CIPHERMAIL EMAIL ENCRYPTION

CipherMail Email Encryption Gateway Installation Guide

May 8, 2018, Rev: 12370

Contents

1 Introduction

This install guide explains how to install CipherMail on Ubuntu, Debian, RedHat, CentOS and SUSE. The .deb and .rpm packages have been tested on Ubuntu 16.04, Debian 9, RedHat/CentOS 7 and SLES 12. For installation on systems not supported by the .deb or .rpm packages, you are advised to use the manual installation guide. It is recommended to install CipherMail on a dedicated and clean machine.

Requirements

- PostgreSQL, MySQL or Oracle
- Postfix
- OpenJDK 7 or 8
- ANT
- Tomcat (or Jetty)

Note: Commands that should be executed by the user are shown on lines starting with a *$* sign (the *$* sign is not part of the command to execute). It is recommended to copy and paste the commands directly to the command line. Some PDF readers do not support copy-and-paste from PDF. To make sure that copy-and-paste work correctly, it is advised to copy-and-paste the commands directly from the separately downloadable file **installation-guide.txt** and not from this PDF.

> Warning
>
> do not install CipherMail on a live email system!

2 Install CipherMail on Ubuntu & Debian

This section explains how to install CipherMail on Ubuntu and Debian.

> Note
>
> This guide assumes that CipherMail will be configured for PostgreSQL. If MySQL/MariaDB or Oracle Database should be used instead, all PostgreSQL related steps can be skipped [a]. See Appendix A on how to configure CipherMail for MySQL/MariaDB and Appendix B on how to configure CipherMail for Oracle Database.
>
> [a] Alternatively, CipherMail can first be installed with PostgreSQL. After confirming that CipherMail works correctly with PostgreSQL, support for the other database can be configured

Install required packages[1]

```
$ sudo apt-get install postgresql postfix openjdk-8-jre \
openjdk-8-jre-headless ant ant-optional \
mktemp libsasl2-modules symlinks
```

Note: During the installation of Postfix, select “No Configuration”.

2.1 Install CipherMail packages

A full installation of CipherMail requires the CipherMail encryption back-end and the Web GUI front-end. The .deb packages can be downloaded from http://www.ciphermail.com. The following three files are required:

- djigzo_?.?.?-?_all.deb
- djigzo-postgres_?.?.?-?_all.deb
- djigzo-web_?.?.?-?_all.deb

Install back-end packages

```
$ sudo dpkg -i djigzo_?.?.?-?_all.deb
$ sudo dpkg -i djigzo-postgres_?.?.?-?_all.deb
```

Restart back-end

```
$ sudo service djigzo restart
```

Install Web-GUI package

```
$ sudo dpkg -i djigzo-web_?.?.?-?_all.deb
```

2.2 Configure Postfix

A Postfix after queue filter is used for encrypting and decrypting incoming and outgoing email. This requires some changes to the postfix configuration files. CipherMail installs a pre-configured Postfix main and master configuration file which should be copied to the postfix configuration directory.

Warning

The following commands will overwrite all settings in the original postfix config files. If existing postfix settings must be kept, the required changes should be manually applied.

[1] The sudo package is required by CipherMail. Debian does not install sudo by default. If installing on Debian, sudo must be installed prior to installing CipherMail.

Copy postfix configuration files

```
$ sudo cp /etc/postfix/djigzo-main.cf /etc/postfix/main.cf
$ sudo cp /etc/postfix/djigzo-master.cf /etc/postfix/master.cf
```

Update aliases Postfix uses /etc/alias as the alias file. Make sure that the alias file is available and up-to-date.

```
$ sudo newaliases
```

Restart postfix

```
$ sudo service postfix restart
```

2.3 Install Tomcat

Install the required Tomcat package

```
$ sudo apt-get install tomcat8
```

Note: On older releases, install tomcat7 and change the commands below to match tomcat7

Set djigzo-web.home The system property **djigzo-web.home** should reference the location where CipherMail Web GUI is stored. The property will be added to the Tomcat default config file.

```
$ sudo bash -c 'echo "JAVA_OPTS=\"\$JAVA_OPTS -Ddjigzo-web.home=\
/usr/share/djigzo-web\"" >> /etc/default/tomcat8'
```

Configure Tomcat memory usage In order to allow the import of very large certificate files (.p7b or .pfx files with thousands of certificates) CipherMail requires that Tomcat is setup with at least 128 MB heap size.

```
$ sudo bash -c 'echo "JAVA_OPTS=\"\$JAVA_OPTS \
-Djava.awt.headless=true -Xmx128M\"" >> /etc/default/tomcat8'
```

Allow reading and writing of SSL certificate CipherMail Web GUI allows new SSL certificates for the Web GUI to be uploaded using the SSL import page. To support this functionality, Tomcat should be allowed to read and write the SSL certificate.

```
$ sudo chown tomcat8:djigzo /usr/share/djigzo-web/ssl/sslCertificate.p12
```

Adding an HTTPS connector An HTTPS connector should be added to the Tomcat server configuration. If Tomcat is only used by CipherMail, it's advised to replace the existing Tomcat configuration file (/etc/tomcat8/server.xml) with the configuration file provided by CipherMail.

