Red Hat Enterprise Linux 9

部署邮件服务器

配置和维护邮件服务器服务

Last Updated: 2024-06-26

Red Hat Enterprise Linux 9 部署邮件服务器

配置和维护邮件服务器服务

法律通告

摘要

在 Red Hat Enterprise Linux 上，您可以使用邮件传输代理 Postfix 作为 SMTP 服务,并使用邮件发送代理 Dovecot 作为 IMAP 和 POP3 服务，为您的客户和内部用户提供可靠的安全邮件服务。两种服务都相互集成，它们都支持中央后端，如用于存储帐户数据和用于验证用户的 LDAP 目录。

目录

对红帽文档提供反馈

我们感谢您对我们文档的反馈。让我们了解如何改进它。

通过 Jira 提交反馈（需要帐户）

1. 登录到 Jira 网站。
2. 在顶部导航栏中点 **Create**
3. 在 **Summary** 字段中输入描述性标题。
4. 在 **Description** 字段中输入您对改进的建议。包括文档相关部分的链接。
5. 点对话框底部的 **Create**。

第 1 章 配置和维护 DOVECOT IMAP 和 POP3 服务器

Dovecot 是一个高性能邮件发送代理(MDA)，专注于安全性。您可以使用 IMAP 或 POP3 兼容电子邮件客户端连接到 Dovecot 服务器，并读取或下载电子邮件。

Dovecot 的主要特性：

- 设计和实施侧重于安全性
- 对高可用性的双向复制支持以提高大型环境中的性能
- 支持高性能 **dbox** 邮箱格式，但出于兼容性的原因，也支持 **mbox** 和 **Maildir**
- 自我修复功能，如修复有问题的索引文件
- 遵守 IMAP 标准
- 临时解决方案支持绕过 IMAP 和 POP3 客户端中的 bug

1.1. 建立具有 PAM 验证的 DOVECOT 服务器

Dovecot 支持名称服务交换机(NSS)接口作为用户数据库，以及可插拔验证模块(PAM)框架作为身份验证后端。使用这个配置，Dovecot 可以通过 NSS 为服务器上的本地用户提供服务。

以下帐户使用 PAM 身份验证：

- 在 **/etc/passwd** 文件中本地定义的
- 存储在远程数据库中，但可以通过系统安全服务守护进程(SSSD)或其他 NSS 插件在本地提供。

1.1.1. 安装 Dovecot

dovecot 软件包提供：

- **dovecot** 服务以及维护它的工具
- Dovecot 按需启动的服务，如用于身份验证
- 插件，如服务器端的邮件过滤
- **/etc/dovecot/** 目录中的配置文件
- **/usr/share/doc/dovecot/** 目录中的文档

流程

- 安装 **dovecot** 软件包：

```
# dnf install dovecot
```

注意

如果 Dovecot 已安装，并且需要清理配置文件，请重命名或删除 **/etc/dovecot/** 目录。之后，重新安装软件包。在不删除配置文件的情况下，**dnf reinstall dovecot** 命令不会重置 **/etc/dovecot/** 中的配置文件。

后续步骤

- 在 Dovecot 服务器上配置 TLS 加密。

1.1.2. 在 Dovecot 服务器上配置 TLS 加密

Dovecot 提供一个安全的默认配置。例如，默认启用 TLS 通过网络来传输加密的凭证和数据。要在 Dovecot 服务器上配置 TLS，您只需设置证书和私钥文件的路径。另外，您可以通过生成并使用 Diffie-Hellman 参数来提供 perfect forward secrecy(PFS)来提高 TLS 连接的安全性。

先决条件

- Dovecot 已安装。
- 以下文件已复制到服务器上列出的位置：
  - 服务器证书：**/etc/pki/dovecot/certs/server.example.com.crt**
  - 私钥：**/etc/pki/dovecot/private/server.example.com.key**
  - 证书颁发机构(CA)证书：**/etc/pki/dovecot/certs/ca.crt**
- 服务器证书 **Subject DN** 字段中的主机名与服务器的完全限定域名(FQDN)匹配。
- 如果服务器运行 RHEL 9.2 或更高版本，并且启用了 FIPS 模式，则客户端必须支持 Extended Master Secret(EMS)扩展或使用 TLS 1.3。没有 EMS 的 TLS 1.2 连接会失败。如需更多信息，请参阅 强制 TLS 扩展"Extended Master Secret" 知识库文章。

流程

1. 对私钥文件设置安全权限：

```
# chown root:root /etc/pki/dovecot/private/server.example.com.key
# chmod 600 /etc/pki/dovecot/private/server.example.com.key
```

2. 使用 Diffie-Hellman 参数生成文件：

```
# openssl dhparam -out /etc/dovecot/dh.pem 4096
```

   根据服务器上的硬件和熵，生成 4096 位的 Diffie-Hellman 参数可能需要几分钟。

3. 在 **/etc/dovecot/conf.d/10-ssl.conf** 文件中设置证书和私钥文件的路径：
   a. 更新 **ssl_cert** 和 **ssl_key** 参数，并将其设置为使用服务器的证书和私钥的路径：

```
ssl_cert = </etc/pki/dovecot/certs/server.example.com.crt
ssl_key = </etc/pki/dovecot/private/server.example.com.key
```

   b. 取消 **ssl_ca** 参数的注释，并将其设置为使用 CA 证书的路径：

```
ssl_ca = </etc/pki/dovecot/certs/ca.crt
```

   c. 取消 **ssl_dh** 参数的注释，并将其设置为使用 Diffie-Hellman 参数文件的路径：

```
ssl_dh = </etc/dovecot/dh.pem
```

