30. DFN Konferenz (2023):

Einsatz und Nutzung von DNS TLSA-Records für Mailserver (MTA)

Eine Internet-Studie

Erwin Hoffmann

[ehoffmann@fb2.fra-uas.de]

Frankfurt University of Applied Sciences

9. Februar 2023

Fragestellungen und Inhalt

E-Mail wird häufig als zentrale Bedrohung (*Threat*) im Internet angesehen, ist es doch häufig erfolgreicher Auslöser von Ransom-Angriffen. Rund um den Versand von E-Mails wurden Technologien entwickelt, die versuchen die Risiken zu vermindern.

Neben den *Domainkey Identified Mails* (DKIM), gehört auch die *Transport Layer Security Authentication* (TLSA) bw. *Domain Name Authenticated Name Entities* (DANE) zum Sammelsurium der Gegenmassnahmen.

Hierzu wird ein Blick auf die Arbeitsweise von TLSA/DANE benötigt und dessen Nutzung im Internet untersucht:

1. Wozu dient TLSA?
2. TLSA und DANE: Arbeitsweise und Paradigmen.
3. Internet-Survey: Methodik und Vorgehensweise.
4. TLSA-Survey: Resultate und Überraschungen.
5. Zusammenfassung: TLSA, DNSSec und DANE.
6. Email, DNS und die Rolle von *DNS4EU*.

Autoren

- Die vorliegende Untersuchung fusst auf einer Bachelor-Arbeit von B.Sc. Inf. *Jihad El Hayek* am FB2 der *Frankfurt University of Applied Sciences*, der insbesondere die Domain-Daten herbeigeschafft eine initiale Datennahme und Auswertung (auf einem *RasPI 4*) 2021 durchgeführt hat.
- Die Graphiken für die statistischen Auswertung wurden von *Prof. Egbert Falkenberg* mittels des Programmpaketes 'R' erstellt.
- Ausgangspunkt für die Untersuchung war die Entwicklung eines TLSA-DNS-Lookups im Rahmen der Weiterentwicklung des MTAs **Qmail**, der von *Daniel Bernstein* aus dem Jahre 1998 stammt und nun von mir als Fork **s/qmail** weitergeführt wird.
  Ergänzend hierzu habe ich den (verschlüsselnden) DNS-Nameserver **tinydns** für die einfache Unterstützung von TLSA-Records erweitert (**djbdnscurve6**).
  Die gewonnenen Erkenntnisse nutze ich in meinem Lehrveranstaltungen an der FRA-UAS, speziell *IT-Security*, *C-Programmierung*, *Computer-Netzwerke* und *Software-Engineering*.

Autoren

- Die vorliegende Untersuchung fusst auf einer Bachelor-Arbeit von B.Sc. Inf. *Jihad El Hayek* am FB2 der *Frankfurt University of Applied Sciences*, der insbesondere die Domain-Daten herbeigeschafft eine initiale Datennahme und Auswertung (auf einem *RasPI 4*) 2021 durchgeführt hat.
- Die Graphiken für die statistischen Auswertung wurden von *Prof. Egbert Falkenberg* mittels des Programmpaketes 'R' erstellt.
- Ausgangspunkt für die Untersuchung war die Entwicklung eines TLSA-DNS-Lookups im Rahmen der Weiterentwicklung des MTAs **Qmail**, der von *Daniel Bernstein* aus dem Jahre 1998 stammt und nun von mir als Fork **s/qmail** weitergeführt wird.
  Ergänzend hierzu habe ich den (verschlüsselnden) DNS-Nameserver **tinydns** für die einfache Unterstützung von TLSA-Records erweitert (**djbdnscurve6**).
  Die gewonnenen Erkenntnisse nutze ich in meinem Lehrveranstaltungen an der FRA-UAS, speziell *IT-Security*, *C-Programmierung*, *Computer-Netzwerke* und *Software-Engineering*.

Autoren

- Die vorliegende Untersuchung fusst auf einer Bachelor-Arbeit von B.Sc. Inf. *Jihad El Hayek* am FB2 der *Frankfurt University of Applied Sciences*, der insbesondere die Domain-Daten herbeigeschafft eine initiale Datennahme und Auswertung (auf einem *RasPI 4*) 2021 durchgeführt hat.
- Die Graphiken für die statistischen Auswertung wurden von *Prof. Egbert Falkenberg* mittels des Programmpaketes 'R' erstellt.
- Ausgangspunkt für die Untersuchung war die Entwicklung eines TLSA-DNS-Lookups im Rahmen der Weiterentwicklung des MTAs **Qmail**, der von *Daniel Bernstein* aus dem Jahre 1998 stammt und nun von mir als Fork **s/qmail** weitergeführt wird.
  Ergänzend hierzu habe ich den (verschlüsselnden) DNS-Nameserver **tinydns** für die einfache Unterstützung von TLSA-Records erweitert (**djbdnscurve6**).
  Die gewonnenen Erkenntnisse nutze ich in meinem Lehrveranstaltungen an der FRA-UAS, speziell *IT-Security*, *C-Programmierung*, *Computer-Netzwerke* und *Software-Engineering*.

Spoiler Ende.

SMTP Email-Transport mittels TLS

Das SMTP Email-Protokoll bietet folgende Übertragungsmodi:

- (E)SMTP mit opportunistischer StartTLS-Unterstützung über Port 25; das Workhorse.
- (E)SMTPS mit mandatorischer TLS-Übertragung über Port 465 → keine MX-Records.
- Email-Submission per (E)SMTP mit mandatorischer TLS-Übertragung und Benutzer-Authentisierung unter Nutzung von Port 587.

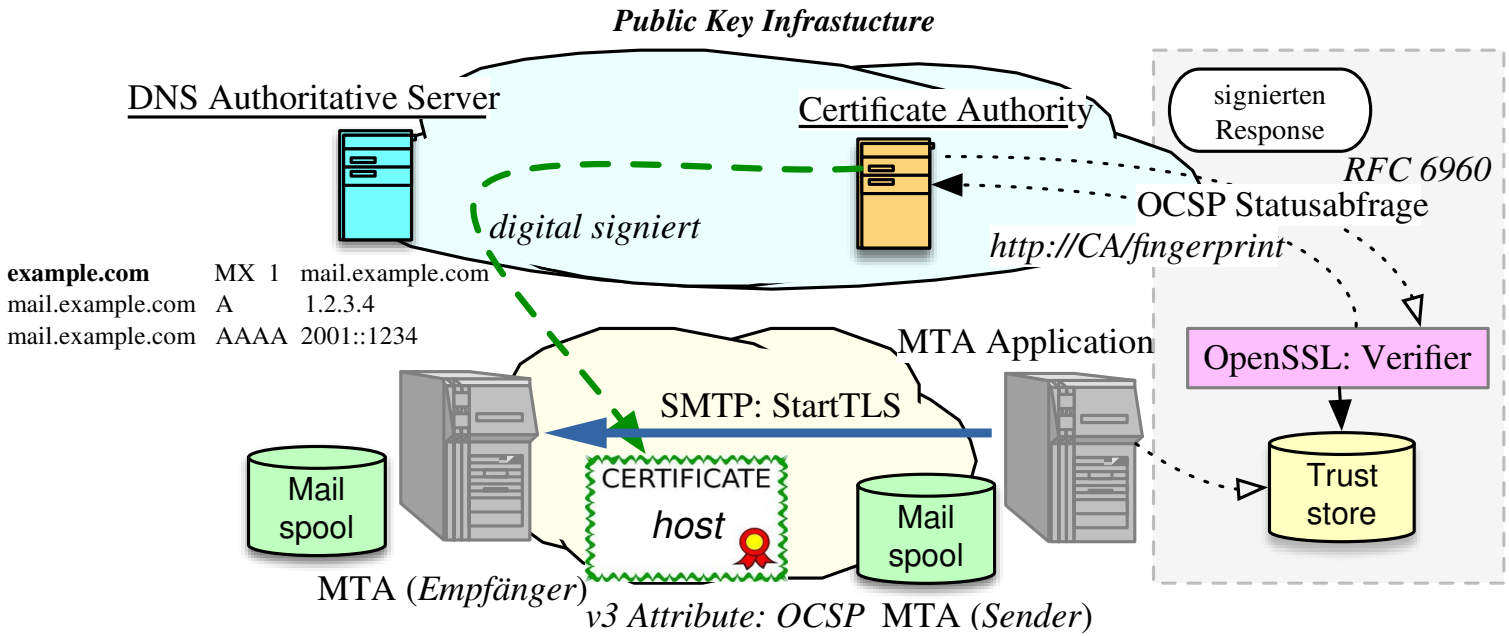

Abbildung: Typischer Email-Kommunikationsablauf mittels StartTLS und X.509 Zertifikat

- Der Empfänger-MTA wurde per TLS-Handshake *authentisiert* → private key. √

SMTP Email-Transport mittels TLS

Das SMTP Email-Protokoll bietet folgende Übertragungsmodi:

- (E)SMTP mit opportunistischer StartTLS-Unterstützung über Port 25; das Workhorse.
- (E)SMTPS mit mandatorischer TLS-Übertragung über Port 465 → keine MX-Records.
- Email-Submission per (E)SMTP mit mandatorischer TLS-Übertragung und Benutzer-Authentisierung unter Nutzung von Port 587.

