

Proxmox Mail Gateway

- [Informationen](#)
- [TLS 1.0 und 1.1 deaktivieren](#)
- [DANE einrichten](#)
- [Eigene Blacklisten \(RBL\) integrieren](#)

Informationen

DKIM

Bei einem 2048 Bit langen Schlüssel kann es sein, dass dieser zu lang wird - dieser muss dann gesplittet werden. Dies kann mit Tools wie z.B. <https://www.mailhardener.com/tools/dns-record-splitter> erledigt werden. Prüfen Sie jedoch unbedingt danach, ob der DKIM Eintrag korrekt ist.

TLS 1.0 und 1.1 deaktivieren

<https://tgraeber.de/proxmox-mail-gateway-7-tls/>

Die Protokolldefintionen jedoch von hier übernehmen:

<https://schroederdennis.de/tipps-tricks/mailcow-smtp-smtpd-tls-1-0-tls1-1-deaktivieren/>

Anschließend kann man mit Internet.nl testen, ob die Einstellungen passen.

DANE einrichten

Um DANE richtig einzurichten, müsst ihr die Hashes eures Zertifikates, des Intermediate- und Root-Zertifikates erstellen.

Dafür ist es wichtig, dass ihr die richtigen Zertifikate benutzt, diese ermitteln wir mit checktls.com. Wichtig ist auch später dass ihr die TLSA Records mit 3 1 1 (eignes Zertifikat) und 2 1 1 (Root und Intermediate) erstellt, dadurch erzwingt ihr einen Fallbackmodus - dieser sorgt dafür dass DANE weiterhin gültig bleibt, auch wenn euer eigenes Zertifikat nach 90 Tagen erneuert wurde.

Geht dazu wie folgt vor:

- Ermittelt die Root und Intermediate Zertifikate, geht dazu zu <https://www.checktls.com/TestReceiver> und prüft euren Mailserver.
- In der Ausgabe seht ihr welche Root (bei mir in dem Falle X1) und Intermediate (in meinem Falle R10). Die URL der entsprechenden PEM Dateien findet ihr unter <https://letsencrypt.org/certificates/>.
- Ladet diese nun auf die PMR in einen temporären Ordner (z.B. `wget -P ./ kodierte_url`).
- Erstellt nun in dem Verzeichnis das Serverzertifikat (kopiert den Inhalt aus dem Zertifikat unter `PMR > Certificates > pmg-tls.pem > View Certificates` und fügt den Inhalt in eine neue erstellte Datei `server.pem` mit `vi` oder `nano`).
- Erstellt nun die entsprechenden Hashes dieses Zertifikates mit dem Befehl:

```
openssl x509 -in zertifikatesnamen.pem -noout -pubkey | openssl pkey -pubin -outform DER |  
openssl sha256
```

- Mit diesem Eintrag erstellt ihr dann die entsprechenden TLSA Records im DNS - z.B. `_25._tcp.mx` (Port, Protokoll, Hostname) - danach dann 2 1 1 Wert (2 für Root / Intermediate, 3 für Endzertifikat)
- Bei mir würden die Einträge in dem Falle wie folgt aussehen:

```
root@mx:~/le# dig _25._tcp.mx.freesoc.de IN TLSA  
  
; <<>> DiG 9.18.24-1-Debian <<>> _25._tcp.mx.freesoc.de IN TLSA  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2550  
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:
```

```

; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;_25._tcp.mx.freesoc.de.          IN      TLSA

;; ANSWER SECTION:
_25._tcp.mx.freesoc.de. 83132   IN      TLSA    3 1 1
07A5A87B97B38EAD67BD1A9960D3535C18A36DE1FA6C725A1ADEA1FF 5074795F
_25._tcp.mx.freesoc.de. 83132   IN      TLSA    2 1 1
8D02536C887482BC34FF54E41D2BA659BF85B341A0A20AFADB5813DC FBCF286D
_25._tcp.mx.freesoc.de. 83132   IN      TLSA    2 1 1
E5545E211347241891C554A03934CDE9B749664A59D26D615FE58F77 990F2D03

;; Query time: 0 msec
;; SERVER: 192.168.70.246#53(192.168.70.246) (UDP)
;; WHEN: Fri Aug 30 09:06:20 CEST 2024
;; MSG SIZE rcvd: 192

