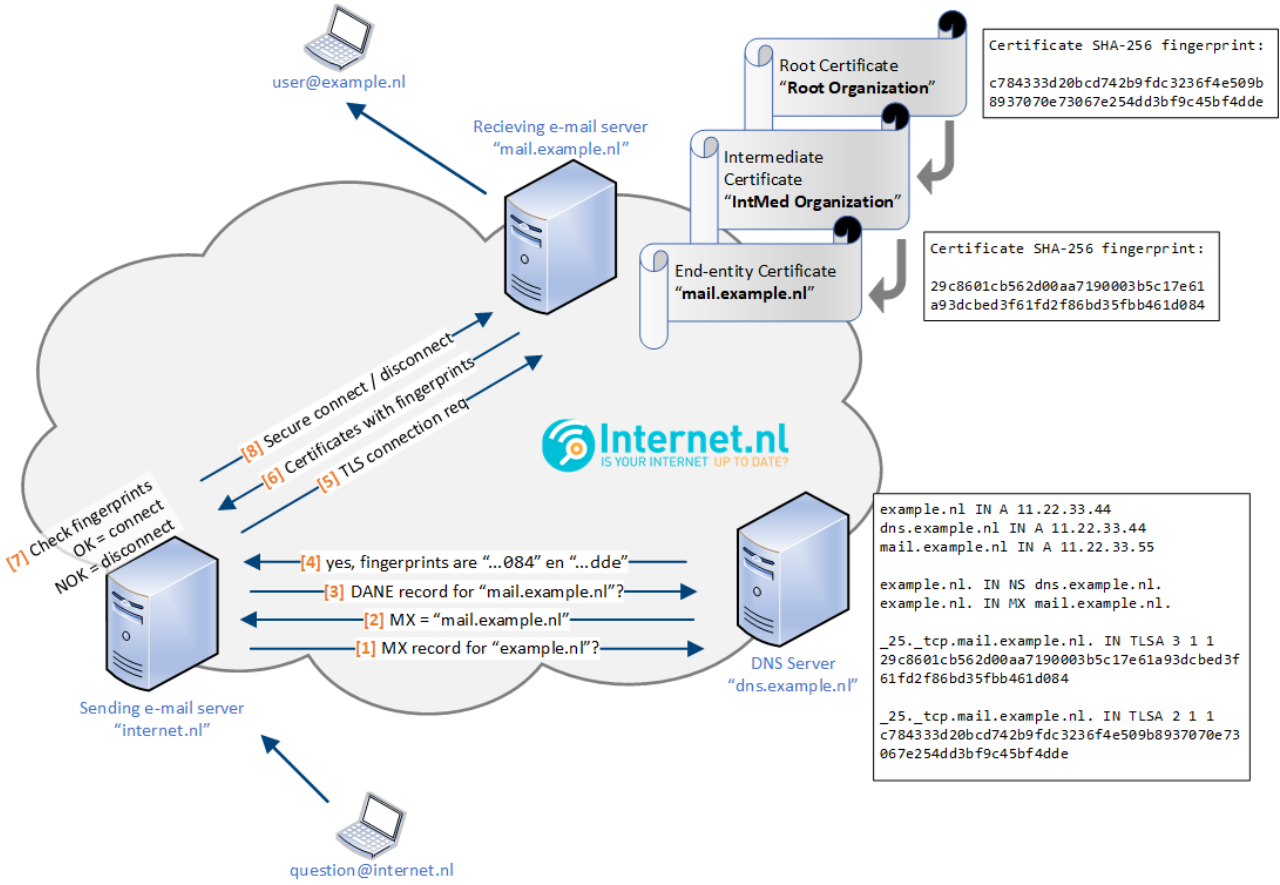
Secure transport with DANE

To ensure reliable TLS connections DNSSEC is used for retrieving information that is published by a domain name's owner or administrator: the TLSA record.