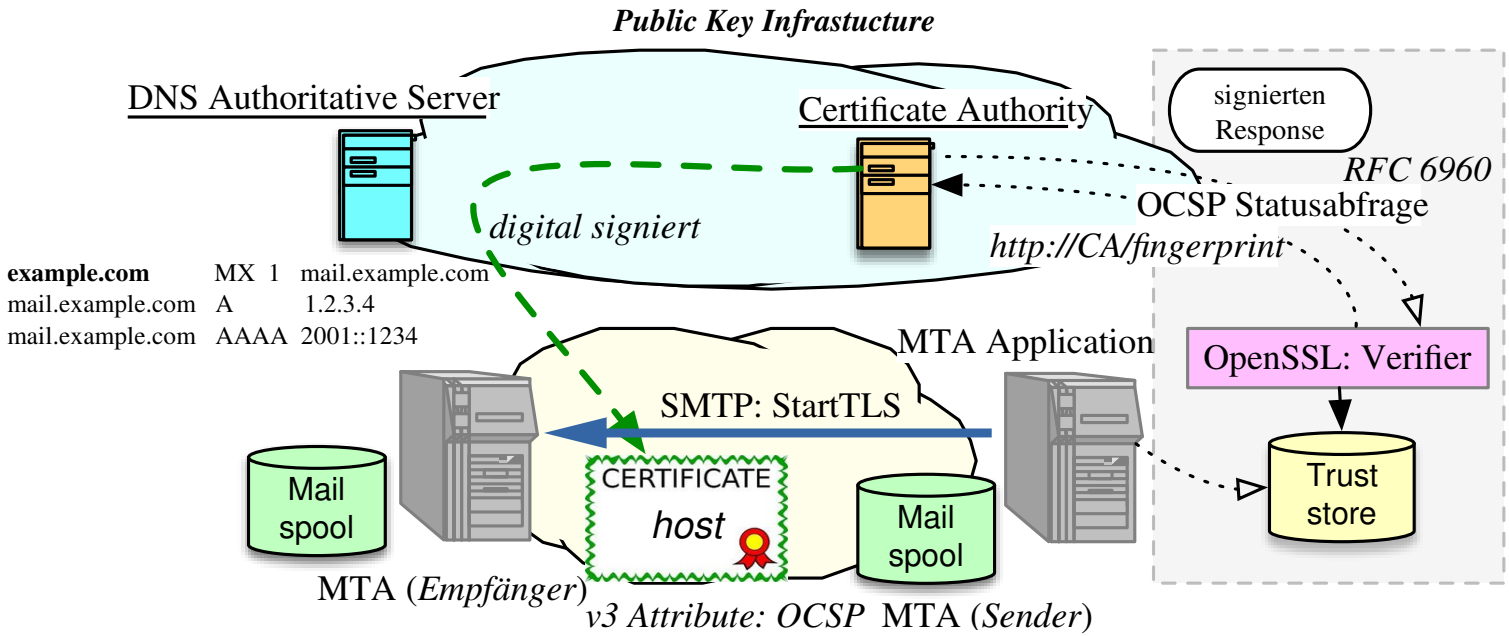

Abbildung: Typischer Email-Kommunikationsablauf mittels StartTLS und X.509 Zertifikat

- Der Empfänger-MTA wurde per TLS-Handshake *authentisiert* → private key. √
- Der Empfänger-MTA wird *legitimiert* → gültige Signatur (CA) → *Trust Store* (unüblich).

Gebrochene Public Key Infrastructure und die Alternativen

Vor 11 Jahren hiess es hier an dieser Stelle: '*PKI ist Tod!*'

- Hintergrund war der Skandal um die niederländischen CA '*DigiNotar*', die über ihren *Exchange* Server gehackt wurden und hierdurch insbesondere (falsche) X.509 Zertifikate für Google ausgestellt wurden.
- Daher führte Google in seinem Webbrowser Chrome X.509 *cert pinning* ein (*HTTP-Based Public Key Pinning*) ein, was aber mittlerweile (seit Chrome 68) durch das Verfahren *Certificate Transparency* (CT) ersetzt ist.
- *Transport Layer Security Authentication* (TLSA) ist ein alternativer Versuch, die X.509 Zertifikatsprüfung ausserhalb der PKI zu unterstützen; benötigt aber ('*RFC 6698 specifies that TLSA RRs are only valid in "secure" zones.*' [1]) eine DNSSec Infrastruktur, was dann als *Domain Name Authenticated Name Entities* (DANE) bezeichnet wird.

↪ TLSA/DANE wird massgeblich vorangetrieben durch *Viktor Dukhovni* – Autor diverser DANE RFCs – der bei *OpenSSL* mitarbeitet und die Implementierung für **Postfix** geschrieben hat. Aus Deutschland hat *Patrick Ben Koetter* an RFC 7672 mitgewirkt. Dieser RFC beschreibt zusammen mit RFC 6698 auch die hier zugrundeliegenden Verfahren.

Email Made in Germany! (?)