Warning

This overwrites the existing server.xml file. If you want to keep the existing server.xml file, you need to manually add the HTTPS Connector. See Appendix C for more information.

```
$ sudo cp /usr/share/djigzo-web/conf/tomcat/server.xml /etc/tomcat8/
```

Note: If using Tomcat8, because of a bug in Tomcat 8 (https://bz.apache.org/bugzilla/show_bug.cgi?id=60940), the setting "unpackWARs" in /etc/tomcat/server.xml should be changed from "false" to "true"

```
$ sudo sed -i 's/unpackWARs="false"/unpackWARs="true"/' /etc/tomcat8/server.xml
```

Adding the Web admin context A context should be added to Tomcat to enable the Web admin application.

```
$ sudo bash -c 'echo "<Context docBase=\"/usr/share/djigzo-web/djigzo.war\
\" />" > /etc/tomcat8/Catalina/localhost/ciphermail.xml'
```

Note: if you want CipherMail web admin to use the root context, save the context file to ROOT.xml (overwriting the existing file) instead of to ciphermail.xml[2].

Adding the Web portal context If the portal functionality is required, a specific portal context should be added to Tomcat.

```
$ sudo bash -c 'echo "<Context docBase=\"/usr/share/djigzo-web/djigzo-portal.war\
\" />" > /etc/tomcat8/Catalina/localhost/web.xml'
```

Restart Tomcat Tomcat should be restarted to make it use the new Tomcat configuration.

```
$ sudo service tomcat8 restart
```

[2] the root context allows you to access CipherMail using a URL of the form https://192.168.178.2:8443 instead of https://192.168.178.2:8443/ciphermail

2.4 Finish

Open the Web GUI CipherMail should now be running (wait some time for Tomcat to startup). The login page can be accessed using the following URL `https://192.168.178.2:8443/ciphermail`[3] (change the IP address accordingly)

> Note
>
> CipherMail comes with a pre-installed SSL certificate which is not by default trusted by your browser. You should therefore manually accept the SSL certificate the first time you open the page. A new trusted SSL certificate can be uploaded from the web GUI.

Login Use the following login credentials:

username:	admin
password:	admin

> Note
>
> The login procedure can take some time after a restart because the Web GUI requires some internal initialization after a restart.

Log output If CipherMail is not running, check the following log files for errors:

CipherMail log

```
$ less /var/log/djigzo.log
```

Tomcat log

```
$ less /var/log/tomcat8/catalina.out
```

[3] if CipherMail was installed as the root context, the URL should be https://192.168.178.2:8443

3 Install CipherMail on Red Hat 7 & CentOS 7

This section explains how to install CipherMail on Red Hat 7 and CentOS 7. It is assumed that all commands are run as root (i.e., the user is logged in as root).

3.1 SELinux

By default SELinux is enabled on RedHat/CentOS. SELinux prevents certain operations to be executed which are required by CipherMail. For example, a local listening port on port 10026 must be opened by Postfix (this port is used by CipherMail as the Postfix "reinjection" port). SELinux however, by default, does not allow this. Disabling SELinux is recommended if you are not familiar with SELinux.

SELinux can be disabled by editing the file `/etc/sysconfig/selinux`. Set `SELINUX` to `disabled` and reboot the server.

3.2 Configure firewall

Red Hat and CentOS by default block access to most ports. The firewall should therefore be configured to allow access to certain ports used by CipherMail. The following ports should be remotely accessible: SMTP (25) and 8443[4].

```
$ firewall-cmd --zone=public --add-port=25/tcp --permanent
$ firewall-cmd --zone=public --add-port=8443/tcp --permanent
$ firewall-cmd --reload
```

If the web GUI should be accessible on the standard https port (443) instead of 8443, add the following additional firewall rules

```
$ firewall-cmd --zone=public --add-forward-port=port=443:proto=tcp:toport=8443 \
--permanent
$ firewall-cmd --reload
```

Note

This guide assumes that CipherMail will be configured for PostgreSQL. If MySQL/MariaDB or Oracle Database should be used instead, all PostgreSQL related steps can be skipped [a]. See Appendix A on how to configure CipherMail for MySQL/MariaDB and Appendix B on how to configure CipherMail for Oracle Database.

[a] Alternatively, CipherMail can first be installed with PostgreSQL. After confirming that CipherMail works correctly with PostgreSQL, support for the other database can be configured

[4] See Appendix E.1 for an overview of all ports used by CipherMail.

