SECURE MAILGATEWAY
OTC

CONTENT

1. INTRODUCTION SECURE MAIL GATEWAY

1.1. DESCRIPTION MAIL-GATEWAY / PRINCIPLE

If you regularly send large amounts of e-mail from your e-mail server, anti-spam databases can quickly put them on the list of unwanted senders - in other words, they will be set to blacklisted ("blacklist"). In that case, the IP address from which the allegedly unwanted emails are sent is classified as untrust and emails sent from there are no longer delivered.

To avoid such unwanted consequences, the Open Telekom Cloud works with a Secure Mail Gateway service. This service ensures the official and secure delivery of outgoing e-mails and prevents their inclusion in that called blacklists, the various blacklist providers. The original IP address of the sender is not directly recognizable because the Secure Mail Gateway always carries out the communication.

For security reasons, only this type of e-mail communication is activated in the Open Telekom Cloud.

The Secure Mail Gateway is the interface for sending mail from the Open Telekom Cloud to the Internet. The Secure Mail Gateway environment consists of three mail servers, which are distributed over three locations for redundancy. The locations are: eu-de-01 / eu-de-02 / eu-de-03

1.2. SCOPE OF SERVICE

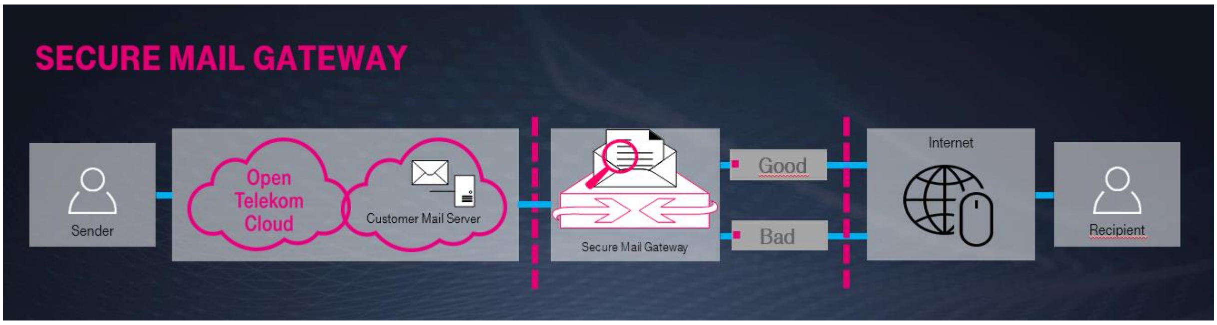
Included Services:

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

Info:
In order to ensure the reliability and performance of the Secure Mailgateways, the mailing per customer can be limited to 100 Mails/min.
In case the limit is activated and exceeded, the sender will receive the error message: „451 Ratelimit reached“.
The limit is not active by default.

Note: Due to the availability of the target systems, no guarantee can be given for delivery in time. ISPs may discard or lose email messages; recipients may have mistakenly given an incorrect address or ISPs may reject or tacitly reject the message if the recipient does not want to receive it.

1.3. SCHEMATIC



2. SECURITY MECHANISMS

2.1. GOOD / BAD GATEWAY / ANTI SPAM

The Good / Bad Routing which redirects Spam Mails internally to the Bad Gateway helps to ensure that the Good Gateways do not end up on external blacklists. The Good / Bad Routing thus helps that the mails of the customers do not influence each other.

"Good": If the mail is unremarkable, it will be sent via one of the Good Gateways.
"Bad": In the case of suspected spam, the mail is sent via one of the Bad Gateways (higher probability of a blacklist.) If the Bad Gateway is on a blacklist, it will be checked if this can be repaired (but there are none) Guarantee)

On the Secure Mail Gateway all mails are checked for spam. If Mails classified as spam are redirected to the BAD Gateway within the Secure Mail Gateway environment and sent to the Internet from there.

Note: The Secure Mail Gateway: "otc-de-mta03.mms.t-systems-service.com" is the Bad Gateway. Only mails are sent via this Secure Mail Gateway if mails from the Open Telekom Cloud have already been classified as spam. It is therefore possible that this system (Bad Gateway) will be listed under certain circumstances on "blacklists". Regular mails are not sent to the Internet via this bad gateway. If an NDR (None Delivery Report) message is sent from the Bad Gateway to the customer, it should be checked on the customer's side why this mail was classified as spam.