重要

为确保 Dovecot 从文件中读取参数的值，该路径必须以 **<** 字符开头。

后续步骤

- 准备 Dovecot 以使用虚拟用户

其他资源

- **/usr/share/doc/dovecot/wiki/SSL.DovecotConfiguration.txt**

1.1.3. 准备 Dovecot 以使用虚拟用户

默认情况下，Dovecot 以使用服务的用户的身份对文件系统执行许多操作。但是，将 Dovecot 后端配置为使用一个本地用户来执行这些操作有以下几点好处：

- Dovecot 以特定的本地用户身份执行文件系统操作，而不使用用户的 ID (UID)。
- 用户不需要在服务器上本地提供。
- 您可以将所有邮箱和特定于用户的文件存储在一个根目录中。
- 用户不需要 UID 和组 ID (GID)，这可以减少管理工作。
- 有权访问服务器上文件系统的用户无法破坏其邮箱或索引，因为它们无法访问这些文件。
- 设置复制很简单。

先决条件

- Dovecot 已安装。

流程

1. 创建 **vmail** 用户：

```
# useradd --home-dir /var/mail/ --shell /usr/sbin/nologin vmail
```

Dovecot 之后将使用此用户来管理邮箱。出于安全考虑，请不要为此使用 **dovecot** 或 **dovenull** 系统用户。

2. 如果您使用与 **/var/mail/** 不同的路径，请对其设置 **mail_spool_t** SELinux 上下文，例如：

```
# semanage fcontext -a -t mail_spool_t "<path>(/.*)?"
# restorecon -Rv <path>
```

3. 仅将 **/var/mail/** 的写权限授予 **vmail** 用户：

```
# chown vmail:vmail /var/mail/
# chmod 700 /var/mail/
```

4. 取消 **/etc/dovecot/conf.d/10-mail.conf** 文件中 **mail_location** 参数的注释，并将其设置为 mailbox 格式和位置：

```
mail_location = sdbox:/var/mail/%n/
```

使用这个设置：

- Dovecot 在 **single** 模式下使用高性能 **dbox** 邮箱格式。在此模式下，服务将每个邮件存储在单独的文件中，类似于 **maildir** 格式。
- Dovecot 将路径中的 **%n** 变量解析为用户名。这需要确保每个用户对其邮箱都有一个单独的目录。

后续步骤

- 使用 PAM 作为 Dovecot 身份验证后端。

其他资源

- **/usr/share/doc/dovecot/wiki/VirtualUsers.txt**
- **/usr/share/doc/dovecot/wiki/MailLocation.txt**
- **/usr/share/doc/dovecot/wiki/MailboxFormat.dbox.txt**
- **/usr/share/doc/dovecot/wiki/Variables.txt**

1.1.4. 使用 PAM 作为 Dovecot 身份验证后端

默认情况下，Dovecot 使用名称服务交换机(NSS)接口作为用户数据库，使用可插拔验证模块(PAM)框架作为身份验证后端。

自定义设置，以使 Dovecot 适应您的环境，并使用虚拟用户功能简化管理。

先决条件

- Dovecot 已安装。
- 虚拟用户功能已配置。

流程

1. 更新 **/etc/dovecot/conf.d/10-mail.conf** 文件中的 **first_valid_uid** 参数，以定义可以验证到 Dovecot 的最低用户 ID (UID)：

```
first_valid_uid = 1000
```

默认情况下，UID 大于或等于 **1000** 的用户可以进行身份验证。如果需要，您也可以设置 **last_valid_uid** 参数，以定义 Dovecot 允许登录的最高 UID。

2. 在 **/etc/dovecot/conf.d/auth-system.conf.ext** 文件中，将 **override_fields** 参数添加到 **userdb** 部分，如下所示：

```
userdb {
  driver = passwd
```

```
  override_fields = uid=vmail gid=vmail home=/var/mail/%n/
}
```

由于值固定，Dovecot 不会从 **/etc/passwd** 文件查询这些设置。因此，**/etc/passwd** 中定义的主目录不需要存在。

后续步骤

- 完成 Dovecot 配置。

其他资源

- **/usr/share/doc/dovecot/wiki/PasswordDatabase.PAM.txt**
- **/usr/share/doc/dovecot/wiki/VirtualUsers.Home.txt**

1.1.5. 完成 Dovecot 配置

安装和配置 Dovecot 后，在 **firewalld** 服务中打开所需的端口，然后启用并启动服务。之后，您可以测试服务器。

先决条件

- 在 Dovecot 中已配置了以下内容：
  - TLS 加密
  - 身份验证后端
- 客户端信任证书颁发机构(CA)证书。

流程

1. 如果您只想向用户提供 IMAP 或 POP3 服务，请取消 **/etc/dovecot/dovecot.conf** 文件中**protocols** 参数的注释，并将其设置为所需的协议。例如，如果您不需要 POP3，请设置：

   ```
   protocols = imap lmtp
   ```

   默认情况下启用 **imap**、**pop3** 和 **lmtp** 协议。

2. 在本地防火墙中打开端口。例如，要为 IMAPS、IMAP、POP3S 和 POP3 协议打开端口，请输入：

   ```
   # firewall-cmd --permanent --add-service=imaps --add-service=imap --add-service=pop3s --add-service=pop3
   # firewall-cmd --reload
   ```

3. 启用并启动 **dovecot** 服务：

   ```
   # systemctl enable --now dovecot
   ```

验证

1. 使用 Mozilla Thunderbird 等邮件客户端连接到 Dovecot，并读取电子邮件。邮件客户端的设置取决于您要使用的协议：

   表 1.1. 到 Dovecot 服务器的连接设置

协议	端口	连接安全性	身份验证方法
IMAP	143	STARTTLS	PLAIN[a]
IMAPS	993	SSL/TLS	PLAIN[a]
POP3	110	STARTTLS	PLAIN[a]
POP3S	995	SSL/TLS	PLAIN[a]

[a] 客户端通过 TLS 连接传输加密的数据。因此，凭证不会被披露。

   请注意，这个表不会列出未加密连接的设置，因为默认情况下，Dovecot 在没有 TLS 的连接上不接受纯文本身份验证。

2. 显示具有非默认值的配置设置：

```
# doveconf -n
```

其他资源

- **firewall-cmd(1)** 手册页

1.2. 设置具有 LDAP 身份验证的 DOVECOT 服务器

如果您的基础架构使用 LDAP 服务器来存储帐户，您可以对其验证 Dovecot 用户。在这种情况下，您可以在目录中集中管理帐户，用户不需要对 Dovecot 服务器上的文件系统进行本地访问。