3.3 Install PostgreSQL

```
$ yum install postgresql-server
```

PostgreSQL should be initialized and restarted.

```
$ postgresql-setup initdb
$ systemctl restart postgresql.service
```

3.4 Install required packages

Certain packages need to be installed before installing CipherMail.

```
$ yum install redhat-lsb sudo postfix ant mktemp symlinks \
java-1.8.0-openjdk-headless java-1.8.0-openjdk java-1.8.0-openjdk-devel
```

3.5 RPM signing keys

The CipherMail RPM packages are signed with a GPG key. To validate the signature of the packages, the GPG key from https://www.ciphermail.com/downloads/ciphermail-signing-key.asc should be installed.

```
$ rpm --import https://www.ciphermail.com/downloads/ciphermail-signing-key.asc
```

The signature of the rpm packages can be validated with the following command

```
$ rpm -K <file>
```

3.6 Install CipherMail packages

A full installation of CipherMail requires the CipherMail encryption back-end and the Web GUI front-end. The RPM packages can be downloaded from http://www.ciphermail.com. The following three files are required:

- djigzo-?.?.?-?.noarch.rpm
- djigzo-postgres-?.?.?-?.noarch.rpm
- djigzo-web-?.?.?-?.noarch.rpm

Install back-end packages

```
$ yum install djigzo-?.?.?-?.noarch.rpm
$ yum install djigzo-postgres-?.?.?-?.noarch.rpm
```

Install Web-GUI package

```
$ yum install djigzo-web-?.?.?-?.noarch.rpm
```

3.7 Configure Postfix

A Postfix after queue filter is used for encrypting and decrypting incoming and outgoing email.

Copy Postfix config A Postfix after queue filter is used for encrypting and decrypting incoming and outgoing email. This requires some changes to the postfix configuration files. CipherMail installs a pre-configured Postfix main and master configuration file which should be copied to the postfix configuration directory.

Warning

The following commands will overwrite all settings in the original postfix config files. If existing postfix settings must be kept, the required changes should be manually applied.

```
$ cp /etc/postfix/djigzo-master.cf /etc/postfix/master.cf
$ cp /etc/postfix/djigzo-main.cf /etc/postfix/main.cf
```

Update aliases Postfix uses /etc/alias as the alias file. Make sure that the alias file is available and up-to-date.

```
$ newaliases
```

Restart postfix

```
$ systemctl restart postfix.service
```

Make mail logs readable The mail logs should be readable by user *djigzo*. We will therefore add a special maillog group.

Note: this can be skipped if you do not want the MTA log to be shown on the MTA page

```
$ groupadd maillog
$ usermod -a -G maillog djigzo

$ chown root:maillog /var/log/maillog
$ chmod g+r /var/log/maillog
```

Configure logrotate By default mail logs are rotated with the date appended to the filename (see *dateext* setting). CipherMail however expects the rotated log files to be appended with an increasing number. To allow multiple mail log files to be read, modify the logrotate setting for maillog. Create a separate rotate rule for maillog by removing the default rule and appending the following lines to `/etc/logrotate.d/syslog`[5].

[5] This can be skipped if it is sufficient to only show the most recent log file (/var/log/maillog) directly from the mail log GUI page

```
$ vi /etc/logrotate.d/syslog
```

```
/var/log/maillog
{
  nodateext
  compress
  create 640 root maillog
  delaycompress
  sharedscripts
  postrotate
    /bin/kill -HUP `cat /var/run/syslogd.pid 2> /dev/null` 2> /dev/null || true
  endscript
}
```

Note: don't forget to remove the original `/var/log/maillog` line.

3.8 Install Tomcat

```
$ yum install tomcat
```

Update Javamail Red Hat/CentOS by default installs an older version of Javamail. The newer version of Javamail provided by CipherMail will be added as a new alternative.

```
$ alternatives --install /usr/share/java/javamail.jar javamail \
/usr/share/djigzo/lib/mail.jar 20000
```

Set djigzo-web.home and Tomcat memory usage The system property **djigzo-web.home** should reference the location where CipherMail Web GUI is stored. In order to allow the import of very large certificate files (.p7b or .pfx files with thousands of certificates) CipherMail requires that Tomcat is setup with at least 128 MB heap size. These settings will be added to the Tomcat default config file:

```
$ echo "JAVA_OPTS=\"-Ddjigzo-web.home=/usr/share/djigzo-web \
-Djava.awt.headless=true -Xmx128M\"" >> /etc/sysconfig/tomcat
```

Adding an HTTPS connector An HTTPS connector should be added to the Tomcat server configuration. If Tomcat is only used by CipherMail, it's advised to replace the existing Tomcat configuration file (/etc/tomcat/server.xml) with the configuration file provided by CipherMail.

Warning

The following command overwrites the existing server.xml file. If you want to keep the existing server.xml file, you need to manually add the HTTPS Connector. See Appendix C for more information.

```
$ cp /usr/share/djigzo-web/conf/tomcat/server.xml /etc/tomcat/
```

Adding the Web admin context A context should be added to Tomcat to enable the Web admin application.

```
$ echo "<Context docBase=\"/usr/share/djigzo-web/djigzo.war\
\" />" > /etc/tomcat/Catalina/localhost/ciphermail.xml
```

Note: if you want CipherMail web admin to use the root context, save the context file to ROOT.xml (overwriting the existing file) instead of to ciphermail.xml[6].

