Securing Security - DANE TLSA Certificate Rollover

Patrick Koetter und Carsten Strotmann, sys4 AG

CREATED: 2023-10-21 SAT 09:48

DANE TLSA Certificate Rollover

Certificate Rollover

- The default certificate lifetime is 1 year (as of 2023)
    - Some certification authorities limit the lifetime of certificates to 3 month (90 days)
- An active certificate should be exchanged (***rolled***) before it expires
    - A good rollover time is after 2/3 of the certificates lifetime, or one month before certification expiry

Certificate Rollover and DANE

- With DANE, an active certificate has a dependency on the TLSA record in DNS
- Whenever the certificate on a server will change, the TLSA record set needs to be update first

Steps for a DANE secured TLS record rollover

1. Create or request a new certificate, but do not deploy it on the server
2. Create the TLSA record for the new certificate and publish it in DNS
3. Wait for the TTL of the TLSA record set in the DNS zone (add some buffer time for resolver that do not honor the TTL)
4. Exchange the certificate on the server
5. Test the new DANE setup, e.g. via https://dane.sys4.de or https://www.huque.com/bin/danecheck
6. Remove the old TLSA record from the DNS zone

DANE-TLSA rollover automation

- The python tool ***Let's DNS*** can automate TLSA certification rollover
    - Website: https://letsdns.org

DANE Best Practices

DANE TLSA CA Binding

- The validation of an x509 certificate with DANE is successful as long as **one** TLSA record for the service is valid
- As a *fallback* to prevent outages on TLS certificate rollovers, add a TLSA record for the certificate of the issuing CA (DANE CA pinning / DANE-TA)
    - With such a record, every x509 certificate issued by this CA is seen as valid (even if there is no direct match with an leaf-cert TLSA *3 1 1* record)
    - Recommendation: In addition to an DANE TLSA record that validates the end-certificate on the server (flags 3 1 1 = DANE-EE) publish an additional TLSA record with the flags 2 1 1 (DANE-TA)

Example: create a DANE-TA record for Let's encrypt

- Load the ***Let's Encrypt*** root certificate (see https://letsencrypt.org/certificates/)

```
% curl -O https://letsencrypt.org/certs/isrgrootx1.pem
```

- Create a DANE-TA record based on the Root-Certificate

```
% ldns-dane -n -c isrgrootx1.pem create mail.zXX.dane.onl 25 2 1 1
```

- Publish the new TLSA record in the DNS zone, test the new record (e. g. https://dane.sys4.de)

Automating TLSA record updates via dynamic DNS

- Dynamic DNS (RFC 2136) can ease the automation and the management of TLSA records
- Dynamic DNS updates ...
  - prevent syntax errors in DNS records
  - the SOA serial number is automatically incremented
  - the new TLSA record is published immediately
  - DNSSEC signatures are automatically generated

Benefits of dynamic DNS

- The TLSA record rollover can be scripted
- Certificates can be managed from a central system using the ACME protocol (RFC 8555 - Automatic Certificate Management Environment (ACME)) using the ***DNS-01 challenge***. There is no webserver needed on the target system for which the certificate is being requested (as with the default ACME web-challenge).
    - The DNS-01 challenge requires dynamic access to the DNS zone content, either via dynamic DNS or via API interfaces of a DNS server or a DNS hosting service provider

Common DANE TLSA mistakes

(Source: https://dane.sys4.de/common_mistakes)

Publishing DNSSEC DS records, and DANE TLSA records as a fashion statement

- Keeping these correct requires operational discipline. Administrators who expect "fire and forget" should not publish DNSSEC signed zones or DANE TLSA records. Or they can pay others to host their zones and DANE TLS services.
- Operating poorly maintained DNSSEC zones or TLSA records creates problems not only for the domain in question, but also for all the domains trying to communicate with such a domain. Everyone is better off if DNSSEC and DANE are taken seriously.

Failure to automate DNS zone signing

- Multiple domains have been observed to become non-operational because the RRSIG records have expired. MAKE SURE you've automated zone signing reliably.

Failure to update TLSA records before updating the matching server certificate (1)

Failure to update TLSA records before updating the matching server certificate (2)

- Remember to check that all secondary nameservers are getting timely updates of changes. The TTL clock starts only after the last secondary nameserver is serving the updated TLSA records.
- When your DANE TLSA records are CNAMEs to a location where your organization's issuing authority maintains suitable TLSA records for you, you can deploy new certificates from that authority without updating the server's TLSA records. The burden of key rollover falls on that authority, before they issue any certificates via a new certificate or key.