如果您计划设置具有复制的多个 Dovecot 服务器，以使您的邮箱具有高可用性，则集中管理的帐户也是一个好处。

1.2.1. 安装 Dovecot

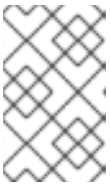
dovecot 软件包提供：

- **dovecot** 服务以及维护它的工具
- Dovecot 按需启动的服务，如用于身份验证
- 插件，如服务器端的邮件过滤
- **/etc/dovecot/** 目录中的配置文件
- **/usr/share/doc/dovecot/** 目录中的文档

流程

- 安装 **dovecot** 软件包：

```
# dnf install dovecot
```



注意

如果 Dovecot 已安装，并且需要清理配置文件，请重命名或删除 **/etc/dovecot/** 目录。之后，重新安装软件包。在不删除配置文件的情况下，**dnf reinstall dovecot** 命令不会重置 **/etc/dovecot/** 中的配置文件。

后续步骤

- 在 Dovecot 服务器上配置 TLS 加密。

1.2.2. 在 Dovecot 服务器上配置 TLS 加密

Dovecot 提供一个安全的默认配置。例如，默认启用 TLS 通过网络来传输加密的凭证和数据。要在 Dovecot 服务器上配置 TLS，您只需设置证书和私钥文件的路径。另外，您可以通过生成并使用 Diffie-Hellman 参数来提供 perfect forward secrecy(PFS)来提高 TLS 连接的安全性。

先决条件

- Dovecot 已安装。
- 以下文件已复制到服务器上列出的位置：
  - 服务器证书：**/etc/pki/dovecot/certs/server.example.com.crt**
  - 私钥：**/etc/pki/dovecot/private/server.example.com.key**
  - 证书颁发机构(CA)证书：**/etc/pki/dovecot/certs/ca.crt**
- 服务器证书 **Subject DN** 字段中的主机名与服务器的完全限定域名(FQDN)匹配。
- 如果服务器运行 RHEL 9.2 或更高版本，并且启用了 FIPS 模式，则客户端必须支持 Extended Master Secret(EMS)扩展或使用 TLS 1.3。没有 EMS 的 TLS 1.2 连接会失败。如需更多信息，请参阅 强制 TLS 扩展"Extended Master Secret" 知识库文章。

流程

1. 对私钥文件设置安全权限：

```
# chown root:root /etc/pki/dovecot/private/server.example.com.key
# chmod 600 /etc/pki/dovecot/private/server.example.com.key
```

2. 使用 Diffie-Hellman 参数生成文件：

```
# openssl dhparam -out /etc/dovecot/dh.pem 4096
```

根据服务器上的硬件和熵，生成 4096 位的 Diffie-Hellman 参数可能需要几分钟。

3. 在 **/etc/dovecot/conf.d/10-ssl.conf** 文件中设置证书和私钥文件的路径：

   a. 更新 **ssl_cert** 和 **ssl_key** 参数，并将其设置为使用服务器的证书和私钥的路径：

```
ssl_cert = </etc/pki/dovecot/certs/server.example.com.crt
ssl_key = </etc/pki/dovecot/private/server.example.com.key
```

b. 取消 **ssl_ca** 参数的注释，并将其设置为使用 CA 证书的路径：

```
ssl_ca = </etc/pki/dovecot/certs/ca.crt
```

c. 取消 **ssl_dh** 参数的注释，并将其设置为使用 Diffie-Hellman 参数文件的路径：

```
ssl_dh = </etc/dovecot/dh.pem
```

重要

为确保 Dovecot 从文件中读取参数的值，该路径必须以 **<** 字符开头。

后续步骤

- 准备 Dovecot 以使用虚拟用户

其他资源

- **/usr/share/doc/dovecot/wiki/SSL.DovecotConfiguration.txt**

1.2.3. 准备 Dovecot 以使用虚拟用户

默认情况下，Dovecot 以使用服务的用户的身份对文件系统执行许多操作。但是，将 Dovecot 后端配置为使用一个本地用户来执行这些操作有以下几点好处：

- Dovecot 以特定的本地用户身份执行文件系统操作，而不使用用户的 ID (UID)。
- 用户不需要在服务器上本地提供。
- 您可以将所有邮箱和特定于用户的文件存储在一个根目录中。
- 用户不需要 UID 和组 ID (GID)，这可以减少管理工作。
- 有权访问服务器上文件系统的用户无法破坏其邮箱或索引，因为它们无法访问这些文件。
- 设置复制很简单。

先决条件

- Dovecot 已安装。

流程

1. 创建 **vmail** 用户：

```
# useradd --home-dir /var/mail/ --shell /usr/sbin/nologin vmail
```

Dovecot 之后将使用此用户来管理邮箱。出于安全考虑，请不要为此使用 **dovecot** 或 **dovenull** 系统用户。

2. 如果您使用与 **/var/mail/** 不同的路径，请对其设置 **mail_spool_t** SELinux 上下文，例如：

```
# semanage fcontext -a -t mail_spool_t "<path>(/.*)?"
# restorecon -Rv <path>
```

3. 仅将 **/var/mail/** 的写权限授予 **vmail** 用户：

```
# chown vmail:vmail /var/mail/
# chmod 700 /var/mail/
```

4. 取消 **/etc/dovecot/conf.d/10-mail.conf** 文件中 **mail_location** 参数的注释，并将其设置为 mailbox 格式和位置：

```
mail_location = sdbox:/var/mail/%n/
```

使用这个设置：

- Dovecot 在 **single** 模式下使用高性能 **dbox** 邮箱格式。在此模式下，服务将每个邮件存储在单独的文件中，类似于 **maildir** 格式。
- Dovecot 将路径中的 **%n** 变量解析为用户名。这需要确保每个用户对其邮箱都有一个单独的目录。

后续步骤

- 使用 LDAP 作为 Dovecot 身份验证后端.