[1] Quelle: `https://www.mail-archive.com/dane@ietf.org/msg00802.html`

SMTP Email-Transport und Unterstützung durch TLSA

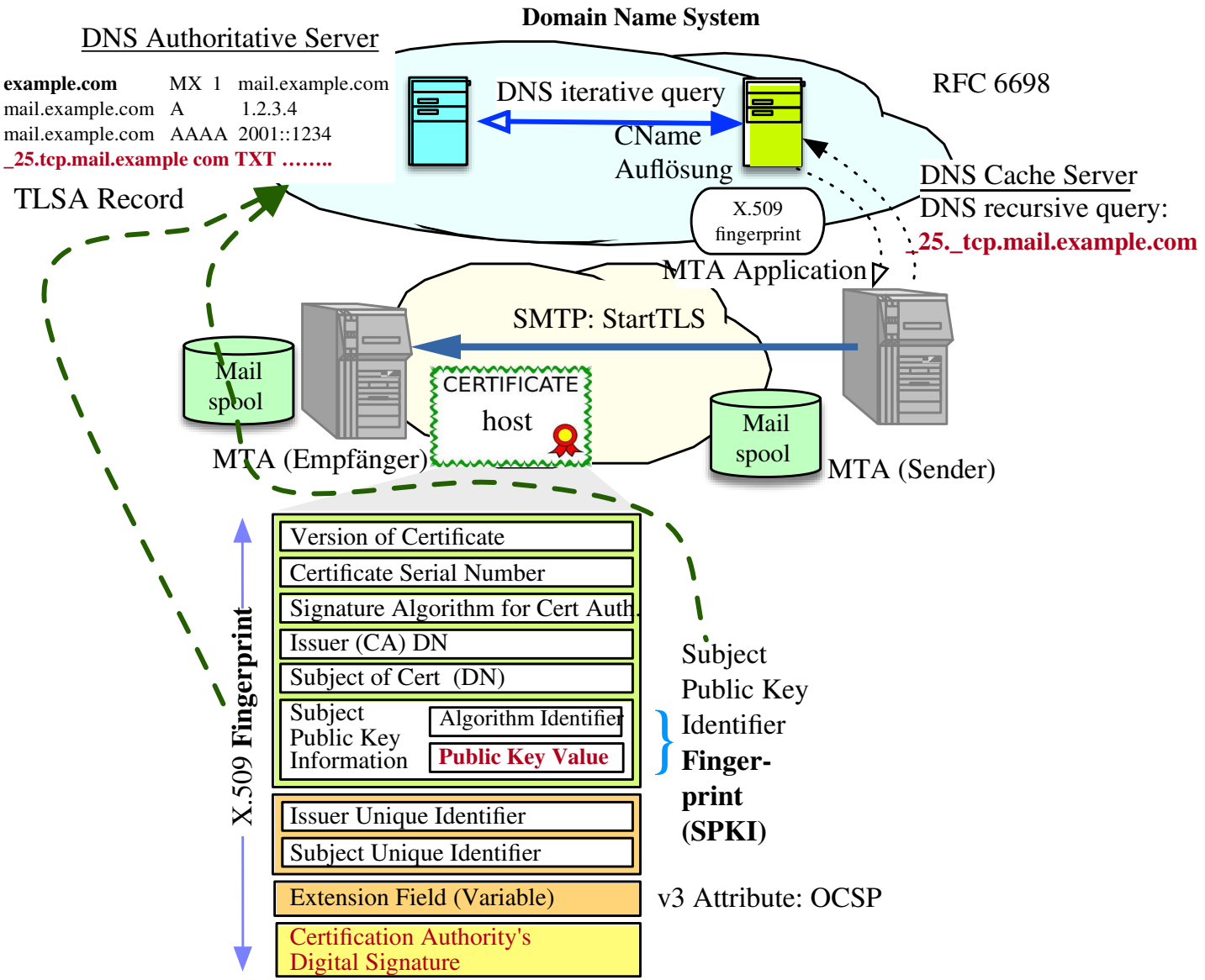

Abbildung: Typischer Ablauf bei der DNS Query/Response für einen TLSA-Record

SMTP Email-Transport und Unterstützung durch DANE

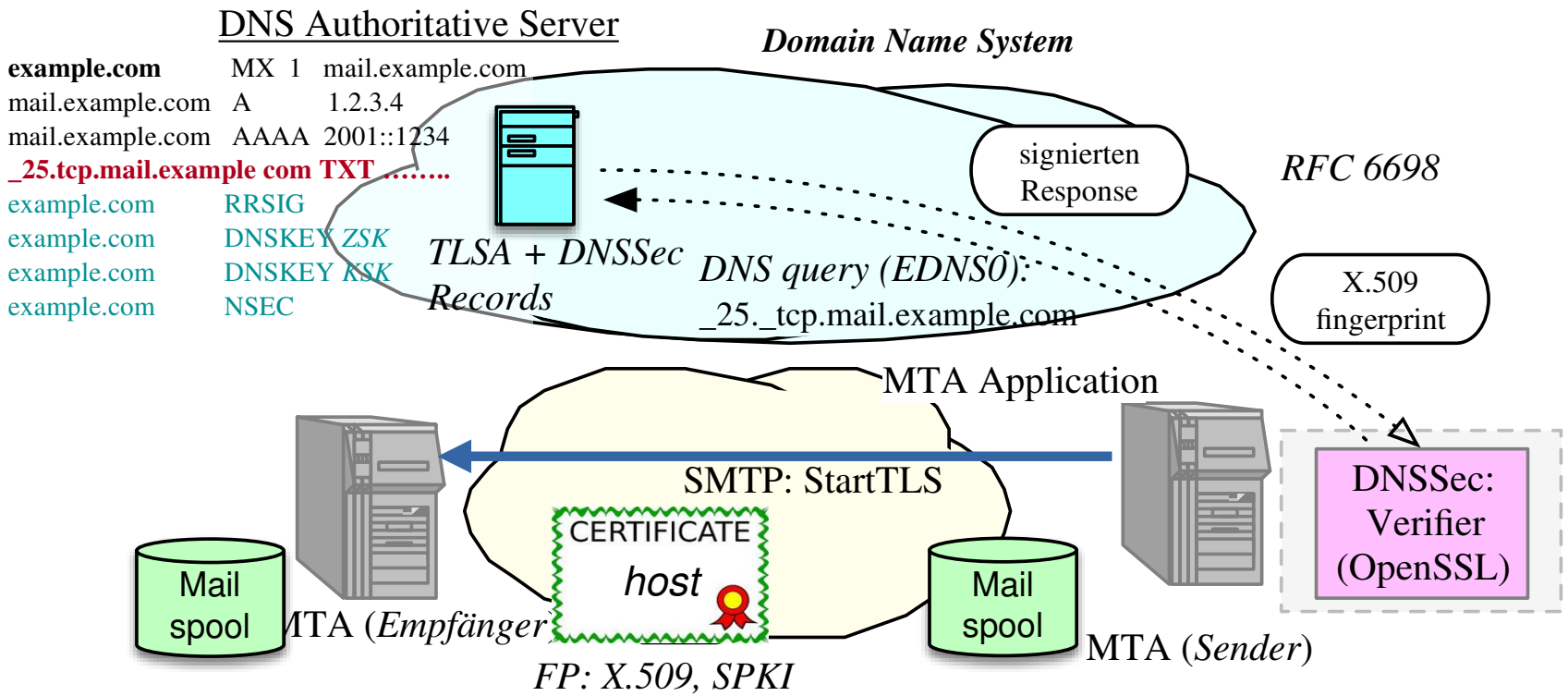

Abbildung: Typischer Ablauf bei der DNS Query/Response für einen TLSA-Record mittels DANE

Der Ablauf ist im Prinzip identisch zur reinen 'TLSA'-Abfrage:

- Der MTA muss aber zusätzlich die DNSSec-Information auswerten → *indicating Resolver* (beim MTA), oder über einen *validating DNS Cache-Server* beziehen; was aber laut RFC 6698 (Abschnitt 8.2/8.3) nicht erwünscht ist.
- Erst dann kann der MTA sicher sein, dass die TLSA-Response 'gültig' ist.
- Wird kein DNSSec angeboten, muss die DNS-Response verworfen werden (RFC 6698 Abschnitt 4.1).

Aufbau des TLSA-Records im Detail

Der DNS TLSA-Record beinhaltet Spezifikationen, wie mit der eingegebenen Information (im DNS) umzugehen ist → es wird eine *Policy* definiert.

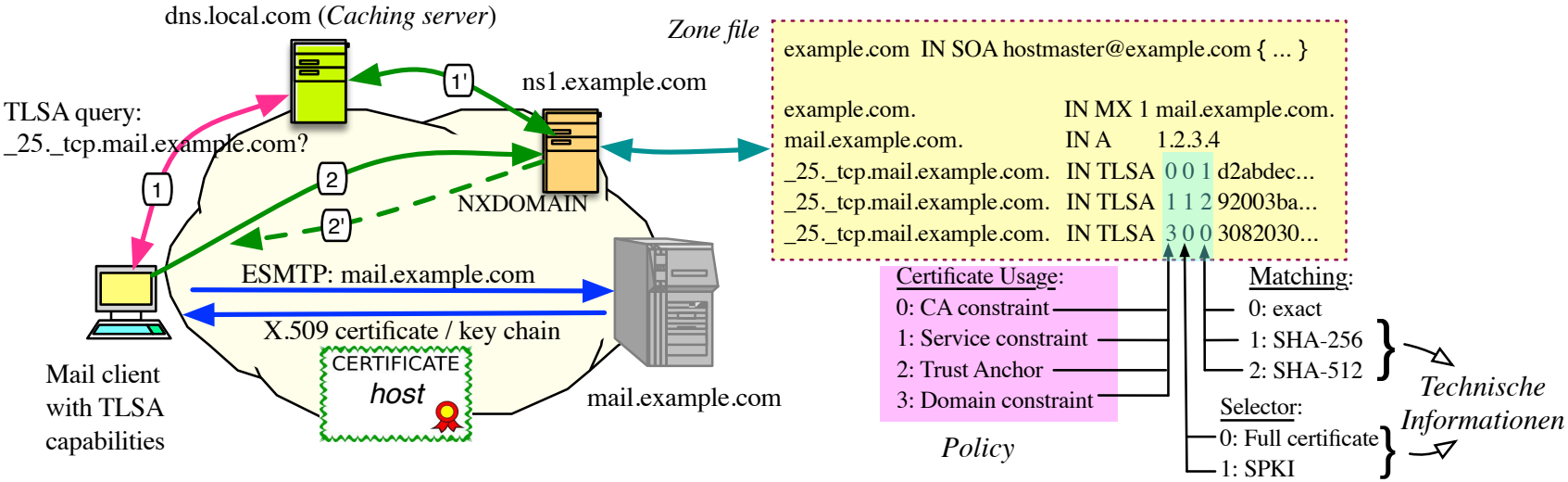

Abbildung: Typischer TLSA-Abfrage eines MTA-Clients mit den möglichen TLSA-Responses; TA: Trust Anchor, EE: Endpoint Entity

- Im TLSA-Record werden technischen Informationen übermittelt:
  - Welche Information in der RDATA-Sektion angeboten wird.
  - Welche Hash-Funktion eingesetzt wurde (überflüssig).
- Welche Verifikation beim Erhalt des X.509-Zertifikats vorzunehmen ist (*Policy*):
  - PKI-Überprüfung (Usage 0 und 1).
  - DNS-Überprüfung (Usage 2 und 3) → verbindlich für SMTP-Mail (RFC 7672 Abschnitt 3.1).

