



DANE, Explained

Using your own DNS to say exactly which
certificate your mail servers will present

A plain-English guide for people who run businesses, not mail servers.
The fifth in a series, and the companion to the MTA-STS guide. It needs
one thing the others do not: a DNSSEC-signed domain. Section 01
explains how to tell.

01

The one prerequisite

Read this page before any other. It decides whether the rest applies to you.

DANE lets you publish, in your own DNS, a fingerprint of the certificate your mail servers will present. A sending server looks it up and refuses to deliver to anything that does not match. No certificate authorities, no policy files, no web server.

That only works if the sending server can trust what your DNS told it. Otherwise an attacker who can tamper with DNS simply replaces your fingerprint with their own. The thing that makes DNS trustworthy is DNSSEC.

DANE is built on DNSSEC. Not alongside it — on top of it.

Your mail is safe from interception

because the sending server checked the right certificate

The TLSA record says which certificate is yours

a fingerprint published in your DNS

DNSSEC proves that record really came from you

without this, an attacker can simply replace it

If your domain is not signed with DNSSEC, DANE is not available to you. There is no partial version and no way around it. That is the first thing to check, before anything else in this guide.

Figure 1 — Remove the bottom layer and nothing above it stands.

How to tell whether you have it

Most domains do not. Ask your DNS provider whether DNSSEC is enabled for your zone, or check with any DNSSEC validator — ReaperCloud's DNS tool reports it. If the answer is no, DANE is not available to you, and the MTA-STS guide covers what to do instead. That is a legitimate destination, not a consolation prize.

What it protects, and what it does not

The same job as MTA-STS, by a different route.

DANE protects the connection carrying your mail, exactly as MTA-STS does. It has nothing to do with forgery, phishing or spam — those are the business of SPF, DKIM and DMARC, and they should be in place first.

What it adds over MTA-STS is that there is no gap on first contact. MTA-STS asks a sender to fetch a policy over the same internet an attacker may be interfering with. DANE's answer arrives signed, so a tampered answer is detected rather than believed.

The cost of that strength

DANE has no testing mode. The moment you publish a usable record, senders that support DANE are required to check it and to refuse delivery if it does not match. Everything else in this guide follows from that one sentence.

The record you publish

Three small numbers and one long fingerprint.

```
_25._tcp.mx.yourdomain.com.  IN  TLSA
3  1  1  d2abde240d7c...
```

3	Usage	Which certificate this record is about, and how much of the normal certificate system still applies. Only two of the four values may be used for mail.
1	Selector	Whether the fingerprint covers the whole certificate or just its public key. The key alone survives renewals.
1	Matching	Which fingerprinting method. SHA-256 is the one every implementation must support.
d2abde240d7c...	Fingerprint	Generated from your certificate. You do not choose this; a tool produces it.

Published once per mail server, under its own hostname — not under your domain name.

Figure 2 — A TLSA record, pulled apart.

Note where it lives. The record is published under your *mail server's* hostname, not under your domain — so a domain with three mail servers needs three records, one for each. If several servers share a certificate, DNS aliases can point them all at a single record rather than repeating it.

04

The first number, and why two values are wrong

Generators offer four. Mail may use two.

Four possible values. Mail may only use two of them.

0	~~PKIX-TA~~	Not for mail — relies on a list of trusted authorities that mail servers do not keep.
1	~~PKIX-EE~~	Not for mail — same problem.
2	DANE-TA	Your own issuing authority. Useful if one authority signs several servers.
3	DANE-EE	The server certificate itself. The recommended choice for mail.

The first two exist for other services, such as web servers. For mail on port 25 the standard says not to publish them, and sending servers may simply ignore records that use them.

Figure 3 — Two of these must not be used for mail.

The reason the first two are unusable is worth one sentence: they ask the sending server to also check the certificate against a list of trusted certificate authorities, and mail servers do not keep such a list. A record that depends on one is a record many senders cannot act on.

What to actually publish

Two combinations, and a default.

In practice there are two sensible answers.

RECOMMENDED

3 1 1

DANE-EE, public key, SHA-256

The recommended choice. The fingerprint covers your public key alone, so renewing the certificate with the same key needs no DNS change at all.

ALTERNATIVE

2 0 1

DANE-TA, whole certificate, SHA-256

A second choice, for when one internal authority signs several servers. That authority's certificate must be sent during every connection.