其他资源

- **/usr/share/doc/dovecot/wiki/VirtualUsers.txt**
- **/usr/share/doc/dovecot/wiki/MailLocation.txt**
- **/usr/share/doc/dovecot/wiki/MailboxFormat.dbox.txt**
- **/usr/share/doc/dovecot/wiki/Variables.txt**

1.2.4. 使用 LDAP 作为 Dovecot 身份验证后端

LDAP 目录中的用户通常可以向目录服务进行身份验证。Dovecot 可在用户登录到 IMAP 和 POP3 服务时使用此来验证它们。这个验证方法有很多优点，例如：

- 管理员可以在目录中集中管理用户。
- LDAP 帐户不需要任何特殊属性。它们只需要能够向 LDAP 服务器进行身份验证。因此，此方法独立于 LDAP 服务器上使用的密码存储方案。
- 用户不需要通过名称服务交换机(NSS)界面和可插拔验证模块(PAM)框架在服务器上本地提供。

先决条件

- Dovecot 已安装。
- 虚拟用户功能已配置。

- 到 LDAP 服务器的连接支持 TLS 加密。
- Dovecot 服务器上的 RHEL 信任 LDAP 服务器的证书颁发机构(CA)证书。
- 如果用户存储在 LDAP 目录中的不同树中，则存在用于 Dovecot 的专用 LDAP 帐户，以搜索目录。此帐户需要搜索其他用户的可辨识名称(DN)的权限。
- 如果 MariaDB 服务器运行 RHEL 9.2 或更高版本，并且启用了 FIPS 模式，则这个 Dovecot 服务器支持 Extended Master Secret (EMS)扩展或使用 TLS 1.3。没有 EMS 的 TLS 1.2 连接会失败。如需更多信息，请参阅 强制 TLS 扩展"Extended Master Secret" 知识库文章。

流程

1. 在 **/etc/dovecot/conf.d/10-auth.conf** 文件中配置身份验证后端：
   a. 注释掉您不需要的 **auth-*.conf.ext** 身份验证后端配置文件的 **include** 语句，例如：

   ```
   #!include auth-system.conf.ext
   ```

   b. 通过取消下列行的注释来启用 LDAP 身份验证：

   ```
   !include auth-ldap.conf.ext
   ```

2. 编辑 **/etc/dovecot/conf.d/auth-ldap.conf.ext** 文件，并按如下所示将 **override_fields** 参数添加到 **userdb** 部分：

   ```
   userdb {
     driver = ldap
     args = /etc/dovecot/dovecot-ldap.conf.ext
     override_fields = uid=vmail gid=vmail home=/var/mail/%n/
   }
   ```

   由于值固定，Dovecot 不会从 LDAP 服务器查询这些设置。因此，这些属性也不是必须出现。

3. 使用以下设置创建 **/etc/dovecot/dovecot-ldap.conf.ext** 文件：
   a. 根据 LDAP 结构，配置以下之一：
      - 如果用户存储在 LDAP 目录中的不同树中，请配置动态 DN 查找：

        ```
        dn = cn=dovecot_LDAP,dc=example,dc=com
        dnpass = password
        pass_filter = (&(objectClass=posixAccount)(uid=%n))
        ```

        Dovecot 使用指定的 DN、密码和过滤器在目录中搜索身份验证用户的 DN。在此搜索中，Dovecot 将过滤器中的 **%n** 替换为用户名。请注意，LDAP 搜索必须只返回一个结果。

      - 如果所有用户都存储在特定条目下，请配置 DN 模板：

        ```
        auth_bind_userdn = cn=%n,ou=People,dc=example,dc=com
        ```

   b. 启用绑定到 LDAP 服务器的身份验证以验证 Dovecot 用户：

   ```
   auth_bind = yes
   ```

c. 将 URL 设置为 LDAP 服务器：

```
uris = ldaps://LDAP-srv.example.com
```

为安全起见，请只使用使用 LDAPS 的加密连接，或使用通过 LDAP 协议的 **STARTTLS** 命令。对于后者，在设置中额外添加 **tls = yes**。

对于正常工作的证书验证，LDAP 服务器的主机名必须与其 TLS 证书中使用的主机名匹配。

d. 启用 LDAP 服务器的 TLS 证书的验证：

```
tls_require_cert = hard
```

e. 将基本 DN 设置为要开始搜索用户的 DN：

```
base = ou=People,dc=example,dc=com
```

f. 设置搜索范围：

```
scope = onelevel
```

Dovecot 仅在指定的基本 DN 中使用 **onelevel** 范围搜索，并且也使用子树中的 **subtree** 范围搜索。

4. 对 **/etc/dovecot/dovecot-ldap.conf.ext** 文件设置安全权限：

```
# chown root:root /etc/dovecot/dovecot-ldap.conf.ext
# chmod 600 /etc/dovecot/dovecot-ldap.conf.ext
```

后续步骤

- 完成 Dovecot 配置。

其他资源

- **/usr/share/doc/dovecot/example-config/dovecot-ldap.conf.ext**
- **/usr/share/doc/dovecot/wiki/UserDatabase.Static.txt**
- **/usr/share/doc/dovecot/wiki/AuthDatabase.LDAP.txt**
- **/usr/share/doc/dovecot/wiki/AuthDatabase.LDAP.AuthBinds.txt**
- **/usr/share/doc/dovecot/wiki/AuthDatabase.LDAP.PasswordLookups.txt**

1.2.5. 完成 Dovecot 配置

安装和配置 Dovecot 后，在 **firewalld** 服务中打开所需的端口，然后启用并启动服务。之后，您可以测试服务器。

先决条件

- 在 Dovecot 中已配置了以下内容：

- TLS 加密
- 身份验证后端

- 客户端信任证书颁发机构(CA)证书。

流程

1. 如果您只想向用户提供 IMAP 或 POP3 服务，请取消 **/etc/dovecot/dovecot.conf** 文件中**protocols** 参数的注释，并将其设置为所需的协议。例如，如果您不需要 POP3，请设置：

```
protocols = imap lmtp
```