Adding the Web portal context If the portal functionality is required, a specific portal context should be added to Tomcat.

```
$ echo "<Context docBase=\"/usr/share/djigzo-web/djigzo-portal.war\
\" />" > /etc/tomcat/Catalina/localhost/web.xml
```

Allow reading and writing of SSL certificate CipherMail Web GUI allows new SSL certificates for the Web GUI to be uploaded using the SSL import page. To support this functionality, Tomcat should be allowed to read and write the SSL certificate.

```
$ chown tomcat:djigzo /usr/share/djigzo-web/ssl/sslCertificate.p12
```

Make Tomcat start at reboot Tomcat should be automatically started at reboot.

```
$ systemctl enable tomcat.service
```

3.9 Finalize

Start services

```
$ systemctl restart djigzo.service
$ systemctl restart tomcat.service
```

Open the Web GUI CipherMail should now be running (wait some time for Tomcat to startup). The login page can be accessed using the following URL `https://192.168.178.2:8443/ciphermail`[7] (change the IP address accordingly)

> **Note**
>
> CipherMail comes with a pre-installed SSL certificate which is not by default trusted by your browser. You should therefore manually accept the SSL certificate the first time you open the page. A new trusted SSL certificate can be uploaded from the web GUI.

[6] the root context allows you to access CipherMail using a URL of the form https://192.168.178.2:8443 instead of https://192.168.178.2:8443/ciphermail

[7] if CipherMail was installed as the root context, the URL should be https://192.168.178.2:8443

Login Use the following login credentials:

username:	admin
password:	admin

Note

The login procedure can take some time after a restart because the Web GUI requires some internal initialization after a restart.

Log output If CipherMail is not running, check the following logs for errors:

CipherMail log

```
$ less /var/log/djigzo.log
```

Tomcat log

```
$ journalctl -u tomcat.service
```

4 Install CipherMail on SUSE

This section explains how to install CipherMail on SUSE.

4.1 Configure firewall

If a local firewall is enabled, it should be configured to allow access to certain ports. The following ports should be remotely accessible: SMTP (25) and 8443[8]. The firewall can for example be configured with yast.

```
$ yast
```

4.2 Configure logging

Because CipherMail reads the logs from /var/log it's advised to install rsyslog.

Note: this can be skipped. However the MTA log view will no longer show the MTA logs

```
$ zypper install rsyslog
```

Note: If a warning "Problem: systemd-logger conflicts with namespace:otherproviders(syslog)..." is shown, select Solution 1: "deinstallation of systemd-logger-..."

After installing rsyslog, a reboot is required

```
$ reboot
```

4.3 Install required packages

Certain packages need to be installed before installing CipherMail.

> Note
>
> This guide assumes that CipherMail will be configured for PostgreSQL. If MySQL/MariaDB or Oracle Database should be used instead, all PostgreSQL related steps can be skipped [a]. See Appendix A on how to configure CipherMail for MySQL/MariaDB and Appendix B on how to configure CipherMail for Oracle Database.
>
> [a] Alternatively, CipherMail can first be installed with PostgreSQL. After confirming that CipherMail works correctly with PostgreSQL, support for the other database can be configured

```
$ zypper install sudo postfix ant postgresql-server \
java-1_8_0-openjdk-headless java-1_8_0-openjdk-devel java-1_8_0-openjdk-devel
```

[8] See Appendix E.1 for an overview of all ports used by CipherMail.

Note: if CipherMail cannot be installed because of a conflict with postfix, select “Solution 1: deinstallation of patterns-openSUSE-minimal_base-conflicts” or manually remove the package “patterns-openSUSE-minimal_base-conflicts” before installing CipherMail.

4.4 RPM signing keys

The CipherMail RPM packages are signed with a GPG key. To validate the signature of the packages, the GPG key from https://www.ciphermail.com/downloads/ciphermail-signing-key.asc should be installed.

```
$ rpm --import https://www.ciphermail.com/downloads/ciphermail-signing-key.asc
```

4.5 Install CipherMail packages

A full installation of CipherMail requires the CipherMail encryption back-end and the Web GUI front-end. The RPM packages can be downloaded from http://www.ciphermail.com. The following three files are required:

- djigzo-?.?.?-?.SUSE.noarch.rpm
- djigzo-postgres-?.?.?-?.SUSE.noarch.rpm
- djigzo-web-?.?.?-?.noarch.rpm

Install back-end packages

```
$ zypper install djigzo-?.?.?-?.SUSE.noarch.rpm
$ zypper install djigzo-postgres-?.?.?-?.SUSE.noarch.rpm
```

Install Web-GUI package

```
$ zypper install djigzo-web-?.?.?-?.noarch.rpm
```

4.6 Configure Postfix

A Postfix after queue filter is used for encrypting and decrypting incoming and outgoing email.

Copy Postfix config A Postfix after queue filter is used for encrypting and decrypting incoming and outgoing email. This requires some changes to the postfix configuration files. CipherMail installs a pre-configured Postfix main and master configuration file which should be copied to the postfix configuration directory.

Warning

The following commands will overwrite all settings in the original postfix config files. If existing postfix settings must be kept, the required changes should be manually applied.

```
$ cp /etc/postfix/djigzo-master.cf /etc/postfix/master.cf
$ cp /etc/postfix/djigzo-main.cf /etc/postfix/main.cf
```

Update aliases Postfix uses /etc/alias as the alias file. Make sure that the alias file is available and up-to-date.

```
$ newaliases
```

Restart postfix

```
$ service postfix restart
```

Make mail logs readable The mail logs should be readable by user *djigzo*. We will therefore add a special maillog group.