Fragestellung und Werkzeuge

- Vor etwa zwei Jahren war für das Release meiner Software **s/qmail** in der Version 4.1 eine mandatorische TLSA-Überprüfung in der Planung.
- Die DNS-Library wurde um eine generische TLSA-Abfrage ergänzt und das Kommandozeilen-Tool **dnstlsa** hinzugefügt.
- Ergänzend wurde der DNS-Server **tinydns** um die einfache Einbettung von TLSA-Records ergänzt.

Jetzt stellte sich die Frage:

Wie sieht die Verbreitung von TLSA-Records draussen im Internet aus?

Das sollte Gegenstand einer Bachelorarbeit werden und das Vorgehen wurde so gewählt:

1. Ausgehend von den Daten einiger Domains (z.B. .net), sollte die Verbreitung von TLSA-Records überprüft werden.
2. Eine ergänzende DNSSEC-Nutzung wird nicht verlangt.
3. Es soll eine übliche DNS-Abfrage vorgenommen werden, wie sie ein typischer MTA-Client für SMTP-Mail auf Port 25 vornimmt.
4. Es wird aber keine (Start)TLS Verbindung aufgebaut und somit wird auch das X.509 Zertifikat weder angefordert noch gegen die TLSA- Information abgeglichen.
5. Experimentelle TLSA-Records finden keine Berücksichtigung.

Detaillierte Fragestellung und Domain-Material

- Wie weit sind TLSA-Records im Internet im Hinblick auf die Nutzung von SMTP-Mailservices (auf TCP-Port 25) verbreitet?
- Da TLSA/DANE vier *Usage* Möglichkeiten bietet, stellt sich die Frage: Wie ist die konkrete Situation?
- Welche anderen Randbedingungen wie *Selector* und *Matching Types* werden typischer in TLSA-Records genutzt?
- Wir gehen die DNS-Administratoren mit dem *key rollover* bei X.509 Zertifikaten um?
- Wie sieht somit die übliche operative Nutzung von TLSA/DANE-Records aus, die TLS-Services für Mailserver zu unterstützen?

↪ Im Rahmen der Untersuchung wurde ein beachtlicher Teil der Domains im Internet abgefragt.

Die Bachelor-Arbeit nutzte die Zonendaten aus [`https://zonefiles.io/`]. In Q4/2021 wurde hierüber die folgenden gTLDs und ccTLDs einem MX- und TLSA-Lookup unterzogen:

- gTLD: `INFO`, `ORG`, `NET`
- Europe: `AT`, `BE`, `CH`, `CZ`, `DE`, `ES`, `EU`, `FR`, `IT`, `PL`, `RU`, `SE`, `UK`
- America, Asia & Pacific: `AU`, `BR`, `CA`, `CN`, `JP`, `NZ`

↪ Es bestand für diese Domains auch zu diesem Zeitpunkt kein Anspruch auf Vollständigkeit!

Werkzeugkasten und Vorgehen

Die DNS-Abfrage wurde wie folgt vorgenommen:

- Die Query wurde über die Kommandozeilen-Werkzeuge **dnsmxip** und **dnstla** als rekursive Query gestartet.
- Der DNS-Cacheserver **dnscache** nahm eine iterative Query über die autoritativen Nameserver für die entsprechende DNS-Zone vor.
- Die autoritativen Nameserver wurden über IPv4 und IPv6 angefragt.
- Alle Antworten über UDP/UDP+EDNS(0) sowie TCP wurden berücksichtigt.
- Der verfügbare UDP-Puffer für DNS-Nachrichten war `MTUSIZE-52`, also 1228 Bytes.
- Für jede Domain wurde ein üblicher, 'quadratischer' DNS-Lookup vorgenommen, der das Verfahren eines typischen (E)SMTP-Client nachvollzieht.
- DNSSec wurde nicht verlangt.
- Domains wie *BuddyNS*, die per *CurveDNS* verschlüsselte Queries/Responses ermöglichen, wurden per CurveDNS abgefragt.

↪ Bei bei Verarbeitung der TCP-Responses gab es zwei Pannen, weil der verfügbare Speicher für **dnscache** zu klein gewählt war, der Prozess dadurch abbrach und die Auswertung wiederholt werden musste.

Zweistufige DNS-Query: MX → TLSA

Die Auswertung fand zweistufig statt:

A) MX-Qualifizierung: In jeder TLD wurden alle(!) Domains mittels **dnsmxip** dahingehend überprüft, ob sie einen MX-Records besitzen:
   - ▷ Domains ohne MX oder mit Einträgen wie '`localhost`' wurden nicht betrachtet.
   - ▷ Domains, die auf MX-Services wie '`google.com`', '`hotmail.com`' oder auch '`outlook.com`' verweisen, wurden ebenfalls ausgeschlossen → diese besitzen keinen TLSA-Record.

   ↪ Im Ganzen wurden 66 Mio. FQDNs in diesem Schritt identifiziert und der weiteren Analyse zugeleitet.

B) TLSA-Qualifizierung: Für alle akzeptierten MX-Responses wurde eine nachfolgende TLSA-Query per **dnstlsa** durchgeführt und die Ergebnisse protokolliert.

↪ Insgesamt wurden > 300 mio. DNS-Queries erzeugt. Die erhaltenen DNS-Responses (etwa ein GByte) stehen für weitere Untersuchungen per Domain auf GitHub bereit.

Im folgenden werden die MX-Records, die auf `google.*`, `gmail.*`, `hotmail.com` und `outlook.com` zeigen, ausgesteuert. Alle anderen werden als 'unique MX' betrachtet, für die also ein dedizierter MTA für eine Domäne aufgesetzt wurde (in erster Näherung).

Ergebnisse des ersten Schritts: MX-Daten

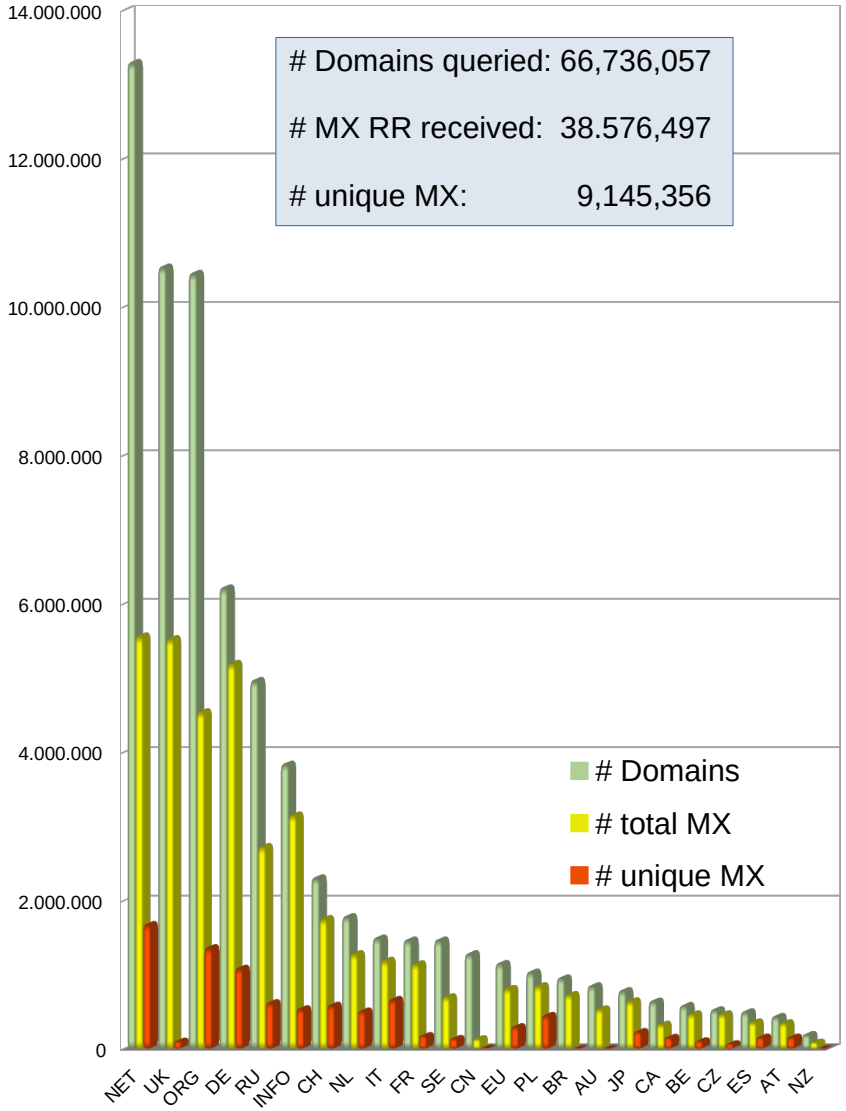

Abbildung: Anzahl der abgefragten Domains per TLD, Anzahl der empfangenen MX-Records und solche, die nicht auf 'Google&Co' verweisen ('unique MX').