默认情况下启用 **imap**、**pop3** 和 **lmtp** 协议。

2. 在本地防火墙中打开端口。例如，要为 IMAPS、IMAP、POP3S 和 POP3 协议打开端口，请输入：

```
# firewall-cmd --permanent --add-service=imaps --add-service=imap --add-service=pop3s --add-service=pop3
# firewall-cmd --reload
```

3. 启用并启动 **dovecot** 服务：

```
# systemctl enable --now dovecot
```

验证

1. 使用 Mozilla Thunderbird 等邮件客户端连接到 Dovecot，并读取电子邮件。邮件客户端的设置取决于您要使用的协议：

表 1.2. 到 Dovecot 服务器的连接设置

协议	端口	连接安全性	身份验证方法
IMAP	143	STARTTLS	PLAIN[a]
IMAPS	993	SSL/TLS	PLAIN[a]
POP3	110	STARTTLS	PLAIN[a]
POP3S	995	SSL/TLS	PLAIN[a]

[a] 客户端通过 TLS 连接传输加密的数据。因此，凭证不会被披露。

请注意，这个表不会列出未加密连接的设置，因为默认情况下，Dovecot 在没有 TLS 的连接上不接受纯文本身份验证。

2. 显示具有非默认值的配置设置：

```
# doveconf -n
```

其他资源

- **firewall-cmd(1)** 手册页

1.3. 设置具有 MARIADB SQL 身份验证的 DOVECOT 服务器

如果您将用户和密码存储在 MariaDB SQL 服务器中，您可以将 Dovecot 配置为将其用作用户数据库和身份验证后端。使用这个配置，您可以在数据库中集中管理帐户，用户对 Dovecot 服务器上的文件系统没有本地访问权限。

如果您计划设置具有复制的多个 Dovecot 服务器，以使您的邮箱具有高可用性，则集中管理的帐户也是一个好处。

1.3.1. 安装 Dovecot

dovecot 软件包提供：

- **dovecot** 服务以及维护它的工具
- Dovecot 按需启动的服务，如用于身份验证
- 插件，如服务器端的邮件过滤
- **/etc/dovecot/** 目录中的配置文件
- **/usr/share/doc/dovecot/** 目录中的文档

流程

- 安装 **dovecot** 软件包：

```
# dnf install dovecot
```

注意

如果 Dovecot 已安装，并且需要清理配置文件，请重命名或删除 **/etc/dovecot/** 目录。之后，重新安装软件包。在不删除配置文件的情况下，**dnf reinstall dovecot** 命令不会重置 **/etc/dovecot/** 中的配置文件。

后续步骤

- 在 Dovecot 服务器上配置 TLS 加密。

1.3.2. 在 Dovecot 服务器上配置 TLS 加密

Dovecot 提供一个安全的默认配置。例如，默认启用 TLS 通过网络来传输加密的凭证和数据。要在 Dovecot 服务器上配置 TLS，您只需设置证书和私钥文件的路径。另外，您可以通过生成并使用 Diffie-Hellman 参数来提供 perfect forward secrecy(PFS)来提高 TLS 连接的安全性。

先决条件

- Dovecot 已安装。
- 以下文件已复制到服务器上列出的位置：
  - 服务器证书：**/etc/pki/dovecot/certs/server.example.com.crt**
  - 私钥：**/etc/pki/dovecot/private/server.example.com.key**
  - 证书颁发机构(CA)证书：**/etc/pki/dovecot/certs/ca.crt**
- 服务器证书 **Subject DN** 字段中的主机名与服务器的完全限定域名(FQDN)匹配。
- 如果服务器运行 RHEL 9.2 或更高版本，并且启用了 FIPS 模式，则客户端必须支持 Extended Master Secret(EMS)扩展或使用 TLS 1.3。没有 EMS 的 TLS 1.2 连接会失败。如需更多信息，请参阅 强制 TLS 扩展"Extended Master Secret" 知识库文章。

流程

1. 对私钥文件设置安全权限：

```
# chown root:root /etc/pki/dovecot/private/server.example.com.key
# chmod 600 /etc/pki/dovecot/private/server.example.com.key
```

2. 使用 Diffie-Hellman 参数生成文件：

```
# openssl dhparam -out /etc/dovecot/dh.pem 4096
```

   根据服务器上的硬件和熵，生成 4096 位的 Diffie-Hellman 参数可能需要几分钟。

3. 在 **/etc/dovecot/conf.d/10-ssl.conf** 文件中设置证书和私钥文件的路径：
   a. 更新 **ssl_cert** 和 **ssl_key** 参数，并将其设置为使用服务器的证书和私钥的路径：

```
ssl_cert = </etc/pki/dovecot/certs/server.example.com.crt
ssl_key = </etc/pki/dovecot/private/server.example.com.key
```

   b. 取消 **ssl_ca** 参数的注释，并将其设置为使用 CA 证书的路径：

```
ssl_ca = </etc/pki/dovecot/certs/ca.crt
```

   c. 取消 **ssl_dh** 参数的注释，并将其设置为使用 Diffie-Hellman 参数文件的路径：

```
ssl_dh = </etc/dovecot/dh.pem
```

重要

为确保 Dovecot 从文件中读取参数的值，该路径必须以 **<** 字符开头。

后续步骤

- 准备 Dovecot 以使用虚拟用户

其他资源

- **/usr/share/doc/dovecot/wiki/SSL.DovecotConfiguration.txt**

1.3.3. 准备 Dovecot 以使用虚拟用户

默认情况下，Dovecot 以使用服务的用户的身份对文件系统执行许多操作。但是，将 Dovecot 后端配置为使用一个本地用户来执行这些操作有以下几点好处：

- Dovecot 以特定的本地用户身份执行文件系统操作，而不使用用户的 ID (UID)。
- 用户不需要在服务器上本地提供。
- 您可以将所有邮箱和特定于用户的文件存储在一个根目录中。
- 用户不需要 UID 和组 ID (GID)，这可以减少管理工作。
- 有权访问服务器上文件系统的用户无法破坏其邮箱或索引，因为它们无法访问这些文件。
- 设置复制很简单。