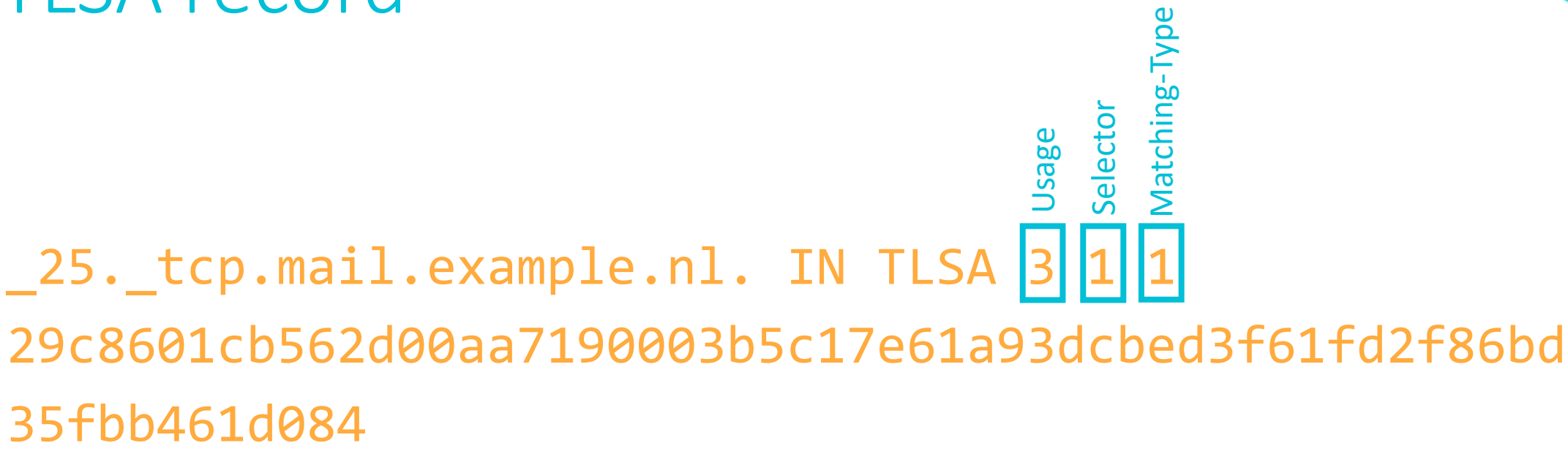
This TLSA record enables SMTP servers to:

- determine up front (before setting up the connection) whether or not another SMTP server supports an encrypted connection.
- validating the authenticity of the other mail server's certificate.

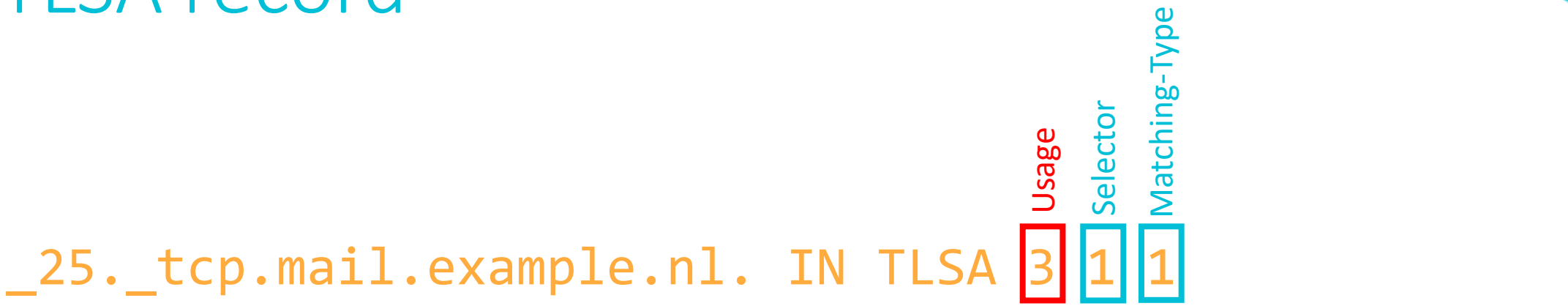
Email transport is now secure.



TLSA record


Usage
Selector
Matching-Type
_25._tcp.mail.example.nl. IN TLSA 3 1 1
29c8601cb562d00aa7190003b5c17e61a93dcbed3f61fd2f86bd
35fbb461d084

TLSA record



Usage: *about the type of certificate that is used for this TLSA record*

~~0~~	~~PKIX-TA~~	(not used / not recommended)
~~1~~	~~PKIX-EE~~	(not used / not recommended)
2	DANA-TA	Specifies an intermediate / root certificate
3	DANE-EE	Specifies an end-entity certificate

TLSA record

Selector: *about the scope of the fingerprint regarding this TLSA record*

0	Full certificate	Fingerprint regarding full certificate
1	Public key	Fingerprint regarding public key

TLSA record

Matching-Type: about the hashing mechanism used for fingerprinting

0	Exact match	
1	SHA-256	Fingerprint is a SHA-256 hash
2	SHA-512	Fingerprint is a SHA-512 hash

Ensuring DANE validation

- When installing a new certificate, it is import to make sure that there is always a valid TLSA record which can be used to verify the certificate offered.
  - Add the TLSA record of the new certificate well before you start using the new certificate.
  - Take into account the time needed for DNS records to spread across the internet (TTL).