Note: this can be skipped if you do not want the MTA log to be shown on the MTA page

```
$ groupadd maillog
$ usermod -a -G maillog djigzo

$ chown root:maillog /var/log/mail.info
$ chmod g+r /var/log/mail.info
```

Configure logrotate By default mail logs are rotated with the date appended to the filename (see *dateext* setting). CipherMail however expects the rotated log files to be appended with an increasing number. To allow multiple mail log files to be read, modify the logrotate setting for maillog. Create a separate rotate rule for maillog by removing the default rule and appending the following lines to `/etc/logrotate.d/syslog`[9].

```
$ vi /etc/logrotate.d/syslog
```

```
/var/log/mail.info
{
    compress
    delaycompress
    nodateext
    maxage 365
    rotate 99
    missingok
    notifempty
    size +4096k
    create 640 root maillog
    sharedscripts
    postrotate
        /usr/bin/systemctl reload syslog.service > /dev/null
    endscript
}
```

[9] This can be skipped if it is sufficient to only show the most recent log file (/var/log/mail.info) directly from the mail log GUI page

Note: don't forget to remove the original `/var/log/mail.info` part.

4.7 Install Tomcat

```
$ zypper install tomcat
```

Set djigzo-web.home and Tomcat memory usage The system property **djigzo-web.home** should reference the location where CipherMail Web GUI is stored. In order to allow the import of very large certificate files (.p7b or .pfx files with thousands of certificates) CipherMail requires that Tomcat is setup with at least 128 MB heap size. These settings will be added to the Tomcat default config file:

```
bash -c 'echo "JAVA_OPTS=\"-Ddjigzo-web.home=\
/usr/share/djigzo-web -Djava.awt.headless=true \
-Xmx128M\"" >> /etc/tomcat/tomcat.conf'
```

Allow reading and writing of SSL certificate CipherMail Web GUI allows new SSL certificates for the Web GUI to be uploaded using the SSL import page. To support this functionality, Tomcat should be allowed to read and write the SSL certificate.

```
$ chown tomcat:djigzo /usr/share/djigzo-web/ssl/sslCertificate.p12
```

Adding an HTTPS connector An HTTPS connector should be added to the Tomcat server configuration. If Tomcat is only used by CipherMail, it's advised to replace the existing Tomcat configuration file (/etc/tomcat/server.xml) with the configuration file provided by CipherMail.

> **Warning**
>
> The following command overwrites the existing server.xml file. If you want to keep the existing server.xml file, you need to manually add the HTTPS Connector. See Appendix C for more information.

```
$ cp /usr/share/djigzo-web/conf/tomcat/server.xml /etc/tomcat/
```

Note: Because of a bug in Tomcat 8 (https://bz.apache.org/bugzilla/show_bug.cgi?id=60940), the setting "unpackWARs" in /etc/tomcat/server.xml should be changed from "false" to "true"

```
$ sudo sed -i 's/unpackWARs="false"/unpackWARs="true"/' /etc/tomcat/server.xml
```

Adding the Web admin context A context should be added to Tomcat to enable the Web admin application.

```
$ bash -c 'echo "<Context docBase=\"/usr/share/djigzo-web/djigzo.war\
\" />" > /etc/tomcat/Catalina/localhost/ciphermail.xml'
```

Note: if you want CipherMail web admin to use the root context, save the context file to ROOT.xml (overwriting the existing file) instead of to ciphermail.xml[10].

Adding the Web portal context If the portal functionality is required, a specific portal context should be added to Tomcat.

```
$ bash -c 'echo "<Context docBase=\"/usr/share/djigzo-web/djigzo-portal.war\
\" />" > /etc/tomcat/Catalina/localhost/web.xml'
```

Make Tomcat start at reboot Tomcat should be automatically started at reboot.

```
$ chkconfig tomcat on
```

4.8 Finalize

Start services

```
$ service djigzo restart
$ service tomcat restart
```

Open the Web GUI CipherMail should now be running (wait some time for Tomcat to startup). The login page can be accessed using the following URL https://192.168.178.2:8443/ciphermail[11] (change the IP address accordingly)

Note

CipherMail comes with a pre-installed SSL certificate which is not by default trusted by your browser. You should therefore manually accept the SSL certificate the first time you open the page. A new trusted SSL certificate can be uploaded from the web GUI.

Login Use the following login credentials:

username:	admin
password:	admin

Note

The login procedure can take some time after a restart because the Web GUI requires some internal initialization after a restart.

[10] the root context allows you to access CipherMail using a URL of the form https://192.168.178.2:8443 instead of https://192.168.178.2:8443/ciphermail

[11] if CipherMail was installed as the root context, the URL should be https://192.168.178.2:8443

Log output If CipherMail is not running, check the following logs for errors:

CipherMail log

```
$ less /var/log/djigzo.log
```

Tomcat log

```
$ journalctl -u tomcat.service
```

A MySQL/MariaDB

This section explains on how to configure CipherMail with support for MySQL/MariaDB.