先决条件

- Dovecot 已安装。

流程

1. 创建 **vmail** 用户：

```
# useradd --home-dir /var/mail/ --shell /usr/sbin/nologin vmail
```

   Dovecot 之后将使用此用户来管理邮箱。出于安全考虑，请不要为此使用 **dovecot** 或 **dovenull** 系统用户。

2. 如果您使用与 **/var/mail/** 不同的路径，请对其设置 **mail_spool_t** SELinux 上下文，例如：

```
# semanage fcontext -a -t mail_spool_t "<path>(/.*)?"
# restorecon -Rv <path>
```

3. 仅将 **/var/mail/** 的写权限授予 **vmail** 用户：

```
# chown vmail:vmail /var/mail/
# chmod 700 /var/mail/
```

4. 取消 **/etc/dovecot/conf.d/10-mail.conf** 文件中 **mail_location** 参数的注释，并将其设置为 mailbox 格式和位置：

```
mail_location = sdbox:/var/mail/%n/
```

   使用这个设置：

   - Dovecot 在 **single** 模式下使用高性能 **dbox** 邮箱格式。在此模式下，服务将每个邮件存储在单独的文件中，类似于 **maildir** 格式。
   - Dovecot 将路径中的 **%n** 变量解析为用户名。这需要确保每个用户对其邮箱都有一个单独的目录。

后续步骤

- 使用 MariaDB SQL 数据库作为 Dovecot 身份验证后端

其他资源

- **/usr/share/doc/dovecot/wiki/VirtualUsers.txt**
- **/usr/share/doc/dovecot/wiki/MailLocation.txt**
- **/usr/share/doc/dovecot/wiki/MailboxFormat.dbox.txt**
- **/usr/share/doc/dovecot/wiki/Variables.txt**

1.3.4. 使用 MariaDB SQL 数据库作为 Dovecot 身份验证后端

Dovecot 可以从 MariaDB 数据库读取帐户和密码，并在用户登录到 IMAP 或 POP3 服务时使用它来验证用户。这个验证方法的好处包括：

- 管理员可以在数据库中集中管理用户。
- 用户在服务器上没有本地访问权限。

先决条件

- Dovecot 已安装。
- 虚拟用户功能已配置。
- 到 MariaDB 服务器的连接支持 TLS 加密。
- MariaDB 中存在 **dovecotDB** 数据库，**users** 表至少包含 **username** 和 **password** 列。
- **password** 列包含使用 Dovecot 支持的方案加密的密码。
- 密码可以使用相同的方案，或者有 **{*pw-storage-scheme*}** 前缀。
- **dovecot** MariaDB 用户对 **dovecotDB** 数据库中的 **users** 表有读权限。
- 发布 MariaDB 服务器的 TLS 证书的证书颁发机构(CA)的证书存储在 Dovecot 服务器上的 **/etc/pki/tls/certs/ca.crt** 文件中。
- 如果 MariaDB 服务器运行 RHEL 9.2 或更高版本，并且启用了 FIPS 模式，则这个 Dovecot 服务器支持 Extended Master Secret (EMS)扩展或使用 TLS 1.3。没有 EMS 的 TLS 1.2 连接会失败。如需更多信息，请参阅 强制 TLS 扩展"Extended Master Secret" 知识库文章。

流程

1. 安装 **dovecot-mysql** 软件包：

```
# dnf install dovecot-mysql
```

2. 在 **/etc/dovecot/conf.d/10-auth.conf** 文件中配置身份验证后端：

   a. 注释掉您不需要的 **auth-*.conf.ext** 身份验证后端配置文件的 **include** 语句，例如：

```
#!include auth-system.conf.ext
```

b. 通过取消以下行的注释来启用 SQL 身份验证：

```
!include auth-sql.conf.ext
```

3. 编辑 **/etc/dovecot/conf.d/auth-sql.conf.ext** 文件，并将 **override_fields** 参数添加到 **userdb** 部分，如下所示：

```
userdb {
  driver = sql
  args = /etc/dovecot/dovecot-sql.conf.ext
  override_fields = uid=vmail gid=vmail home=/var/mail/%n/
}
```

由于值固定，Dovecot 不会从 SQL 服务器查询这些设置。

4. 使用以下设置创建 **/etc/dovecot/dovecot-sql.conf.ext** 文件：

```
driver = mysql
connect = host=mariadb_srv.example.com dbname=dovecotDB user=dovecot
password=dovecotPW ssl_ca=/etc/pki/tls/certs/ca.crt
default_pass_scheme = SHA512-CRYPT
user_query = SELECT username FROM users WHERE username='%u';
password_query = SELECT username AS user, password FROM users WHERE
username='%u';
iterate_query = SELECT username FROM users;
```

要对数据库服务器使用 TLS 加密，请将 **ssl_ca** 选项设置为发布 MariaDB 服务器证书的 CA 的证书路径。对于正常工作的证书验证，MariaDB 服务器的主机名必须与其 TLS 证书中使用的主机名匹配。

如果数据库中的密码值包含 **{*pw-storage-scheme*}** 前缀，则您可以省略 **default_pass_scheme** 设置。

文件中的查询必须设置如下：

- 对于 **user_query** 参数，查询必须返回 Dovecot 用户的用户名。查询还必须只返回一个结果。
- 对于 **password_query** 参数，查询必须返回用户名和密码，并且 Dovecot 必须在 **user** 和 **password** 变量中使用这些值。因此，如果数据库使用不同的列名称，请使用 **AS** SQL 命令重命名结果中的列。
- 对于 **iterate_query** 参数，查询必须返回所有用户的列表。

5. 对 **/etc/dovecot/dovecot-sql.conf.ext** 文件设置安全权限：

```
# chown root:root /etc/dovecot/dovecot-sql.conf.ext
# chmod 600 /etc/dovecot/dovecot-sql.conf.ext
```