2.2. VIRUS CHECK

On the Secure Mail Gateways all mails are checked for viruses. Mails include a virus are rejected with a corresponding info and sent an NDR (Non delivery Report) to the sender of the mail.

Note: Virus scanning only takes place "outbound". The sender will be informed via info mail.

2.3. BLACKLIST MONITORING

An automated monitoring the Blacklists by the Blacklist Provider has been set up: the script runs daily at 5am. If a blocked list is detected, FMB Change-OpenTelekomCloud (DD-OTC-SDM@telekom.de) will be informed; Removal requests from blacklist by the Blacklist Providers are created immediately. Depending on the processing time of the blacklist provider, this lock will be removed accordingly.

2.4. SENDER DOMAIN CHECK

The mail domain used to send mail from the Open Telekom Cloud must be a valid public domain. (This means that the domain must have a public MX record or at least a public A record.) Mails with sender addresses where these criteria are not met are discarded.

Note: Error message if rejected: 550 Sender Domain must resolve

2.5. FORCED TLS

Based on sender and recipient domain, a forced TLS communication can be set up. For Mails where those setup is hitting the connection between the secure mail Gateway and the external mailserver is mandatory encrypted via TLS. (In case or error the incoming mail will be directly rejected or hold back, if a TLS secured connection to the external mailserver can't be established.)

2.6. DKIM SIGNING

Based on sender domain a DKIM signing can be configured for outgoing mails.
(DKIM signing can be requested over Cloud Handling Support contact.)

3. VARIOUS SET UP OPTIONS

In order to be able to use the Secure Mail Gateway, the following setup steps are necessary on the customer side (mail server or application of the customer).

3.1. SMARTHOST

The following DNS A Record must be stored as SMARTHOST in the mail server or application:

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

For the connection of systems from the Open Telekom Cloud to the SMARTHOST the **Port 25** is available.
(Other ports are not active on the secure mail gateway for mail acceptance.)

3.2. LOGIN-DATA

Authentication with username is now necessary to reduce the suspicioin of spam and increase security through TLS authentication. Username will be provided by Cloud Handling Support
(contact: service@open-telekom-cloud.com).

Customer costs: ~23€/month. Will be invoiced via OTC..

3.3. T-SYSTEMS EMAIL PROTECT PRO (IN CASE OF INCOMING MAIL REQUIREMENT)

This is a direct and full managed service from T-Systems (not an OTC service!) available at:
https://cloud.telekom.de/de/magenta-security/e-mail-protect-pro

Contact: Available contacts: FMB Secure-E-Mail secure-email@t-systems.com

Customer costs: to be requested at Secure Mail team. Will not be invoiced by OTC.

4. ADDITIONAL INFORMATION

4.1. SPF - SENDER POLICY FRAMEWORK

If the customer has set up an SPF record for his email domains, the customer must include the following DNS A record to his SPF record:

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

Info: The Sender Policy Framework (in short SPF) is a technique to make it more difficult to falsify the sender of an e-mail.

IP addresses contained in the A Record:

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

Table 1: SPF Record IPs

4.2. DNS - CONFIGURATION SENDER MAIL DOMAINS

To ensure a proper mailhandling over the Secure Mail Gateway it is necessary to create an MX Record or minimum an DNS A Record for the from customer used sender mail domain. Ideally the MX Record should point to a mailserver which is capable of receiving replies and NDRs (Non Delivery Reports) for the used mail domain.

Comment: / Tip:
In case there is no mailserver available for the used sender mail domain, then the reply-to header could be added to the outgoing mails. With this header it is possible to reroute the reply and NDR (Non Delivery Report) messages to an existing Mailbox (for example to an functional mailbox below an existing customer mail domain).