Failure to update TLSA records before updating the matching server certificate (3)

With DANE-TA(2) certificate usage, failure to include the Issuing CA (trust-anchor) certificate in the server's certificate chain.

- With DANE-TA(2), the issuing CA certificate MUST be configured in the server's certificate chain file. DANE SMTP clients typically do not look for these in any local list of trusted roots, even if they have such a list (no such list is required or expected with DANE for SMTP).

Unsupported certificate usage

- For SMTP the TLSA record certificate usage MUST be either DANE-TA(2) or DANE-EE(3). The usages PKIX-TA(0) and PKIX-EE(1) are NOT supported.

Incorrect TLSA selector

- Some domains publish TLSA records with a selector of SPKI(1), which indicates a digest of a public key, but the digest in the TLSA record is that of the containing certificate. Make sure the digest is computed over the correct object. Use a command-line tool or website that automatically extracts the correct data from the certificate.

Incorrect TLSA digest

- Some domains compute the digest of something other than the binary ASN.1 DER form of the certificate or public key (SPKI format). Make sure you compute the digest of the correct encoding. Use a command-line tool or website that automatically converts the data into the correct form.

Selective availability of STARTTLS

- Some SMTP servers only enable STARTTLS for clients that have a history of sending non-spam email to the receiving domain. If TLSA records are published by a domain with such servers, new senders are in a catch-22. They can't send email with STARTTLS disabled (the domain's DANE TLSA records require use of TLS). And STARTTLS won't be enabled until they've sent email.
- Selective availability of STARTTLS is not compatible with DANE. Make sure that either STARTTLS is always on, or DANE TLSA records are NOT published for your domain.
- Keep in mind that STARTTLS may be disabled by a proxy such as "spamd" or similar, that sits between remote clients and your SMTP server.

Firewalls that filter out TLSA queries

- If your domain is DNSSEC validated, Make sure that your firewalls allow TLSA queries to reach your nameservers, over every address type. (Some firewalls block TLSA lookups only for IPv4).

Broken nameservers

- Some nameservers (various djbdns versions patched for DNSSEC, older PowerDNS versions, ...) don't handle "denial of existence" correctly. The negative reply for "IN TLSA _25._tcp." lookups appear "bogus" to validating resolvers when the records in fact don't exist.
- Make sure to check each of your nameserver's for valid denial of existence responses. This applies to all DNSSEC domains, not just those implementing DANE, and especially if they are NOT publishing TLSA records.

Partial implementation

- DANE only protects your domain if the domain is DNSSEC validated, ALL its MX hosts are also in DNSSEC validated zones (their A/AAAA records are "secure"), and all of the MX hosts have "_25._tcp" TLSA records.
- When only some of the MX hosts are protected, an active attacker can block access to the protected ones, allowing connections to just the unprotected hosts, where SMTP transport is subject to STARTTLS downgrade or other attacks.
- Publishing DANE TLSA records for your SMTP servers only makes sense if you are planning to eventually publish TLSA records for all your MX hosts.

Further Information

Additional information

- DANE How-To https://github.com/internetstandards/toolbox-wiki/blob/master/DANE-for-SMTP-how-to.md
- A sensible "3 1 1" + "2 1 1" key rotation approach: https://mail.sys4.de/pipermail/dane-users/2018-February/000440.html
- Please avoid “3 0 1” and “3 0 2” DANE TLSA records with LE certificates https://community.letsencrypt.org/t/please-avoid-3-0-1-and-3-0-2-dane-tlsa-records-with-le-certificates/7022

Additional information

- DANE for SMTP and TLS certificate agility https://mail.sys4.de/pipermail/dane-users/2017-August/000417.html
- DANE Implementation resources https://github.com/baknu/DANE-for-SMTP/wiki/2.-Implementation-resources

Useful RFC references

- RFC 7671: Key Rollover with Fixed TLSA Parameters
  https://datatracker.ietf.org/doc/html/rfc7671#section-8.1
- RFC 7671: TLSA Publisher Requirements: Summary
  https://datatracker.ietf.org/doc/html/rfc7671#section-8.4

End

Questions? Answers!