后续步骤

- 完成 Dovecot 配置。

其他资源

- **/usr/share/doc/dovecot/example-config/dovecot-sql.conf.ext**
- **/usr/share/doc/dovecot/wiki/Authentication.PasswordSchemes.txt**

1.3.5. 完成 Dovecot 配置

安装和配置 Dovecot 后，在 **firewalld** 服务中打开所需的端口，然后启用并启动服务。之后，您可以测试服务器。

先决条件

- 在 Dovecot 中已配置了以下内容：
  - TLS 加密
  - 身份验证后端
- 客户端信任证书颁发机构(CA)证书。

流程

1. 如果您只想向用户提供 IMAP 或 POP3 服务，请取消 **/etc/dovecot/dovecot.conf** 文件中**protocols** 参数的注释，并将其设置为所需的协议。例如，如果您不需要 POP3，请设置：

   ```
   protocols = imap lmtp
   ```

   默认情况下启用 **imap**、**pop3** 和 **lmtp** 协议。

2. 在本地防火墙中打开端口。例如，要为 IMAPS、IMAP、POP3S 和 POP3 协议打开端口，请输入：

   ```
   # firewall-cmd --permanent --add-service=imaps --add-service=imap --add-service=pop3s --add-service=pop3
   # firewall-cmd --reload
   ```

3. 启用并启动 **dovecot** 服务：

   ```
   # systemctl enable --now dovecot
   ```

验证

1. 使用 Mozilla Thunderbird 等邮件客户端连接到 Dovecot，并读取电子邮件。邮件客户端的设置取决于您要使用的协议：

   表 1.3. 到 Dovecot 服务器的连接设置

协议	端口	连接安全性	身份验证方法
IMAP	143	STARTTLS	PLAIN[a]
IMAPS	993	SSL/TLS	PLAIN[a]

协议	端口	连接安全性	身份验证方法
POP3	110	STARTTLS	PLAIN[a]
POP3S	995	SSL/TLS	PLAIN[a]

[a] 客户端通过 TLS 连接传输加密的数据。因此，凭证不会被披露。

请注意，这个表不会列出未加密连接的设置，因为默认情况下，Dovecot 在没有 TLS 的连接上不接受纯文本身份验证。

2. 显示具有非默认值的配置设置：

```
# doveconf -n
```

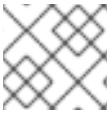
其他资源

- **firewall-cmd(1)** 手册页

1.4. 在两个 DOVECOT 服务器之间配置复制

通过双向复制，您可以使 Dovecot 服务器高度可用，而 IMAP 和 POP3 客户端都可以访问这两个服务器上的邮箱。Dovecot 会跟踪每个邮箱的索引日志中的更改，并以安全的方式解决冲突。

在两个复制合作伙伴上执行这个流程。



注意

复制只在服务器对之间正常工作。因此，在大型集群中，您需要多个独立的后端对。

先决条件

- 两个服务器都使用相同的身份验证后端。最好使用 LDAP 或 SQL 来集中维护帐户。
- Dovecot 用户数据库配置支持用户列表。使用 **doveadm user '*'** 命令来验证这一点。
- Dovecot 以 **vmail** 用户身份而不是用户的 ID (UID)访问文件系统上的邮箱。

流程

1. 创建 **/etc/dovecot/conf.d/10-replication.conf** 文件，并在其中执行以下步骤：
   a. 启用 **notify** 和 **replication** 插件：

```
mail_plugins = $mail_plugins notify replication
```

   b. 添加 **service replicator** 部分：

```
service replicator {
  process_min_avail = 1
```

```
  unix_listener replicator-doveadm {
    mode = 0600
    user = vmail
  }
}
```

使用这些设置，当 **dovecot** 服务启动时，Dovecot 会至少启动一个 replicator 进程。另外，本节定义了对 **replicator-doveadm** 套接字的设置。

c. 添加 **service aggregator** 部分来配置 **replication-notify-fifo** 管道和 **replication-notify** 套接字：

```
service aggregator {
  fifo_listener replication-notify-fifo {
    user = vmail
  }
  unix_listener replication-notify {
    user = vmail
  }
}
```

d. 添加 **service doveadm** 部分来定义复制服务的端口：

```
service doveadm {
  inet_listener {
    port = 12345
  }
}
```

e. 设置 **doveadm** 复制服务的密码：

```
doveadm_password = replication_password
```

两个服务器上的密码必须相同。

f. 配置复制伙伴：

```
plugin {
  mail_replica = tcp:server2.example.com:12345
}
```

g. 可选：定义并行 **dsync** 进程的最大数量：

```
replication_max_conns = 20
```

replication_max_conns 的默认值为 **10**。

2. 对 **/etc/dovecot/conf.d/10-replication.conf** 文件设置安全权限：

```
# chown root:root /etc/dovecot/conf.d/10-replication.conf
# chmod 600 /etc/dovecot/conf.d/10-replication.conf
```

3. 启用 **nis_enabled** SELinux 布尔值，以允许 Dovecot 打开 **doveadm** 复制端口：

```
setsebool -P nis_enabled on
```

4. 将 **firewalld** 规则配置为只允许复制伙伴访问复制端口，例如：

```
# firewall-cmd --permanent --zone=public --add-rich-rule="rule family="ipv4" source
address="192.0.2.1/32" port protocol="tcp" port="12345" accept"
# firewall-cmd --permanent --zone=public --add-rich-rule="rule family="ipv6" source
address="2001:db8:2::1/128" port protocol="tcp" port="12345" accept"
# firewall-cmd --reload
```

IPv4 的子网掩码 **/32** 和 IPv6 子网掩码 **/128** 限制对指定地址的访问。

5. 在其他复制伙伴上也执行这个流程。

6. 重新载入 Dovecot：

```
# systemctl reload dovecot
```

验证

1. 在一个服务器上的邮箱中执行一个操作，然后验证 Dovecot 是否将更改复制到其他服务器上。

2. 显示 replicator 状态：

```
# doveadm replicator status
Queued 'sync' requests       0
Queued 'high' requests       0
Queued 'low' requests        0
Queued 'failed' requests     0
Queued 'full resync' requests 30
Waiting 'failed' requests    0
Total number of known users  75
```