> **Note**
>
> It is assumed that MySQL or MariaDB is already installed (either on the same system as CipherMail or an external system) and that MySQL/MariaDB is accessible from the CipherMail system.

A.1 Configure MySQL/MariaDB

max_allowed_packet CRLs and quarantined emails are often larger than the default configured max_allowed_packet size. The max_allowed_packet size therefore need to be reconfigured to support large binary data.

Ubuntu/Debian

The max_allowed_packet can be set in a local configuration file (/etc/mysql/conf.d/ciphermail.cnf) or directly in the main configuration file /etc/mysql/my.cnf.

```
$ sudo vi /etc/mysql/conf.d/ciphermail.cnf
```

Copy the following lines to `ciphermail.cnf`:

```
[mysqld]
max_allowed_packet = 128M
```

Restart MariaDB:

```
$ sudo service mysql restart
```

RedHat/CentOS

The max_allowed_packet can be set in a local configuration file (/etc/my.cnf.d/ciphermail.cnf) or directly in the main configuration file /etc/my.cnf.

```
$ vi /etc/my.cnf.d/ciphermail.cnf
```

Copy the following lines to `ciphermail.cnf` and restart MySQL/MariaDB.

```
[mysqld]
max_allowed_packet = 128M
```

Restart MariaDB:

```
$ systemctl restart mariadb
```

Note: the max_allowed_packet size should be larger than the largest email or CRL size that should be supported.

A.2 Configure database

CipherMail requires requires a database ’djigzo’ owned by user ’djigzo’.

Login as the database administrator:

```
$ mysql
```

Note: If a special database admin user account is configured, use the database admin account instead of root.

The following SQL commands will add the database user and create the database:

```
CREATE USER 'djigzo'@'localhost' IDENTIFIED BY 'djigzo';
CREATE DATABASE djigzo CHARACTER SET utf8 COLLATE utf8_general_ci;
GRANT DELETE,INSERT,SELECT,UPDATE,LOCK TABLES,DROP,CREATE,ALTER ON djigzo.*
TO 'djigzo'@'localhost';
```

Note: Replace ’localhost’ with the IP address of the CipherMail server if MySQL/MariaDB runs on an external system.

A.2.1 Import table definitions

```
$ mysql djigzo < /usr/share/djigzo/conf/database/sql/djigzo.mysql.sql
```

A.3 Configure CipherMail

CipherMail requires a number of changes.

A.3.1 Configure database type

CipherMail should be configured to use MySQL/MariaDB instead of PostgreSQL.

In the file `wrapper-additional-parameters.conf` set `ciphermail.hibernate.database.type` to `mysql`

```
$ sudo vi /usr/share/djigzo/wrapper/wrapper-additional-parameters.conf
```

Add the following line to the end of `wrapper-additional-parameters.conf`:

```
-Dciphermail.hibernate.database.type=mysql
```

A.3.2 Configure database connection

The database connection, hostname of database server etc., should be configured in the file `hibernate.mysql.connection.xml`.

```
$ sudo vi /usr/share/djigzo/conf/database/hibernate.mysql.connection.xml
```

By default the database connection is configured to connect to MySQL/MariaDB on localhost. Change this to the IP address (or fully qualified domain name) of the MySQL/MariaDB server if the database server runs on a different host.

A.3.3 Configure backup/restore

For backup/restore, a mysql password file with the database password, should be placed in the database directory.

```
$ sudo vi /usr/share/djigzo/conf/database/mysql.cnf
```

Copy the following content to mysql.cnf:

```
[client]
user=djigzo
password=djigzo

[mysqldump]
user=djigzo
password=djigzo
```

Note: change the user and password to match the database user.
set owner and file permissions:

```
$ sudo chown djigzo:djigzo /usr/share/djigzo/conf/database/mysql.cnf
$ sudo chmod 600 /usr/share/djigzo/conf/database/mysql.cnf
```

A.3.4 Restart services

Ubuntu/Debian

```
$ sudo service djigzo restart
$ sudo service tomcat8 restart
```

RedHat/CentOS

```
$ service djigzo restart
$ service tomcat restart
```

Check log file to see whether the back-end starts without any errors:

```
$ less /var/log/djigzo.log
```

B Oracle Database

This section explains on how to configure CipherMail with support for Oracle Database.

Note

It is assumed that Oracle Database is already installed (either on the same system as CipherMail or an external system) and that the Oracle Database is accessible from the CipherMail system.

B.1 Configure database

CipherMail requires requires a database ’djigzo’ owned by user ’djigzo’.

Login as the database administrator:

```
$ sqlplus system
```

Note: Change accordingly to match your database setup.

The following SQL commands will add the database user and grant the required permissions:

```
CREATE USER djigzo IDENTIFIED BY djigzo default tablespace USERS;

GRANT CREATE SESSION TO djigzo;
GRANT CREATE TABLE TO djigzo;
GRANT CREATE VIEW TO djigzo;
GRANT CREATE PROCEDURE TO djigzo;
GRANT CREATE SEQUENCE TO djigzo;
```

Set the quota for the user:

```
ALTER USER djigzo QUOTA 300M ON USERS;
```

Note: 300M should be sufficient for most setups. If the DLP quarantine functionality will be used, the quota might need to be increased.