Vollständigkeit der Abfrage:

- Gesamt: 66.7 von 363 mio. (18.5%)
- DE Zone: 6 von 12 mio.
- EU Zone: 1.1 von 3.6 mio.

Beispiel von MX-Antworten:

```
2021-12-15 21:13:01 rr 55e9a044
86400 mx speedspharmacy.co.uk. 1
speedspharmacy-co-uk.mail.protection.outlook.com.

2021-12-15 21:14:55 rr 42608ea2 3600 mx
publimetrics.net. 30 mx.publimetrics.net.
2021-12-15 21:14:55 rr 42608ea2 3600 mx
publimetrics.net. 1 aspmx.l.google.com.
2021-12-15 21:14:55 rr 42608ea2 3600 mx
publimetrics.net. 10 alt4.aspmx.l.google.com.

2021-12-15 21:25:33 rr 5bc3f008 3600 mx
publisher.net. 0 localhost.

2021-12-15 21:26:03 rr c0ae440a 300 mx
stayzone.org. 0 .
```

↪ Erkenntnis:
Ein erheblicher Teil der Domänen im Internet verwenden **MX-Delegation**!
→ MXaaS

Ergebnisse des zweiten Schritts: TLSA-Antworten

Frage: Wie gross ist der Anteil der MTAs (MX), für die TLSA-Records ausgerollt sind?

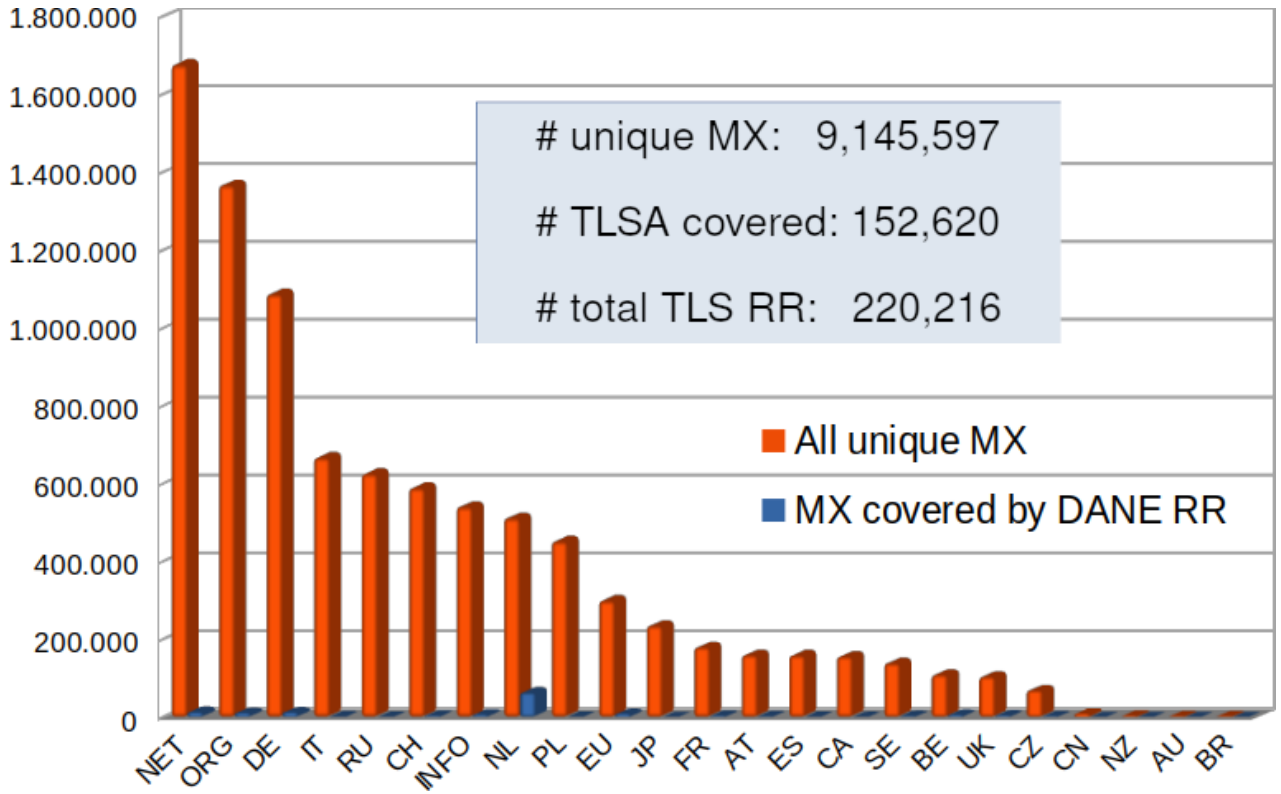

Abbildung: Anzahl der 'unique' MTAs per TLD und ihre TLSA-Abdeckung.

↪ Zur Erinnerung:

Der Inhalt des MX-Records muss nicht auf einen MTA in der eigenen Domäne verweisen!
Es kann mehr als ein TLSA-Record pro Domain-Name vergeben werden.

Ergebnisse des zweiten Schritts: Genauerer Blick

Frage: Kann das 'Rauschen' bei der TLSA-Abdeckung genauer quantifiziert werden?

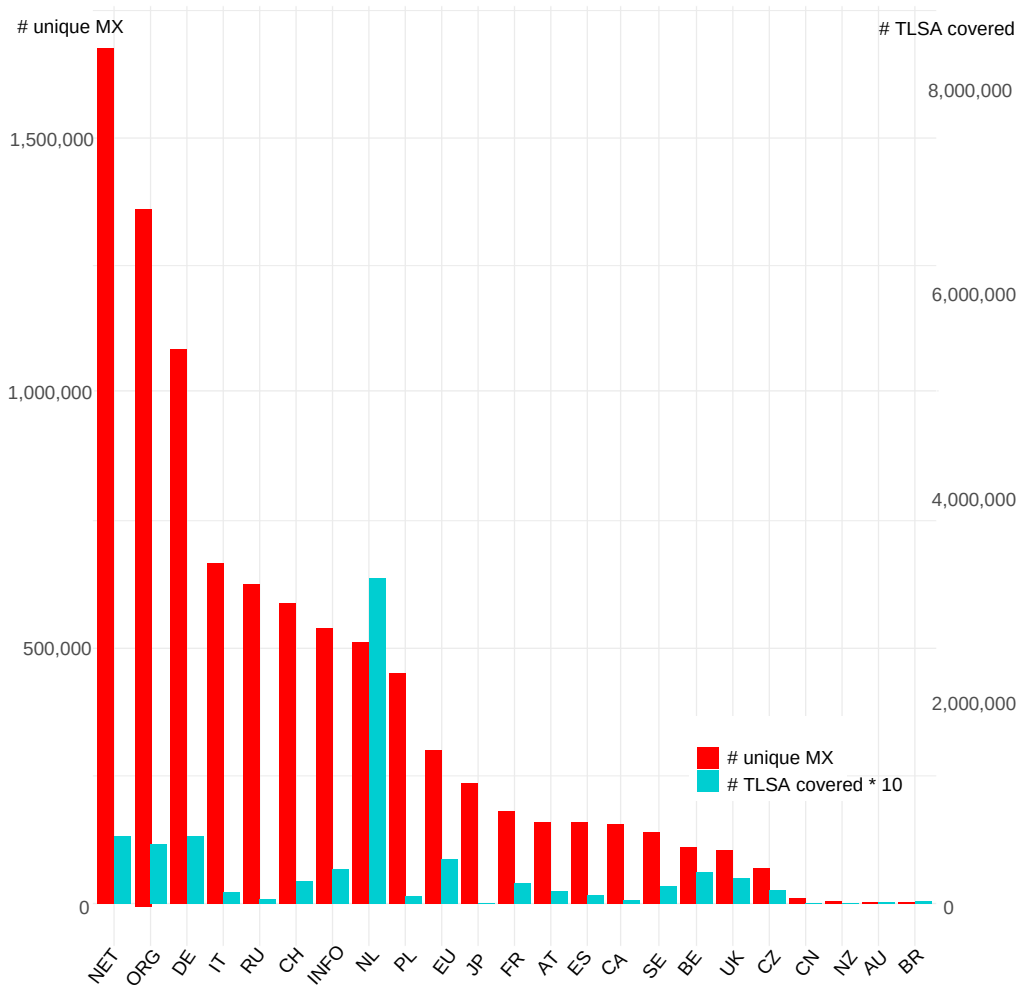

Abbildung: Anzahl der 'unique' MTAs per TLD und ihre TLSA-Abdeckung × 10.