This is called a **roll-over scheme**, and it exists in two flavors.

DANE roll-over: current + next

```
[CURRENT] _25._tcp.mail.example.nl. IN TLSA 3 1 1 current-sha256-publickey
[NEXT] _25._tcp.mail.example.nl. IN TLSA 3 1 1 next-sha256-publickey
```

- 2 TLSA records.
  - One for the current EE certificate.
  - One for the next EE certificate.

DANE roll-over: current + issuer

```
[CURRENT] _25._tcp.mail.example.nl. IN TLSA 3 1 1 current-sha256-publickey
[ISSUER] _25._tcp.mail.example.nl. IN TLSA 2 1 1 issuer-sha256-publickey
```

- 2 TLSA records.
  - One for the current EE certificate.
  - One for a TA certificate; an intermediate or root-certificate in the chain-of-trust.

Tips for implementation

A. DANE is meant to be **used for the MX domain**. So if you are using another domain's mail servers, make sure to ask the administrator of that domain (your mail provider) to support DANE by setting up a TLSA record.

B. It is highly recommended to use a certificates **public key for generating a TLSA signature** (selector type "1") instead of the full certificate (selector type "0"), because this enables the (limited) reuse of key materials.

C. Make sure the **TTL (time-to-live) of your TLSA records is not too high**. This makes it possible to apply changes relatively fast in case of problems. A TTL between 30 minutes (1800) and 1 hour (3600) is recommended.

D. In case of roll-over scheme "current + issuer", the use of the **root certificate is preferred** because in some contexts (PKIoverheid) this makes it easier to switch supplier / certificate without impacting DANE.

E. DANE still works when using **self-signed and/or expired certificates**.

F. Certificate name checks are **not performed** for end-enity certificates (usage type 3). However, they **are performed** for intermediate / root certificates (usage type 2).

Tips for implementation

G. Check if DANE TLSA records (_25._tcp.mail.example.nl) are **properly DNSSEC signed**. A regularly occuring mistake is the presence of "proof of non-existence" (NSEC3) for the ancestor domain (_tcp.mail.example.nl). If this happens then resolvers that use Qname minimization (like the resolver used by Internet.nl) think that _25._tcp.mail.example.nl does not exists since _tcp.mail.example.nl does not exists. Therefore the resolver can't get the TLSA record which makes DANE fail.
   - Note that more or less the same principle goes for regular DNS lookups. According to RFC 8020 a NXDOMAIN response means that all the names under it do not exist.

H. If TLSA records are found but are unusable, email transport will **NOT fallback** to opportunistic TLS (STARTTLS) or ultimately plaintext delivery. Email will NOT be delivered.

How about MTA-STS?

- MTA-STS is less secure compared to DANE for SMTP.
  - Only for domains that are not (yet) able to deploy DNSSEC.
  - MTA-STS is relatively complex because it needs an extra HTTPS interface (including certificate validation).
  - Weaknesses are documented in its specification in section 10 of RFC8461: trust on first use + caching.
- MTA-STS and DANE can co-exists next to each other. They intentionally do not interfere.
- Only a small number of large providers implemented MTA-STS: gmail.com, mail.ru, comcast.net.
- DANE is more popular
  - MTA-STS: 3,000 domains that use (enforce) MTA-STS
  - DANE: almost 3,000,000 domains with TLSA records on the mail server domain

More details: https://www.isi.edu/~hardaker/news/2021-09-20-DANE-vs-STS.html

Questions?

Feel free to contact me at question@internet.nl.

Other interesting sources

- https://github.com/baknu/DANE-for-SMTP/wiki
- https://blog.apnic.net/2019/11/20/better-mail-security-with-dane-for-smtp/
- https://www.globalcyberalliance.org/resource/layers-of-defense-dane-and-dmarc/
- https://stats.dnssec-tools.org/ & https://dane.sys4.de