SECURE MAILGATEWAY OTC

INHALT

1. Einleitung Secure Mail Gateway 2
   1.1. Beschreibung Mail-Gateway / Grundsatz 2
   1.2. Leistungsumfang 2
   1.3. Schaubild 3
2. Sicherheitsmechanismen 3
   2.1. Good / Bad Gateway / Anti Spam 3
   2.2. Virenprüfung 3
   2.3. Blacklist Monitoring 4
   2.4. Sender Domain Check 4
   2.5. Forced TLS 4
   2.6. DKIM Signierung 4
3. Verschiedene Einrichtungsoptionen 4
   3.1. Smarthost 4
   3.2. Anmelde-Daten 4
   3.3. T-Systems eMail Protect Pro (bei Anforderung von Eingangspost) 5
4. Zusätzliche Informationen 5
   4.1. SPF - Sender Policy Framework 5
   4.2. DNS - Konfiguration Absender Mail Domains 5
5. Installationsbeispiele / Grundinstallation 6
   5.1. Beispiel Postfix Ubuntu 6
   5.2. Beispiel EXIM Ubuntu 9
6. Installation / Anpassungsbeispiele 11
   6.1. Postfix 11
   6.2. EXIM 12

1. EINLEITUNG SECURE MAIL GATEWAY

1.1. BESCHREIBUNG MAIL-GATEWAY / GRUNDSATZ

Wenn Sie regelmäßig große Mengen von E-Mails von Ihrem E-Mail-Server versenden, können diese durch Anti-Spam-Datenbanken schnell auf die Liste der unerwünschten Absender gesetzt werden - mit anderen Worten, sie werden auf die schwarze Liste („Blacklist“) gesetzt. In diesem Fall wird die IP-Adresse, von der unerwünschte E-Mails gesendet werden, als nicht vertrauenswürdig eingestuft, und E-Mails, die von dort gesendet werden, werden nicht mehr zugestellt.

Um solche unerwünschten Konsequenzen zu vermeiden, arbeitet Open Telekom Cloud mit einem sicheren Mail-Gateway-Dienst. Dieser Dienst stellt die offizielle und sichere Zustellung ausgehender E-Mails sicher und beugt eine Aufnahme in die Blacklists, der verschiedenen Blacklist-Provider vor. Die originäre IP-Adresse des Absenders ist nicht direkt erkennbar da immer das Secure Mail Gateway die Kommunikation durchführt.

Aus Sicherheitsgründen wird in der Open Telekom Cloud nur diese Art der E-Mail-Kommunikation aktiviert.

Das Secure Mailgateway stellt die Schnittstelle für das Versenden von Mail aus der OTC ins Internet dar. Die Secure Mailgateway Umgebung besteht aus drei Mail-Servern, die zwecks Redundanz auf drei Standort verteilt sind. Die Standorte sind: eu-de-01 / eu-de-02 / eu-de-03

1.2. LEISTUNGSUMFANG

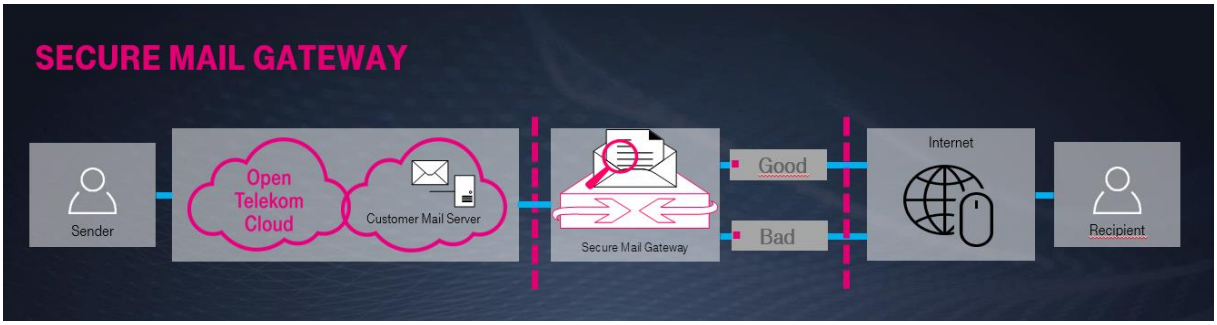
Inklusivleistungen:

- Anti-Virus und Anti-Spam
- Good Bad Gateway

Info:
Um die Vertrauenswürdigkeit und Leistung des Secure Mailgateways zu gewährleisten, kann das Mailing pro Kunde auf 100 Mails/min Open Telekom Cloud seitig begrenzt werden.
Sofern das Limit aktiviert und überschritten ist, erhält der Absender die Fehlermeldung: „451 Ratelimit reached“.
Das Limit ist standardmäßig nicht aktiviert.

Hinweis: Aufgrund der Verfügbarkeit der Zielsysteme kann keine Garantie für die rechtzeitige Lieferung übernommen werden. Internetdienstanbieter können E-Mail-Nachrichten verwerfen oder verlieren, Empfänger können aus Versehen eine falsche Adresse angegeben haben oder Internetdienstanbieter können die Nachricht, wenn sie der Empfänger nicht erhalten möchte, ablehnen oder stillschweigend verwerfen.

1.3. SCHAUBILD



2. SICHERHEITSMECHANISMEN

2.1. GOOD / BAD GATEWAY / ANTI SPAM

Das Good Bad Routing welches Spam Mails intern an das Bad Gateway umleitet hilft dabei, dass die Good Gateways nicht auf externen Blacklisten landen. Das Good Bad Routing hilft somit dabei, dass sich Mails der Kunden gegenseitig nicht beeinflussen.

„Good“: Wenn die E-Mail unauffällig ist, wird sie über eines der „Good Gateways“ versendet.

„Bad“: Im Falle eines Spam-Verdachts wird die E-Mail über eines der „Bad Gateways“ versendet (höhere Wahrscheinlichkeit einer Sperrliste. Wenn das Bad Gateway auf einer Blacklist steht wird geprüft ob dies wieder behoben werden kann. (Es gibt dazu jedoch keine Garantie)

Auf dem Secure Mailgateway werden alle Mails auf Spam überprüft. Als Spam eingestufte Mails werden innerhalb der Secure Mailgateway Umgebung an das BAD Gateway umgeleitet und von dort aus ins Internet gesendet.

Hinweis: Das Secure Mail Gateway “otc-de-mta03.mms.t-systems-service.com” ist das Bad Gateway. Es werden nur Mails über dieses Gateway verschickt, wenn Mails von den Systemen der Open Telekom Cloud bereits als Spam klassifiziert wurden. Es ist daher davon auszugehen, dass dieses System (Bad Gateway) unter Umständen auf sogenannten „Blacklits“ gelistet wird. Reguläre Mails werden niemals über das Bad Gateway ins Internet gesendet, sollte ein NDR (Non Delivery Report) vom Bad Gateway an den Kunden gehen, ist auf Kundenseite zu prüfen warum diese Mails als Spam klassifiziert wurde.

2.2. VIRENPRÜFUNG

Auf den Secure Mailgateways werden alle Mails auf Viren überprüft. Mails die einen Virus enthalten werden mit einer entsprechenden Info abgelehnt und dazu ein NDR (Non Delivery Report) an den Absender der Mail geschickt.

Hinweis: Virenprüfung findet nur „ausgehend“ statt. Der Absender wird per Infomail darüber informiert.

2.3. BLACKLIST MONITORING

Eine automatisierte Blacklist-Überwachung bei Blacklist-Anbietern wurde eingerichtet: Das Skript wird täglich um 5 Uhr morgens ausgeführt. Im Falle einer festgestellten Sperrliste wird FMB Change-OpenTelekomCloud DD-OTC-SDM@telekom.de informiert; Entfernungsanfragen bei Blacklist-Anbietern werden sofort erstellt. Je nach Bearbeitungszeit des Blacklist-Anbieters wird diese Sperre entsprechend entfernt.

2.4. SENDER DOMAIN CHECK

Die Mail Domain, die zum Versenden von Mails aus der Open Telekom Cloud benutzt wird, muss eine valide Public Domain sein. (Das bedeutet die Domain muss einen Public MX Record besitzen oder zumindest einen Public A Record.) Mails mit Absender Adressen, bei denen diese Kriterien nicht erfüllt sind, werden verworfen.