↪
- Der Grad der TLSA-Abdeckung für 'unique' MTAs beträgt etwa 1.67%.
- Der Abdeckungsgrad für alle betrachteten Domains liegt aber lediglich bei 0.38%!

Detailanalyse der TLSA-Responses

Frage: Welche Detailinformationen erhalten wir aus den TLSA-Antworten?

- Die *Usage*: (0) `PKIX-TA`, (1) `PKIX-EE`, (2) `DANE-TA`, (3) `DANE-EE`.
- Den *Selector*: (0) Fingerprints des X.509-Zertifikates, (1) Fingerprint des *Subject Public Key Identifier* (**SPKI**).
- den *Matching Type*: (0) Gesamtes X.509-Zertifikate, als (1) SHA-256 oder als (2) SHA-512 Hashwert.

Zudem bekommen wir

- den *Fingerprint* der X509-Zertifikate sowie
- die *Anzahl* der TLSA-Responses.

Beispiele für TLSA-Antworten (**dnstlsa**) für die AU-Domäne:

```
box.ninkeri.com.au=
Usage: [3], Selector: [1], Type: [1] e0e8272da8b3ecb7f820aaeb85be00e4e4c4fc552b864f39ef1e7ea2fd1c9a

box.ocollins.me=
Usage: [3], Selector: [1], Type: [1] 40d751567dd5e1e5d6bcf6c5ddae3ff5ccb1c78cc7a4b1d9f7dfc1b5b8792f

mail.naturaltherapiesandbeauty.com.au=
Usage: [3], Selector: [0], Type: [1] e6b216a0166da9ab95c0b509585413c53f238300068cdfd2a43f66d8d0fb11

mail.protonmail.ch=
Usage: [3], Selector: [1], Type: [1] 76bb66711da416433ca890a5b2e5a0533c6006478f7d10a4469a947acc8399
Usage: [3], Selector: [1], Type: [1] 6111a5698d23c89e09c36ff833c1487edc1b0c841f87c49dae8f7a09e11e97
```

TLSA-Responses – Usage

Frage: Wie wird die *Usage* eingesetzt, welche *Verifikations-Policy* soll angewandt werden?

Wir erinnern uns, die *Usage* sagt vereinfacht:

a) Soll das empfange X.509-Zertifikat für den betroffenen MTA per PKIX verifiziert werden?

b) Reicht die aus dem DNS (DNSSec) bezogene Informationen aus, dem X.509-Zertifikat zu vertrauen?

Zudem kann mitgeteilt werden,

- ob die gesamte Zertifikatskette im TLS-Handshake mitgegeben wird (-TA), oder
- ob das Zertifikat lediglich als Endpunkt (-EE) betrachtet werden soll.

Im letzten Fall ist dies vergleichbar einem *Zertifikats-Pinning*.

Da dies bereits in RFC 7672 vorgeschrieben wird gibt es einen klaren Gewinner:

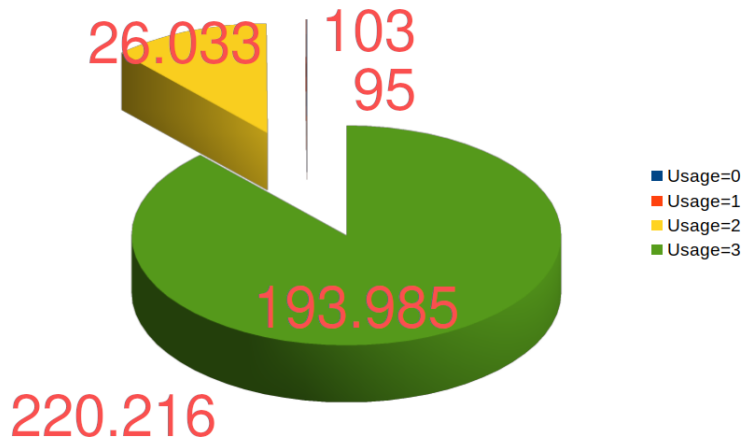

Abbildung: Verteilung der *Usage* für die empfangenen TLSA-Records; Usage (3): 88.09%, Usage (2): 11.82%. Die konkrete Anzahl der TLSA-Records ist in (roten) Zahlen beigefügt.

Ergebnis: Beim Betrieb von MTAs (Mail Exchanger) kann bei Nutzung von TLSA mit gutem Gewissen auf self-signed X.509-Zertifikate gesetzt werden, ohne dass Zwischenzertifikate benötigt werden.

TLSA-Responses – Selector

Sowohl der *Selector* als auch *Matching Type* stellen die technischen Parameter dar.

In Falle *Selector* wird gesagt, welche 'Teile' des X.509-Zertifikats abgedeckt werden (**Was?**), wobei zwei Möglichkeiten gegeben sind:

- Der *Standard-Fingerprint* des X.509-Zertifikats; wobei sich dieser bei jeder Zertifikats-Erneuerung ändert.
- Der Fingerprint des *Subject Public Key Identifier* **SPKI**; dieser bleibt unverändert; ausser, es wird ein komplett neues Zertifikat ausgerollt.

Aus einem betrieblichen Aspekt gesehen, ist Letzteres immun gegenüber einer Zertifikats-Erneuerung. Die Information im TLSA-Record bleiben unverändert über die Erneuerungen der Zertifikate hinweg.

Es ist also kein Wunder, dass dies die bevorzugte Methode ist:

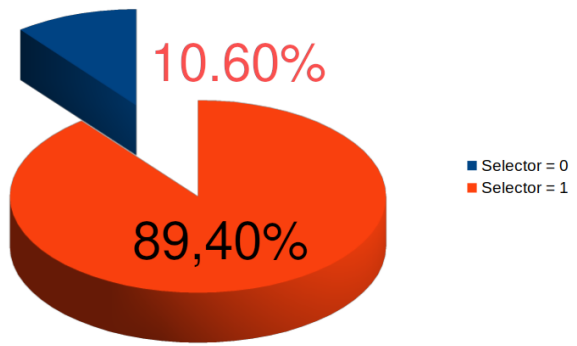

Abbildung: Verteilung des *Selectors* bei den TLSA-Records für Mail Exchanger.

Ergebnis: Die überwiegende Mehrheit nutzt den SPKI. Aus betrieblicher Sicht verständlich, aus Security-Sicht problematisch. Zudem liefert die OpenSSL-Routine zur Überprüfung des SPKI nicht das gewünschte Resultat.

TLSA Responses – Matching Type

Ergänzend zum *Selector*, sagt der *Matching Type* **wie** die IM TLSA-Record hinterlegt Information gebildet wurde und diese zu verwenden ist, um zum gleichen Ergebnis zu kommen:

- Type 0: Das gesamte X.509-Zertifikat (~ 1 kByte).
- Type 1: Der SHA-256 Hashwert (32 Byte).
- Type 2: Der SHA-512 Hashwert (64 Byte).

Im Prinzip ist die Angabe überflüssig, da sich der Verwendungszweck bereits aus der Länge des TLSA-Records ergibt, die man kennt.

Beim Einsatz von DNSSec, also TLSA in der Erweiterung DANE, sind wesentlich mehr Informationen zu übertragen, als die wenigen Byte bei den hier gegebenen Hashwerten.