If you have no particular reason to prefer the second, publish the first.

Figure 4 — The two sensible parameter sets.

The advantage of the recommended set is worth spelling out. Because the fingerprint covers your public key rather than the whole certificate, renewing that certificate with the same key changes nothing in DNS. Renewals stop being a risk. If your renewal process generates a fresh key each time, that advantage disappears and every renewal becomes a rollover — Section 07.

What a sending server does

And what happens when the answer is no.

What a sending server does before it will talk to you.

1

Looks up your MX records — and checks the DNSSEC signatures

If the answer is not signed, DANE stops here.

2

Looks up the TLSA record for that mail server

Again, only a DNSSEC-signed answer counts.

3

Connects, and compares the certificate to the fingerprint

STARTTLS is now required, not optional.

If the certificate does not match, the message is not delivered. There is no testing mode, no percentage, and no way to ask senders to be lenient. A wrong record stops mail for everyone whose server supports DANE.

Figure 5 — Every step depends on a signed answer.

The important detail is at the end. With MTA-STS a sender that cannot make sense of your policy carries on and delivers. With DANE, a published record that does not match is treated as an attack in progress, which is precisely what you asked for.

Changing a certificate without stopping mail

The section this guide exists for.

Your fingerprint describes a certificate. Replace that certificate and the fingerprint is wrong — and a wrong fingerprint is not a warning, it is a refusal. The fix is entirely a matter of order.

Publish the new fingerprint first. Change the certificate second. Never the other way round.