Info: Fehlermeldung bei Ablehnung: 550 Sender Domain must resolve

2.5. FORCED TLS

Es kann auf Basis der Absender und Empfänger Domain eine forcierte TLS Kommunikation eingerichtet werden. Für Mails bei diesen eingerichteten Domänen ist dann die Verbindung zwischen den Mailserver zwingend TLS verschlüsselt. (Im Fehlerfall werden Mails dann direkt abgelehnt oder zurückgehalten, wenn keine TLS Verbindung mit dem externen Mailserver zustande kommt.)

2.6. DKIM SIGNIERUNG

Auf Basis der Absender Domain kann eine DKIM Signierung für ausgehende Mails konfiguriert werden.
(Dies kann über den Cloud Handling Support angefragt werden.)

3. VERSCHIEDENE EINRICHTUNGSOPTIONEN

Um das Secure Mailgateway nutzen zu können sind auf Kundenseite (Mailserver oder Applikation des Kunden) die folgenden Einrichtungs-Schritte notwendig.

3.1. SMARTHOST

Der folgenden DNS A Record muss als SMARTHOST im Mailserver oder in der Anwendung gespeichert werden:

otc-de-out.mms.t-systems-service.com

Für die Anbindung von Systemen aus der Open Telekom Cloud an den SMARTHOST steht der **Port 25** zur Verfügung.
(Andere Ports sind auf dem sicheren Mail-Gateway für die Mailannahme nicht aktiv.)

3.2. ANMELDE-DATEN

Die Authentifizierung von Kunden mittels Benutzername ist zwingend erforderlich, um den Spam-Verdacht zu verringern und die Sicherheit durch TLS-Authentifizierung zu erhöhen. Der Benutzername wird Cloud Handling

Support bereitgestellt. (Kontaktadresse: service@open-telekom-cloud.com)
Der Kunden Kontakt erhält im Anschluss eine Registrierungsmail von den Secure Mail-Gateways, über die der Kunde sich sein Passwort selbst vergeben kann.

Kosten für Kunden: ~23€/Monat. Wird über OTC in Rechnung gestellt.

3.3. T-SYSTEMS EMAIL PROTECT PRO (BEI ANFORDERUNG VON EINGANGSPOST)

Dies ist ein direkter und vollständig verwalteter Service von T-Systems (kein OTC-Service!), der unter https://cloud.telekom.de/de/magenta-security/e-mail-protect-pro verfügbar ist.

Kontakt: Verfügbare Kontakte: FMB Secure-E-Mail secure-email@t-systems.com

Kundenkosten: sind beim Secure Mail-Team zu erfragen. Wird nicht über OTC in Rechnung gestellt.

4. ZUSÄTZLICHE INFORMATIONEN

4.1. SPF - SENDER POLICY FRAMEWORK

Sollte der Kunde für seine Email Domains einen SPF Record eingerichtet haben muss der Kunde den folgende DNS A Record in seinen SPF mit aufnehmen:

a:otc-de-spf.mms.t-systems-service.com

Info: Das Sender Policy Framework (kurz SPF) ist eine Technik, die das Fälschen des Absenders einer E-Mail erschweren soll.

Im A Record enthaltene IP Adressen:

Servername	IP	Typ
otc-de-mta01.mms.t-systems-service.com	80.158.29.159	Good Gateway
otc-de-mta02.mms.t-systems-service.com	80.158.29.232	Good Gateway
otc-de-mta03.mms.t-systems-service.com	80.158.30.229	BAD Gateway

Tabelle 1: SPF Record IPs

4.2. DNS - KONFIGURATION ABSENDER MAIL DOMAINS

Um eine korrekte Mailverarbeitung über das SMG zu gewährleisten ist es erforderlich für die vom Kunden verwendete/n Sender Mail Domain/s einen MX Record und / oder mindestens einen DNS A Record anzulegen. Im Idealfall sollte der MX Record auf einen Server zeigen mit dem Rückantwort Mails sowie NDRs (Non Delivery Reports) empfangen werden können.