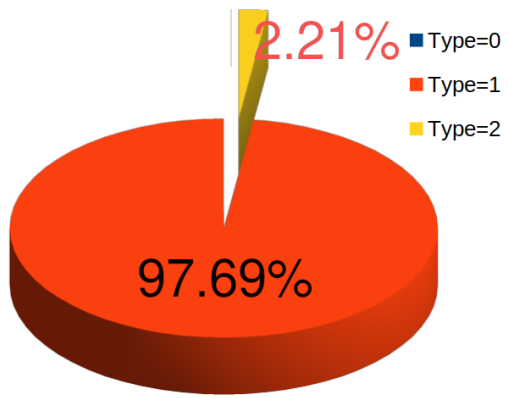

Abbildung: Verteilung des *Matching Type* bei den TLSA-Records für Mail Exchanger.

Ergebnis: Es ist nicht überraschend, dass die DNS-Operatoren SHA-256 Hashes bevorzugen; eventuell auch, um die Grösse der DNSSec-Responses zu minimieren.

TLSA: Resultate

Einsatz von TLSA-Records für die Unterstützung von TLS bei Mail-Exchangern:

- Die Abdeckung von Mail-Exchangern mit DANE/TLSA-Records ist für die untersuchten TLSs immer noch marginal; mit Ausnahme von `NL`.
- Der Abdeckungsgrad von TLSA-Records ist eng korreliert mit dessen Einsatz bei wenigen grossen MX-Providern.

Technische Nutzung von TLSA-Records:

- Die *Usage*-Policy in den TLSA-Records bei Mail Exchangern verlangt von den Mailclients nicht, eine PKIX-Verifikation vorzunehmen, sondern erlaubt auch insbesondere die Nutzung von *self-signed* X.509-Zertifikaten.
- Die X.509-Zertifikate werden in der Regel über ihren *Subject Public Key Identifier* SPKI und einen SHA-256 Hashwert qualifiziert.
- Unterschiede in der TLSA-Nutzung zwischen den TLDs sind vernachlässigbar, da den Empfehlungen von RFC 7672 entsprochen wird.

↪ Diese Ergebnisse komplementieren die periodischen TLSA-Surveys von *Viktor Dukhovni*.

TLSA und DANE: Sicherheitsgewinn

MX-Delegation:

- Bei der Nutzung von MX-Records für Internet-Domänen beobachten wir eine umfangreiche 'MX-Delegation'.
- Die dominierenden Anbieter wie Google und Microsoft bieten in grossem Umfang Mail-Exchanger Dienste an (`gmail.com`, `hotmail.com`, `outlook.com`); diese nutzen kein TLSA/DANE, sondern setzen auf DKIM (*DomainKey Identified Mails*).
- Kleinere Provider wie `mailbox.org` und `protonmail.ch` schwimmen als MX-Anbieter mit; bieten aber ergänzende Sicherheitsmechanismen, wie TLSA/DANE.

Mail-Confidentiality:

- In allen Fällen ist zu berücksichtigen, dass beim SMTP unser Mailverkehr in Klartext in der Mail-Queue liegt, obwohl natürlich TLS für den Transport genutzt wird.
- Dies ist unabhängig davon, ob das genutzte Postfach verschlüsselt angeboten wird.
- Abhilfe schaffen PGP oder S/MIME mit einer Ende-zu-Ende-Verschlüsselung.

TLSA und DNSSec

Frage: Braucht TLSA unbedingt DNSSec → DANE?

- Eine *Policy* per DNS vorzuschreiben (wie bei DANE), halte ich nicht für gewinnbringend: DNS ist ein Informationssystem; kein *Policy-Provider*.
- DNS-Antworten können vertraulich und gesichert auf ihre Nachrichtenintegrität auch über andere Mechanismen übertragen werden.
- Mein **djbdnscurve6** bietet *CurveDNS* als Applikations-spezifisches Verschlüsselungsprotokoll. Der entsprechende IETF-Draft von *Matthew Dempsky* wurde aber nicht weiter verfolgt.
- Der Inhalt des TLSA-Responses ist nicht deshalb *per-se* falsch, falls er nicht per DNSSec überprüft wurde.
- Das X.509-Zertifikat sollte dann noch ergänzenden Checks unterzogen werden:
  - Matching von DN/SAN mit dem MTA-*Hostname*.
  - Gerade bei TLSA wird dies aber häufig durch Wildcard-X.509-Zertifikate unterlaufen.

- Die *DNS PRIVate Working Group* (`https://datatracker.ietf.org/wg/dprive/about/`) verfolgt mehrere Ansätze zur DNS-Verschlüsselung: DoHTTPS, DoTLS, DoQ(uic).
- Hierbei geht es auch darum, das *parvasive Monitoring* des DNS-Verkehrs durch Dritte zu erschweren [RFC 7258] → *DNS4EU*.

TLSA und das DNS4EU-Projekt

Das DNS4EU-Projekt ist mit dem 16. Januar 2023 in seine Realisierungsphase getreten:

- Die Firma *Whalebone* hat die Ausschreibung gewonnen.
- Es soll nun ein 'europäischer' zentraler DNS-Resolverdienst angeboten werden.
- Wesentliches Kennzeichen ist eine 'split-horizon' Arbeitsweise, d.h. die übermittelten Antworten müssen nicht unbedingt von den autoritativen Nameservern stammen, sondern können 'umgebogen' werden.

Frage: Welche Informationen 'leaken' wir denn beim SMTP-Mailverkehr; gerade auch beim Einsatz von DNSSec?

TLSA und das DNS4EU-Projekt

Das DNS4EU-Projekt ist mit dem 16. Januar 2023 in seine Realisierungsphase getreten:

- Die Firma *Whalebone* hat die Ausschreibung gewonnen.
- Es soll nun ein 'europäischer' zentraler DNS-Resolverdienst angeboten werden.
- Wesentliches Kennzeichen ist eine 'split-horizon' Arbeitsweise, d.h. die übermittelten Antworten müssen nicht unbedingt von den autoritativen Nameservern stammen, sondern können 'umgebogen' werden.

Frage: Welche Informationen 'leaken' wir denn beim SMTP-Mailverkehr; gerade auch beim Einsatz von DNSSec?

Mail client DNS lookup (MX)		MTA DNS lookup	
MX	MTA hostname for domain	PTR	'inverse IP name'
A/AAAA	IPv4/IPv6 address of MTA	A/AAAA	IP-Address mail client
TLSA	X.509 cert fingerprint of MTA	RBL	Relay Blacklist
DKIM	Public key of MTA	EHELO Greeting	Client Greeting domain name
		Mail From	Domain of Originator
		DKIM	SDID of sending MX

TLSA und das DNS4EU-Projekt

Das DNS4EU-Projekt ist mit dem 16. Januar 2023 in seine Realisierungsphase getreten:

- Die Firma *Whalebone* hat die Ausschreibung gewonnen.
- Es soll nun ein 'europäischer' zentraler DNS-Resolverdienst angeboten werden.
- Wesentliches Kennzeichen ist eine 'split-horizon' Arbeitsweise, d.h. die übermittelten Antworten müssen nicht unbedingt von den autoritativen Nameservern stammen, sondern können 'umgebogen' werden.

Frage: Welche Informationen 'leaken' wir denn beim SMTP-Mailverkehr; gerade auch beim Einsatz von DNSSec?

Mail client DNS lookup (MX)		MTA DNS lookup	
MX	MTA hostname for domain	PTR	'inverse IP name'
A/AAAA	IPv4/IPv6 address of MTA	A/AAAA	IP-Address mail client
TLSA	X.509 cert fingerprint of MTA	RBL	Relay Blacklist
DKIM	Public key of MTA	EHELO Greeting	Client Greeting domain name
		Mail From	Domain of Originator
		DKIM	SDID of sending MX

Vielen Dank für Ihre Aufmerksamkeit!