5. INSTALLATION EXAMPLES / BASIC INSTALLATION

5.1. EXAMPLES POSTFIX UBUNTU

(Source: 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. EXAMPLE EXIM UBUNTU

(Source: 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

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

Authentication on the smarthost

However, if the SMTP server on the smart host requires a password to send the mail, the newly created configuration **/etc/postfix/main.cf** must be edited again and these lines inserted:

```
smtp_sasl_auth_enable = yes

# noplaintext leave, if passwords have to be transferred in plain text:

# (not recommended, only if it does not work differently)

smtp_sasl_security_options = noplaintext noanonymous

smtp_sasl_password_maps = hash:/etc/postfix/sasl_password
```

As shown in the configuration file, Postfix retrieves the access data from the file / etc / postfix / sasl_password or from a database that is generated from the sasl_password. The file should preferably be created with the following command, otherwise conversion into a database is not always possible. You have to open a terminal window[2] and enter the following command:

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

Now you write your data according to the following pattern in the file

```
smtp.mailanbieter.de username:securepassword
```

So that not everyone can read the password, you should limit the permissions of the file (possibly this ist o be repeated for backup copies or the database created below):

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

Now the database has to be created:

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

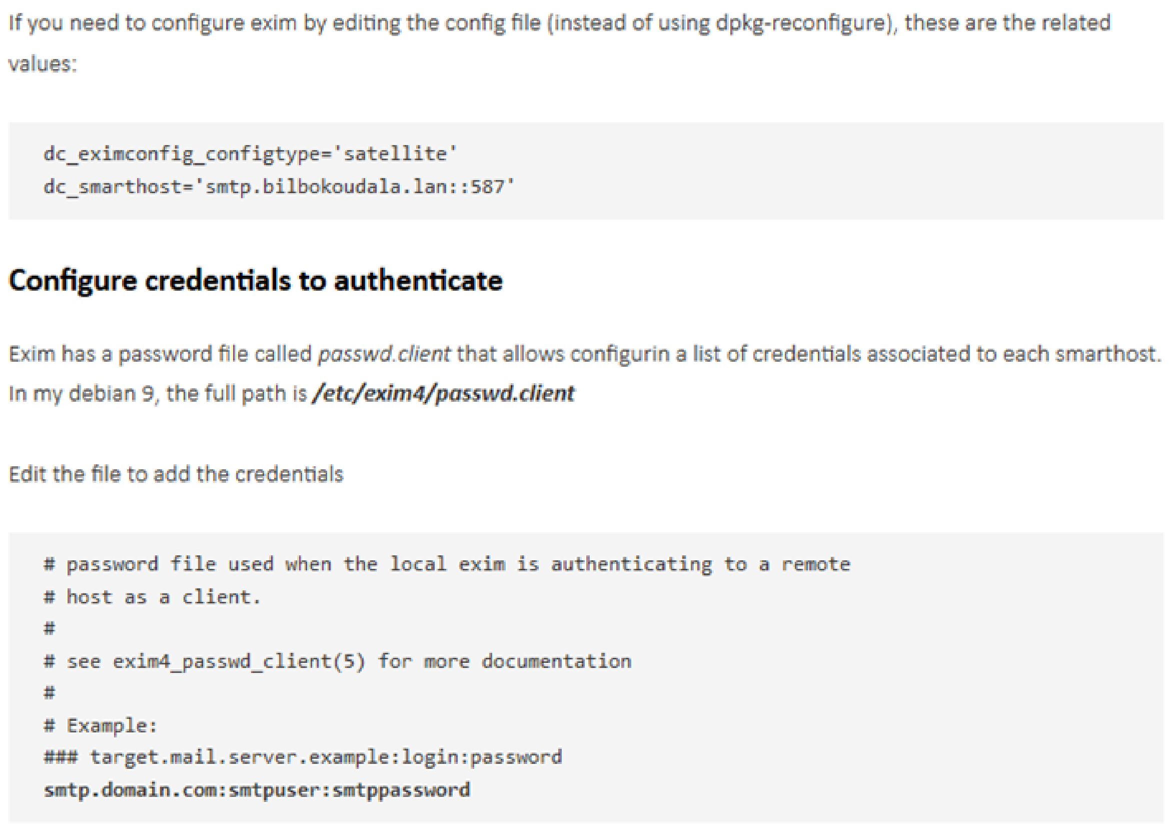
Then you have to restart postfix:

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

6.2. EXIM

Help → Source: https://wiki.debian.org/Exim

Example-Source: 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
```