Anmerkung: / Tipp:
Sollte kein Mailserver für die Sender Mail Domain vorhanden sein, ist es Sinnvoll beim Mailversand in die Mails den **Reply-To** Header einzufügen. Mit diesem lassen sich Rück-Antworten auf versendete Mails an ein beliebiges Postfach umleiten (z.B. Funktionsmailbox unterhalb einer vorhandenen Kunden Mail Domain).

5. INSTALLATIONSBEISPIELE / GRUNDINSTALLATION

5.1. BEISPIEL POSTFIX UBUNTU

(Quelle: ubuntu documentation - https://help.ubuntu.com/lts/serverguide/postfix.html)

Installation

To install postfix run the following command:

```
sudo apt install postfix
```

Simply press return when the installation process asks questions, the configuration will be done in greater detail in the next stage.

Basic Configuration

To configure postfix, run the following command:

```
sudo dpkg-reconfigure postfix
```

The user interface will be displayed. On each screen, select the following values:

1. Internet Site
2. mail.example.com
3. steve
4. mail.example.com, localhost.localdomain, localhost
5. No
6. 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128 192.168.0.0/24
7. 0
8. +
9. all

Replace mail.example.com with the domain for which you'll accept email, 192.168.0.0/24 with the actual network and class range of your mail server, and steve with the appropriate username.

Now is a good time to decide which mailbox format you want to use. By default Postfix will use mbox for the mailbox format. Rather than editing the configuration file directly, you can use the postconf command to configure all postfix parameters. The configuration parameters will be stored in /etc/postfix/main.cf file. Later if you wish to re-configure a particular parameter, you can either run the command or change it manually in the file.

To configure the mailbox format for Maildir:

```
sudo postconf -e 'home_mailbox = Maildir/'
```

This will place new mail in /home/username/Maildir so you will need to configure your Mail Delivery Agent (MDA) to use the same path.

SMTP Authentication

SMTP-AUTH allows a client to identify itself through an authentication mechanism (SASL). Transport Layer Security (TLS) should be used to encrypt the authentication process. Once authenticated the SMTP server will allow the client to relay mail.

1. Configure Postfix for SMTP-AUTH using SASL (Dovecot SASL):

```
2.  sudo postconf -e 'smtpd_sasl_type = dovecot'
3.  sudo postconf -e 'smtpd_sasl_path = private/auth'
4.  sudo postconf -e 'smtpd_sasl_local_domain ='
5.  sudo postconf -e 'smtpd_sasl_security_options = noanonymous'
6.  sudo postconf -e 'broken_sasl_auth_clients = yes'
7.  sudo postconf -e 'smtpd_sasl_auth_enable = yes'
8.  sudo postconf -e 'smtpd_recipient_restrictions = \
9.  permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination'
```

The smtpd_sasl_path configuration is a path relative to the Postfix queue directory.

10. Next, generate or obtain a digital certificate for TLS. See Certificates for details. This example also uses a Certificate Authority (CA). For information on generating a CA certificate see Certification Authority.

    MUAs connecting to your mail server via TLS will need to recognize the certificate used for TLS. This can either be done using a certificate from a commercial CA or with a self-signed certificate that users manually install/accept. For MTA to MTA TLS certficates are never validated without advance agreement from the affected organizations. For MTA to MTA TLS, unless local policy requires it, there is no reason not to use a self-signed certificate. Refer to Creating a Self-Signed Certificate for more details.

11. Once you have a certificate, configure Postfix to provide TLS encryption for both incoming and outgoing mail:

```
12. sudo postconf -e 'smtp_tls_security_level = may'
13. sudo postconf -e 'smtpd_tls_security_level = may'
14. sudo postconf -e 'smtp_tls_note_starttls_offer = yes'
15. sudo postconf -e 'smtpd_tls_key_file = /etc/ssl/private/server.key'
16. sudo postconf -e 'smtpd_tls_cert_file = /etc/ssl/certs/server.crt'
17. sudo postconf -e 'smtpd_tls_loglevel = 1'
18. sudo postconf -e 'smtpd_tls_received_header = yes'
19. sudo postconf -e 'myhostname = mail.example.com'
```

20. If you are using your own Certificate Authority to sign the certificate enter:

```
21. sudo postconf -e 'smtpd_tls_CAfile = /etc/ssl/certs/cacert.pem'
```

Again, for more details about certificates see Certificates.