B.1.1 Import table definitions

```
$ sqlplus djigzo/djigzo@XE
```

Note: Change accordingly to match your database setup.

To import the table definitions use the following sqlplus command:

```
@ /usr/share/djigzo/conf/database/sql/djigzo.oracle.sql
```

B.2 Configure CipherMail

CipherMail requires a number of changes.

B.2.1 Configure database type

CipherMail should be configured to use Oracle Database instead of PostgreSQL.

In the file `wrapper-additional-parameters.conf` set `ciphermail.hibernate.database.type` to `oracle`

```
$ sudo vi /usr/share/djigzo/wrapper/wrapper-additional-parameters.conf
```

Add the following line to the end of `wrapper-additional-parameters.conf`:

```
-Dciphermail.hibernate.database.type=oracle
```

B.2.2 Configure database connection

The database connection, hostname of database server etc., should be configured in the file `hibernate.oracle.connection.xml`.

```
$ sudo vi /usr/share/djigzo/conf/database/hibernate.oracle.connection.xml
```

By default the database connection is configured to connect to Oracle Database on localhost. Change this to the IP address (or fully qualified domain name) of the Oracle Database server if the database server runs on a different host.

B.2.3 Disable backup page

The built-in backup/restore functionality only works with a locally configured PostgreSQL database. The web GUI backup option can be disabled by adding the following option to the tomcat options:

```
-Dciphermail.backup.enabled=false
```

The option should be added to the tomcat default settings file

Ubuntu/Debian

```
$ sudo bash -c 'echo "JAVA_OPTS=\"\$JAVA_OPTS \
-Dciphermail.backup.enabled=false\"" >> /etc/default/tomcat8'
```

RedHat/CentOS Add the option `-Dciphermail.backup.enabled=false` to the last `JAVA_OPTS` line in the tomcat config file:

```
$ vi /etc/sysconfig/tomcat
```

The last `JAVA_OPTS` line should look similar to:

```
JAVA_OPTS="-Ddjigzo-web.home=/usr/share/djigzo-web -Djava.awt.headless=true -Xmx128M
 -Dciphermail.backup.enabled=false"
```

Note: because of the length of the line, the above line is shown on two lines. In the tomcat config file it should only be one line.

B.2.4 Restart services

Ubuntu/Debian

```
$ sudo service djigzo restart
$ sudo service tomcat8 restart
```

RedHat/CentOS

```
$ service djigzo restart
$ service tomcat restart
```

Check log file to see whether the back-end starts without any errors:

```
$ less /var/log/djigzo.log
```

C Adding Tomcat HTTPS connector

CipherMail uses the following Tomcat server.xml configuration files.

C.1 Tomcat 6,7 & 8

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE config [
<!ENTITY proxyName "">
]>
<Server>
    <Service name="Catalina">
        <Connector
            port="8443"
            connectionTimeout="20000"
            maxThreads="150"
            scheme="https"
            secure="true"
            SSLEnabled="true"
            sslProtocol="TLS"
            keystoreFile="/usr/share/djigzo-web/ssl/sslCertificate.p12"
            keystorePass="djigzo"
            keystoreType="PKCS12"
            proxyName="&proxyName;"
            ciphers="
                TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,
                TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
                TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384,
                TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384,
                TLS_DHE_DSS_WITH_AES_256_CBC_SHA256,
                TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA,
                TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
                TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA,
                TLS_ECDH_RSA_WITH_AES_256_CBC_SHA,
                TLS_DHE_DSS_WITH_AES_256_CBC_SHA,
                TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,
                TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
                TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256,
                TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256,
                TLS_DHE_DSS_WITH_AES_128_CBC_SHA256,
                TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA,
                TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
                TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA,
                TLS_ECDH_RSA_WITH_AES_128_CBC_SHA,
                TLS_DHE_DSS_WITH_AES_128_CBC_SHA,
                TLS_ECDHE_ECDSA_WITH_RC4_128_SHA,
                TLS_ECDH_ECDSA_WITH_RC4_128_SHA,
                TLS_ECDH_RSA_WITH_RC4_128_SHA,
                TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
                TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,
                TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,
                TLS_RSA_WITH_AES_256_GCM_SHA384,
                TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384,
                TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384,
                TLS_DHE_DSS_WITH_AES_256_GCM_SHA384,
                TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
                TLS_RSA_WITH_AES_128_GCM_SHA256,
                TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,
                TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256,
                TLS_DHE_DSS_WITH_AES_128_GCM_SHA256,
                TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA,
                TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA,
                TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA,
                TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA,
                TLS_EMPTY_RENEGOTIATION_INFO_SCSVF"
        />