Quellen, Referenzen und Weiterführendes


Hoffmann, E.: *djbware*, https://www.fehcom.de/ipnet/djbware.html


Hoffman, P, Schlyter, J: *RFC 6698: The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA*, https://datatracker.ietf.org/doc/html/rfc6698


Dukhovni, V. and Hardaker, W.: *RFC 7671: The DNS-Based Authentication of Named Entities (DANE) Protocol: Updates and Operations Guidance*, https://datatracker.ietf.org/doc/html/rfc7671


Reiser, H. Feuchtinger, D. Hommel, W., Schmidt, B., Storz, B. *E-Mail made in Science — Sicherheit für den E-Mail Transport mit DANE TLSA*
https://www.degruyter.com/document/doi/10.1515/pik-2015-0004/html?lang=de


Feuchtinger, D, Reier, H. Schmidt, B; 11. DFN-Forum Kommunikationstechnologien DNSSEC als Alternative zur klassischen CA https://dl.gi.de/bitstream/handle/20.500.12116/16578/DFN-Forum-Proceedings-005.pdf?sequence=1&isAllowed=y


Koetter, P. B.: *Sein, wo das Internet entsteht: Der RFC für DANE over SMTP*
https://guug.de/wp-content/uploads/2020/11/uptimes_2015-02.pdf


Lee, H, Rijswijk-Deiy, R. van, Asiq, I., Müller, M., Chung, T. *Under the Hood of DANE Mismanagement in SMTP* https://www.usenix.org/system/files/sec22summer_lee.pdf


Zhu, I., Wessels, D., Mankin, A. and Heidmann, J.: *Measuring DANE TLSA Deployment*,
https://link.springer.com/chapter/10.1007%2F978-3-319-17172-2_15


Hoffmann, E.: *DNS TLSA Survey*, https://github.com/ErwinHo/DNS_TLSA_Survey/releases/tag/v1.0


Dukhovni, V.: *Update on stats 2021-11*,
https://www.mail-archive.com/dane-users@sys4.de/msg00473.html


Encrypted DNS *DNS4EU*, https://419.consulting/encrypted-dns/f/a-new-dns-service-for-europe

Provisionierung der TLSA-Records

Zum Schluss wollen wir noch ermitteln, wie viele TLSA-Records pro MTA ins DNS gestellt werden. Dies zu tun, kann mehrere Gründe aufweisen:

- Das gleiche X.509-Zertifikate wird über verschiedene technische Attribute abgebildet.
- Unterschiedliche TLSA-Einträge dienen zur Unterstützung des *key rollover* und es finden sich *retired*, *aktuelle* und ggf. *zukünftig* gültige Zertifikate hier.

Die Antwort, die man bekommt, lautet :

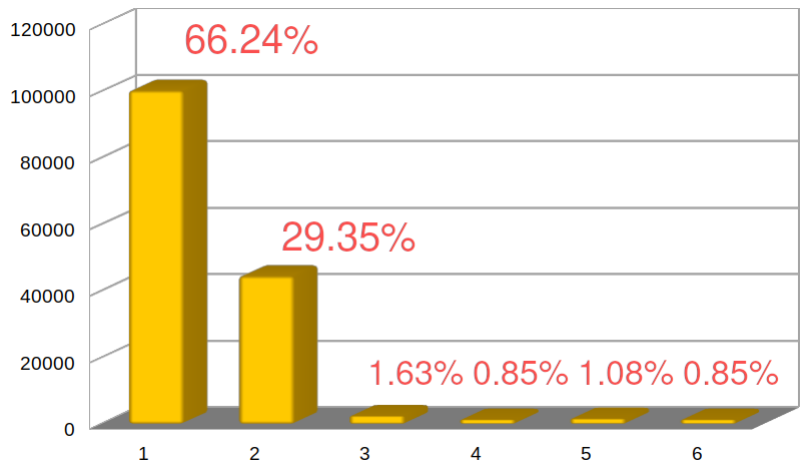

Abbildung: Häufigkeiten der Anzahl von TLSA-Records pro MX-DNS.

↪ Somit nutzen etwa 1/3 der Domain-Administratoren die Möglichkeit, mehrere TLSA-Einträge bereit zu stellen. Im Mittel sind dies 1.44 pro DNS-MX Eintrag.

Ablauf des DNS/TLSA-Lookups für Email an `mpp.mpg.de` im dnscache-Server

```
2023-01-13 16:36:27.699456500 u.query 138395 7f000001:6b63:db50 5 mpp.mpg.de.
2023-01-13 16:36:27.699511500 cached ns mpp.mpg.de. iws132a.mpp.mpg.de.
2023-01-13 16:36:27.699514500 cached ns mpp.mpg.de. dns-3.dfn.de.
2023-01-13 16:36:27.699516500 cached 1 iws132a.mpp.mpg.de.
2023-01-13 16:36:27.699518500 cached 1 dns-3.dfn.de.
2023-01-13 16:36:27.700203500 tx 0 5 mpp.mpg.de. mpp.mpg.de. - c1ae4b3a 866b02b4
2023-01-13 16:36:27.714936500 nodata c1ae4b3a 3600 5 mpp.mpg.de.
2023-01-13 16:36:27.714939500 stats 138395 25387712 1 0 0
2023-01-13 16:36:27.715097500 u.sent 138395 28
2023-01-13 16:36:27.715493500 u.query 138396 7f000001:2694:302a 15 mpp.mpg.de.
2023-01-13 16:36:27.715497500 cached 15 mpp.mpg.de.
2023-01-13 16:36:27.715499500 u.sent 138396 49
2023-01-13 16:36:27.715501500 u.query 138397 7f000001:f605:8cf1 28 smtp.mpp.mpg.de.
2023-01-13 16:36:27.715526500 cached 28 smtp.mpp.mpg.de.
2023-01-13 16:36:27.715527500 u.sent 138397 33
2023-01-13 16:36:27.715787500 u.query 138398 7f000001:657e:7d84 1 smtp.mpp.mpg.de.
2023-01-13 16:36:27.715790500 cached 1 smtp.mpp.mpg.de.
2023-01-13 16:36:27.715792500 u.sent 138398 49
2023-01-13 16:36:27.954015500 u.query 138399 7f000001:1b5f:399d 5 _25._tcp.smtp.mpp.mpg.de.
2023-01-13 16:36:27.954078500 cached ns mpp.mpg.de. iws132a.mpp.mpg.de.
2023-01-13 16:36:27.954081500 cached ns mpp.mpg.de. dns-3.dfn.de.
2023-01-13 16:36:27.954083500 cached 1 iws132a.mpp.mpg.de.
2023-01-13 16:36:27.954085500 cached 1 dns-3.dfn.de.
2023-01-13 16:36:27.954088500 tx 0 5 _25._tcp.smtp.mpp.mpg.de. mpp.mpg.de. - 866b02b4 c1ae4b3a
2023-01-13 16:36:27.966088500 nxdomain 866b02b4 3600 _25._tcp.smtp.mpp.mpg.de.
2023-01-13 16:36:27.966190500 u.sent 138399 42
2023-01-13 16:36:27.966548500 u.query 138400 7f000001:435d:c2b8 52 _25._tcp.smtp.mpp.mpg.de.
2023-01-13 16:36:27.966552500 cached nxdomain _25._tcp.smtp.mpp.mpg.de.
```

OpenSSL Fallen

man `x509_digest`:

```
SYNOPSIS
       #include <openssl/x509.h>

       int X509_digest(const X509 *data, const EVP_MD *type, unsigned char *md,
                       unsigned int *len);

       int X509_pubkey_digest(const X509 *data, const EVP_MD *type,
                              unsigned char *md, unsigned int *len);

...

DESCRIPTION
       X509_pubkey_digest() returns a digest of the DER representation of the
       public key in the specified X509 data object.  All other functions
       described here return a digest of the DER representation of their
       entire data objects.

...
```

↪ `X509_pubkey_digest()` liefert entgegen der Beschreibung nicht den Public Key des X.509 Zertifikats zurück!

Differenz zu Dukovni

Dokuvni Status Report 11/2021[2]:

> *The number of published MX host TLSA RRsets found is 16,295 (16,101 last month). These cover 16,562 distinct MX hosts (16,358 last month, some MX hosts share the same TLSA records through CNAMEs).*

```
This month                    Last month
 ----------                    ----------
 1230165 one.com               1219713 one.com
  272727 hostpoint.ch           270842 hostpoint.ch
  154952 transip.nl             154249 transip.nl
  154347 infomaniak.ch          152372 infomaniak.ch
  149718 argewebhosting.nl      150807 argewebhosting.nl
  106004 domeneshop.no          105814 domeneshop.no
   98029 webhostingserver.nl     98302 webhostingserver.nl
   95100 loopia.se               94851 loopia.se
   71946 forpsi.com              71517 forpsi.com
   48270 zxcs.nl                 46431 active24.com        This month          Last month
 ----------                    ----------
    ....
   23612 hosting2go.nl           23884 hosting2go.nl
   22118 protonmail.ch           21623 protonmail.ch
```

[2] https://www.mail-archive.com/dane-users@sys4.de/msg00473.html