After running all the commands, Postfix is configured for SMTP-AUTH and a self-signed certificate has been created for TLS encryption.

Now, the file /etc/postfix/main.cf should look like this:

```
# See /usr/share/postfix/main.cf.dist for a commented, more complete

# version


smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)

biff = no


# appending .domain is the MUA's job.

append_dot_mydomain = no


# Uncomment the next line to generate "delayed mail" warnings

#delay_warning_time = 4h


myhostname = server1.example.com

alias_maps = hash:/etc/aliases

alias_database = hash:/etc/aliases

myorigin = /etc/mailname

mydestination = server1.example.com, localhost.example.com, localhost

relayhost =

mynetworks = 127.0.0.0/8

mailbox_command = procmail -a "$EXTENSION"

mailbox_size_limit = 0

recipient_delimiter = +

inet_interfaces = all

smtpd_sasl_local_domain =

smtpd_sasl_auth_enable = yes

smtpd_sasl_security_options = noanonymous

broken_sasl_auth_clients = yes

smtpd_recipient_restrictions =

permit_sasl_authenticated,permit_mynetworks,reject _unauth_destination

smtpd_tls_auth_only = no

smtp_tls_security_level = may
```

```
smtpd_tls_security_level = may

smtp_tls_note_starttls_offer = yes

smtpd_tls_key_file = /etc/ssl/private/smtpd.key

smtpd_tls_cert_file = /etc/ssl/certs/smtpd.crt

smtpd_tls_CAfile = /etc/ssl/certs/cacert.pem

smtpd_tls_loglevel = 1

smtpd_tls_received_header = yes

smtpd_tls_session_cache_timeout = 3600s

tls_random_source = dev:/dev/urandom
```

The postfix initial configuration is complete. Run the following command to restart the postfix daemon:

```
sudo systemctl restart postfix.service
```

Postfix supports SMTP-AUTH as defined in RFC2554. It is based on SASL. However it is still necessary to set up SASL authentication before you can use SMTP-AUTH.

5.2. BEISPIEL EXIM UBUNTU

(Quelle: ubuntu documentation . https://help.ubuntu.com/lts/serverguide/exim4.html)

Installation

To install exim4, run the following command:

```
sudo apt install exim4
```

Configuration

To configure Exim4, run the following command:

```
sudo dpkg-reconfigure exim4-config
```

The user interface will be displayed. The user interface lets you configure many parameters. For example, In Exim4 the configuration files are split among multiple files. If you wish to have them in one file you can configure accordingly in this user interface.

All the parameters you configure in the user interface are stored in /etc/exim4/update-exim4.conf.conf file. If you wish to re-configure, either you re-run the configuration wizard or manually edit this file using your favorite editor. Once you configure, you can run the following command to generate the master configuration file:

```
sudo update-exim4.conf
```

The master configuration file, is generated and it is stored in /var/lib/exim4/config.autogenerated.

At any time, you should not edit the master configuration file, /var/lib/exim4/config.autogenerated manually. It is updated automatically every time you run update-exim4.conf

You can run the following command to start Exim4 daemon.

```
sudo systemctl start exim4.service
```

SMTP Authentication

This section covers configuring Exim4 to use SMTP-AUTH with TLS and SASL.

The first step is to create a certificate for use with TLS. Enter the following into a terminal prompt:

```
sudo /usr/share/doc/exim4-base/examples/exim-gencert
```

Now Exim4 needs to be configured for TLS by editing /etc/exim4/conf.d/main/03_exim4-config_tlsoptions add the following:

```
MAIN_TLS_ENABLE = yes
```