        <Engine name="Catalina" defaultHost="localhost">
            <Host name="localhost" appBase="webapps" unpackWARs="false"/>
        </Engine>
    </Service>
</Server>
```

Note: If an existing server.xml should be used, the Connector for port 8443 should be added to the existing server.xml.

D Memory usage

CipherMail requires a lot of memory when it needs to encrypt large messages. By default the back-end encryption process will allocate 60% of the available memory for the heap size. If there are other applications running on the same server, it might be required to set this value to a lower value. The allocated heap size can be set by changing the value `perc_allocate` in the file `/etc/default/djigzo`

```
# 'dynamic' memory allocation
# set max memory based on the total available memory

# the percentage of total memory to allocate for the JVM
perc_allocate="0.6"
```

E Securing the gateway

E.1 Port usage

CipherMail uses the following ports:

external → internal

Port	Service	Description
22	SSH	Console access
25	SMTP	Send/Receive email
8080	HTTP	Web manager
8443	HTTPS	Web manager
9009	SOAP (HTTP)	Back end*

* By default the back-end SOAP service is only accessible from localhost (i.e., it is bound to localhost)

internal → external

Port	Service	Description
25	SMTP	Send/Receive email
80	HTTP	CRL download
139	SMB/CIFS	remote backup and restore
389	LDAP	CRL download
443	HTTPS	CRL download
445	SMB/CIFS	remote backup and restore

When the encryption back-end and Web GUI front-end are installed on the same machine, remote access to port 9000 is not required. It is advised to block remote access to all ports which are not used by CipherMail.

Enable Ubuntu firewall Ubuntu can be protected by installing the "Uncomplicate Firewall" (UFW) with the following commands:

```
$ sudo apt-get install ufw
$ sudo ufw allow smtp/tcp
$ sudo ufw allow ssh/tcp
$ sudo ufw allow 8443/tcp
$ sudo ufw allow 8080/tcp
$ sudo ufw enable
```

Red Hat/CentOS already comes with a pre-installed firewall.

E.2 Passwords

Database By default, CipherMail creates a database user *djigzo* with the password *djigzo*. If a different password should be used, the database password for user *djigzo* should be changed (see PostgreSQL documentation). The database password in the database configuration file /usr/share/djigzo/conf/hibernate.cfg.xml should be changed accordingly.

Back-end The front-end (Web GUI) communicates with the back-end (encryption engine) using password authenticated SOAP messages. If the back-end and front-end are not installed on the same machine, it is advised to change the SOAP password. For the back-end, the password can be changed by modifying the property **protected.system.soap.password** in file:

/usr/share/djigzo/conf/djigzo.properties.

If the password for the back-end is changed, the password used by the front-end should be changed accordingly. The password for the front-end can be changed by adding a property **soap.password** with the password as the property value to /etc/default/tomcat in a similar way as **djigzo-web.home** was set (see 4.7).

E.3 SSL certificate

Access to the administration page is protected with an encrypted HTTPS connection. CipherMail comes with a default SSL certificate. It is advised to install a new SSL certificate using the "SSL certificate manager" from the CipherMail Web GUI.

E.4 Prevent spoofing the From header

CipherMail uses the *From* header as the identity of the sender. If the CipherMail gateway is used for sending email to external recipients (i.e., relaying email), make sure that internal users cannot 'spoof' the *From* header.

E.5 Securing the database

Unless a "Hardware Security Module" (HSM) is used, all private keys used for signing and decrypting of email are stored in the database. The database therefore has to be protected against unauthorized access. If CipherMail and PostgreSQL are installed on the same machine, the djigzo database user should only be allowed to access the database locally. This is done by making sure that only localhost (127.0.0.1) can login with the username *djigzo*. The PostgreSQL config file `pg_hba.conf` should contain a line similar to:

```
host  djigzo  djigzo  127.0.0.1/32  md5
```

E.6 Block access to WEB GUI

If the PDF reply functionality is used, external access to the gateway should be granted to all external IP addresses (otherwise the recipients of the encrypted PDF message cannot open the reply page). It is advised to only allow access to the PDF reply pages and block access to all other pages.

Access to the following URLs should be granted for all IP addresses: `https://192.168.178.24:8443/web/portal/*` (the IP address should be the external IP address and * means that access should be granted to all parent URLs). There are multiple ways to block access to most of the gateway pages while allowing access to the PDF reply page:

Block access with a firewall If a firewall is used and the firewall is capable of blocking access at the HTTP(s) level, a rule should be added to block access to all URLs with the exception of the PDF reply page URL.

Use Apache as a front-end Use Apache as a front-end to the gateway. Apache will handle all HTTP(s) access. Apache can be setup to only allow access to certain URLs. Add a rule to block access to all URLs except to the PDF reply page URL.

Enable the built-in IP filter CipherMail can be setup to only allow access to the WEB GUI from certain IP addresses. To enable the IP filter, create the file `/etc/djigzo/ip-filter.properties` containing the allowed address ranges. The file should contain a property named `ip-filter` with the value set to the allowed IP range.

The `ip-filter` should be a comma separated list of IP addresses. An IP range can be specified either in CIDR format or with a wildcard (*).

Examples:

1. ip-filter=192.168.*
2. ip-filter=192.168.*, 127.*, 222.0.0.0/8