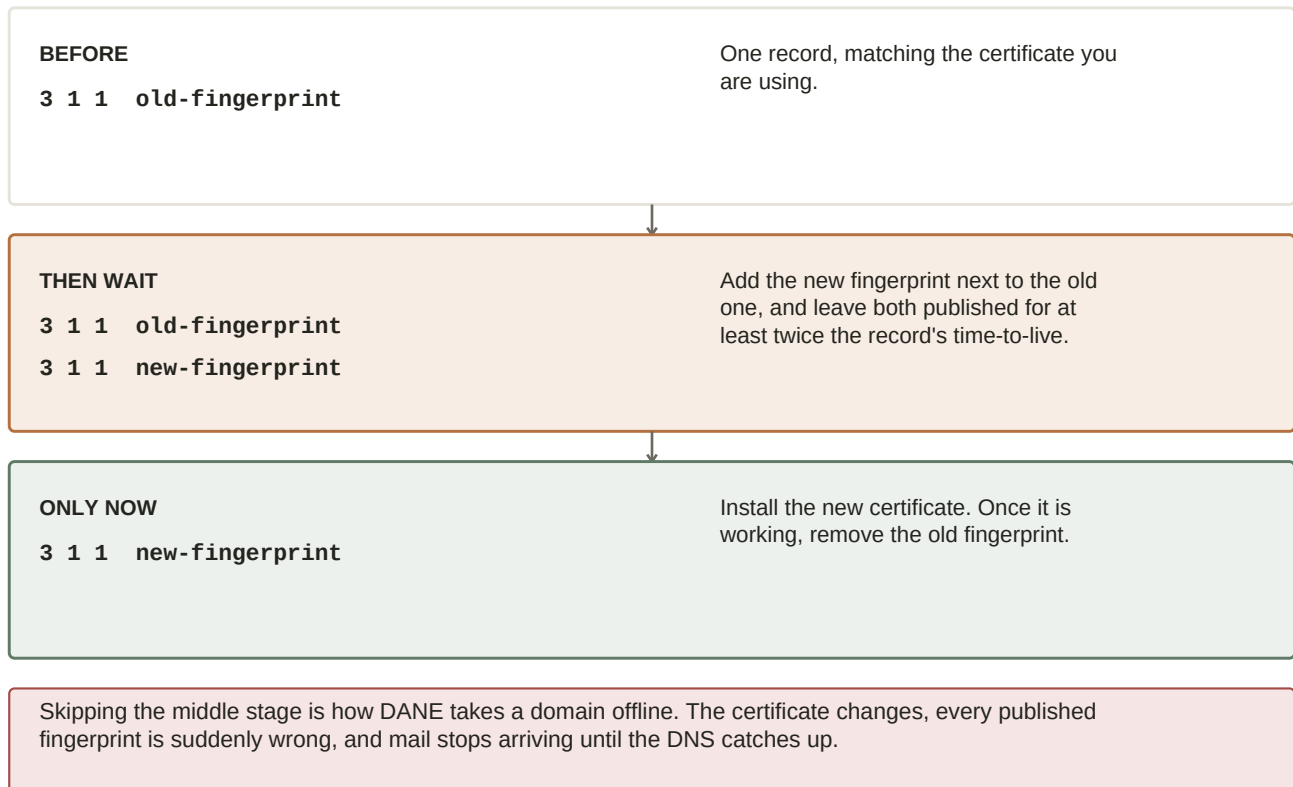


Figure 6 — New fingerprint first, new certificate second.

The waiting stage exists because sending servers remember DNS answers for a while. Publishing both fingerprints and waiting lets every cached copy of the old answer expire before the certificate changes, so no sender is ever holding a fingerprint that does not match what your server presents.

Automate this, or write it down

The danger is not the first setup, which gets attention. It is the renewal eighteen months later, done by someone who was not there, using a tool that knows nothing about DNS. Whoever can replace a certificate must know that a TLSA record exists.

How it fails, compared with MTA-STS

The habits from the previous guide do not all carry over.

	MTA-STS	DANE
Rehearsal available	Yes — testing mode	No
If you publish it wrong	Reports, mail still flows	Mail stops
If discovery is blocked	Falls back to no protection	Refuses to deliver

Neither column is better. DANE is stronger precisely because it refuses to fall back, and that same strictness is what makes a mistake expensive.

Figure 7 — Two protocols, two failure behaviours.

If you have just come from the MTA-STS guide, the instinct to publish something harmless first and watch what happens is a good one, and it is not available here. What you can do instead is check the record against your live server before you publish it, and keep the reporting record from the MTA-STS guide in place — TLS reports cover DANE failures too.

Running DANE and MTA-STS together

Not a choice between them.

The two reach different senders, so publishing both covers more of your mail.

Senders that check DANE

Reached by your TLSA records. A minority of servers, but a growing one, and it includes some large providers.

Senders that check MTA-STS

Reached by your policy file. A different, larger set.

Senders that check neither

Reached by nothing here. They will still use encryption if it is offered — just without any guarantee.

They do not conflict. A sender that supports both will use DANE and ignore the policy file, which is the stronger of the two, so publishing both costs nothing but the setup.

Figure 8 — Different senders, different mechanisms.

A sender that supports both will use DANE and ignore your policy file. A sender that supports only MTA-STS uses the policy file. Nothing conflicts, and the two together cover more of your incoming mail than either alone.

A short checklist

Before, during, and long after.

Before you publish anything

- **Confirm DNSSEC is enabled and validating.** Without it nothing here works.
- **Do SPF, DKIM and DMARC first.** Impersonation is the more commonly exploited problem.
- **Publish the TLS reporting record** from the MTA-STS guide. It reports DANE failures too, and it cannot break anything.
- **Generate the fingerprint from the certificate your server is actually presenting**, not from the file you think it is using.

Getting it right

- **Use 3 1 1** unless you have a specific reason not to.
- **One record per mail server**, under the server's own hostname, including backups.
- **Never change a certificate before republishing.** New fingerprint, wait, then the certificate.
- **Make sure whoever renews certificates knows.** This is an operational dependency, not a one-off setting.

Signs something is wrong

- A TLSA record exists but the zone is not signed. Senders will ignore it entirely — you have the maintenance burden and none of the protection.
- The record uses usage 0 or 1. Not valid for mail; see Section 04.
- You have a backup mail server with no record of its own. It is unprotected, and if it also has a stale record it is unreachable.
- TLS reports show authentication failures right after a certificate renewal. That is the rollover order going wrong.

Check your own domain

ReaperCloud's DANE tool checks whether your zone is signed, finds the TLSA records for every one of your mail servers, and compares each against the certificate that server actually presents — which is the check that catches a bad rollover. reapercloud.com/dane.php, or reapercloud.com/report.php for everything at once.

This document describes DANE as specified in RFC 6698, with the SMTP-specific rules of RFC 7672 and the operational guidance of RFC 7671. It is an explanation, not a specification: where they differ, the RFCs are correct and this is simplified.