Next you need to configure Exim4 to use the saslauthd for authentication. Edit /etc/exim4/conf.d/auth/30_exim4-config_examples and uncomment the plain_saslauthd_server and login_saslauthd_server sections:

```
 plain_saslauthd_server:
   driver = plaintext
   public_name = PLAIN
   server_condition = ${if saslauthd{{$auth2}{$auth3}}{1}{0}}
   server_set_id = $auth2
   server_prompts = :
   .ifndef AUTH_SERVER_ALLOW_NOTLS_PASSWORDS
   server_advertise_condition = ${if eq{$tls_cipher}{}{}{*}}
   .endif
#
 login_saslauthd_server:
   driver = plaintext
   public_name = LOGIN
   server_prompts = "Username:: : Password::"
   # don't send system passwords over unencrypted connections
   server_condition = ${if saslauthd{{$auth1}{$auth2}}{1}{0}}
   server_set_id = $auth1
   .ifndef AUTH_SERVER_ALLOW_NOTLS_PASSWORDS
   server_advertise_condition = ${if eq{$tls_cipher}{}{}{*}}
```

```
    .endif
```

Additionally, in order for outside mail client to be able to connect to new exim server, new user needs to be added into exim by using the following commands.

```
sudo /usr/share/doc/exim4-base/examples/exim-adduser
```

Users should protect the new exim password files with the following commands.

```
sudo chown root:Debian-exim /etc/exim4/passwd
sudo chmod 640 /etc/exim4/passwd
```

Finally, update the Exim4 configuration and restart the service:

```
sudo update-exim4.conf
sudo systemctl restart exim4.service
```

6. INSTALLATION / ANPASSUNGSBEISPIELE

6.1. POSTFIX

Quelle: https://wiki.ubuntuusers.de/Postfix/

Authentifizierung am Smarthost

Wenn der SMTP-Server auf dem Smarthost zum Versenden der Mail ein Passwort verlangt, muss die eben erstellte Konfiguration **/etc/postfix/main.cf** allerdings noch einmal editiert [3] und diese Zeilen eingefügt werden:

```
smtp_sasl_auth_enable = yes
# noplaintext weglassen, wenn Passwörter im Klartext übertragen werden müssen:
# (nicht empfohlen, nur wenn's anders nicht funktioniert)
smtp_sasl_security_options = noplaintext noanonymous
smtp_sasl_password_maps = hash:/etc/postfix/sasl_password
```

Wie in der Konfigurationsdatei ersichtlich, holt Postfix die Zugangsdaten aus der Datei **/etc/postfix/sasl_password** bzw. aus einer Datenbank, die aus der **sasl_password** generiert wird. Die Datei sollte man vorzugsweise mit folgendem Befehl erstellen, da sonst ein Umwandeln in eine Datenbank nicht immer möglich ist. Dazu muss man ein Terminalfenster öffnen [2] und den folgenden Befehl eingeben:

```
sudo touch /etc/postfix/sasl_password
```

Nun schreibt man seine Daten nach folgendem Muster in die Datei

```
smtp.mailanbieter.de username:ganzgeheimespasswort
```

Damit nicht gleich jeder das Passwort lesen kann, sollte man noch die Berechtigungen der Datei einschränken (eventuell ist das für Sicherungskopien oder die nachfolgend erzeugte Datenbank zu wiederholen):

```
sudo chmod 600 /etc/postfix/sasl_password
```

Jetzt muss noch die Datenbank erzeugt werden:

```
sudo postmap hash:/etc/postfix/sasl_password
```

Danach muss man postfix neu starten:

```
sudo /etc/init.d/postfix restart
```

6.2. EXIM

Hilfe → Quelle: https://wiki.debian.org/Exim

Beispiel-Quelle: https://somoit.net/linux/linux-exim-authenticated-and-tls-mail-through-smarthost

If you need to configure exim by editing the config file (instead of using dpkg-reconfigure), these are the related values:

```
dc_eximconfig_configtype='satellite'
dc_smarthost='smtp.bilbokoudala.lan::587'
```

Configure credentials to authenticate

Exim has a password file called *passwd.client* that allows configurin a list of credentials associated to each smarthost. In my debian 9, the full path is ***/etc/exim4/passwd.client***

Edit the file to add the credentials

```
# password file used when the local exim is authenticating to a remote
# host as a client.
#
# see exim4_passwd_client(5) for more documentation
#
# Example:
### target.mail.server.example:login:password
smtp.domain.com:smtpuser:smtppassword
```