```

Wartet die TTLS des DNS Servers ab und prüft dann ob alles richtig ist mit checktls und internet.nl

Info für die Werte 3 1 1 bzw. 2 1 1 (Quelle

https://dokuwiki.tachtler.net/doku.php?id=tachtler:let_s_encrypt_-_tlsa-record_-_dane :

Die erste Zahl ist ein Wert von 0 bis 3:

- **0:** Der Hash gehört der Zertifizierungsstelle die Zertifikate für diesen Host ausstellen darf. Der Client muss die Zertifizierungsstelle kennen oder diese muss von einer vertrauten Zertifizierungsstelle unterschrieben sein.
- **1:** Der Hash gehört dem Serverzertifikat. Es muss von einer Zertifizierungsstelle unterschrieben sein der vom Client vertraut wird.
- **2:** Der Hash gehört einer Zertifizierungsstelle die Zertifikate für diesen Host ausstellen darf. Der Client soll Ihr Vertrauen auch wenn sie ihm Unbekannt und von keiner bekannten Zertifizierungsstelle unterschrieben ist.
- **3:** Der Hash gehört dem Serverzertifikat und der Client soll diesem ohne weitere Prüfung der Vertrauenskette trauen.

Die Zweite Zahl kann 0 oder 1 sein und gibt an wie der Hash überprüft wird:

- **0:** Es wird ein Hash vom kompletten Zertifikat erstellt.
- **1:** Es wird nur ein Hash vom Public Key und des Algorithmus erstellt.

Die Dritte Zahl enthält einen wert von 0 bis 2:

- **0**: Der Hash enthält das komplette Zertifikat (nicht empfohlen).
- **1**: Der Hash enthält einen SHA-256 Hash.
- **2**: Der Hash enthält einen SHA-512 Hash.

Weitere Informationen:

- <https://github.com/internetstandards/toolbox-wiki/blob/main/DANE-for-SMTP-how-to.md>
- https://dokuwiki.tachtler.net/doku.php?id=tachtler:let_s_encrypt_-_tlsa-record_-_dane
- <https://letsencrypt.org/certificates/>
- <https://internet.nl>
- <https://www.checktls.com/TestReceiver>
- <https://dane.sys4.de/smtp>

Eigene Blacklisten (RBL) integrieren

Eigene Blacklisten wie Spamhaus usw. lassen sich integrieren.

Gehen Sie dazu wie folgt vor:

- Erstellen Sie sich einen kostenlosen Account bei Spamhaus (<https://portal.spamhaus.com>)
- Nach dem einloggen gehen Sie unten unter Spamhaus Services > Data Query Services auf das Icon mit den Zahnrädern.
- Unter Access sehen sie die entsprechenden URLs für die Abfragen der entsprechenden Services.

Wenn Sie die URL haben, dann kommt die Anpassungen in der Proxmox Mail Gateway Konfiguration:

- Loggen Sie sich im UI ein.
- Gehen Sie zu Configuration > Email Proxy > Options.
- Bearbeiten Sie den Eintrag DNSBL Sites und fügen Sie die vorgenannte URL ein und speichern den Eintrag.

Sie können auch weitere Provider verwenden, wie z.B.

```
xxxxxxxxx.zen.dq.spamhaus.net,bl.spamcop.net,b.barracudacentral.org
```

Sie sollten jedoch unbedingt die nächste Zeit den Emailfluss beobachten, da es Probleme geben kann - insbesondere wenn man Public DNS Server wie google usw. verwendet.