3. 显示特定用户的 replicator 状态：

```
# doveadm replicator status example_user
username        priority  fast sync  full sync  success sync  failed
example_user    none      02:05:28   04:19:07   02:05:28      -
```

其他资源

- **dsync(1)** 手册页
- **/usr/share/doc/dovecot/wiki/Replication.txt**

1.5. 向 IMAP 邮箱自动订阅用户

通常，IMAP 服务器管理员希望 Dovecot 自动创建某些邮箱，如 **Sent** 和 **Trash**，并向它们订阅用户。您可以在配置文件中设置它。

另外，您可以定义 *特殊用途邮箱*。IMAP 客户端通常支持为特殊用途定义邮箱，如用于发送电子邮件。为避免用户必须手动选择和设置正确的邮箱，IMAP 服务器可以在 IMAP **LIST** 命令中发送 **special-use** 属性。然后，客户端可以使用此属性来识别和设置，例如：发送电子邮件的邮箱。

先决条件

- Dovecot 已配置。

流程

1. 更新 **/etc/dovecot/conf.d/15-mailboxes.conf** 文件中的 **inbox** 命名空间部分：

   a. 将 **auto = subscribe** 设置添加到应该可供用户使用的每个特殊用途邮箱中，例如：

```
namespace inbox {
  ...
  mailbox Drafts {
    special_use = \Drafts
    auto = subscribe
  }

  mailbox Junk {
    special_use = \Junk
    auto = subscribe
  }

  mailbox Trash {
    special_use = \Trash
    auto = subscribe
  }

  mailbox Sent {
    special_use = \Sent
    auto = subscribe
  }
  ...
}
```

   如果您的邮件客户端支持更多特殊用途邮箱，您可以添加类似的条目。**special_use** 参数定义 Dovecot 在 **special-use** 属性中向客户端发送的值。

   b. 可选：如果要定义没有特殊用途的其他邮箱，请在用户的 inbox 中为其添加 **mailbox** 部分，例如：

```
namespace inbox {
  ...
  mailbox "Important Emails" {
    auto = <value>
  }
  ...
}
```

   您可以将 **auto** 参数设置为以下值之一：

   - **subscribe** ：自动创建邮箱并向其订阅用户。
   - **create** ：自动创建邮箱，而无需向其订阅用户。
   - **no**（默认）：Dovecot 不会创建邮箱，也不会向其订阅用户。

2. 重新载入 Dovecot：

```
# systemctl reload dovecot
```

验证

- 使用 IMAP 客户端访问您的邮箱。
  带有 **auto = subscribe** 设置的邮箱会自动可见。如果客户端支持特殊用途的邮箱并定义了用途，客户端会自动使用它们。

其他资源

- RFC 6154：用于特殊用途邮箱的 IMAP LIST 扩展
- **/usr/share/doc/dovecot/wiki/MailboxSettings.txt**

1.6. 配置 LMTP 套接字和 LMTPS 侦听器

SMTP 服务器（如 Postfix）使用本地邮件传输协议(LMTP)向 Dovecot 发送电子邮件。如果 SMTP 服务器运行：

- 在与 Dovecot 相同的主机上，使用 LMTP 套接字
- 在其他主机上，使用 LMTP 服务
  默认情况下，LMTP 协议没有加密。但是，如果您配置了 TLS 加密，则 Dovecot 会自动对 LMTP 服务使用相同的设置。然后，SMTP 服务器可以使用 LMTPS 协议或 LMTP 上的 **STARTTLS** 命令连接它。

先决条件

- Dovecot 已安装。
- 如果要配置 LMTP 服务，TLS 加密会在 Dovecot 中配置。

流程

1. 验证 LMTP 协议是否已启用：

```
# doveconf -a | egrep "^protocols"
protocols = imap pop3 lmtp
```

   如果输出包含 **lmtp**，则该协议已启用。

2. 如果 **lmtp** 协议被禁用，请编辑 **/etc/dovecot/dovecot.conf** 文件，并将 **lmtp** 附加到 **protocols** 参数中的值：

```
protocols = ... lmtp
```

3. 根据您是否需要 LMTP 套接字或服务，在 **/etc/dovecot/conf.d/10-master.conf** 文件的 **service lmtp** 部分中进行以下更改：

   - LMTP 套接字：默认情况下，Dovecot 会自动创建 **/var/run/dovecot/lmtp** 套接字。
     可选：自定义所有权和权限：

```
service lmtp {
  ...
  unix_listener lmtp {
    mode = 0600
    user = postfix
    group = postfix
  }
  ...
}
```

- LMTP 服务：添加一个 **inet_listener** 子部分：

```
service lmtp {
  ...
  inet_listener lmtp {
    port = 24
  }
  ...
}
```

4. 配置 **firewalld** 规则，以只允许 SMTP 服务器访问 LMTP 端口，例如：

```
# firewall-cmd --permanent --zone=public --add-rich-rule="rule family="ipv4" source
address="192.0.2.1/32" port protocol="tcp" port="24" accept"
# firewall-cmd --permanent --zone=public --add-rich-rule="rule family="ipv6" source
address="2001:db8:2::1/128" port protocol="tcp" port="24" accept"
# firewall-cmd --reload
```

IPv4 的子网掩码 **/32** 和 IPv6 子网掩码 **/128** 限制对指定地址的访问。

5. 重新载入 Dovecot：

```
# systemctl reload dovecot
```

验证

1. 如果您配置了 LMTP 套接字，请验证 Dovecot 是否已创建套接字，以及权限是否正确：

```
# ls -l /var/run/dovecot/lmtp
srw-------. 1 postfix postfix 0 Nov 22 17:17 /var/run/dovecot/lmtp
```

2. 配置 SMTP 服务器，以使用 LMTP 套接字或服务向 Dovecot 提交电子邮件。
使用 LMTP 服务时，请确保 SMTP 服务器使用 LMTPS 协议或发送 **STARTTLS** 命令以使用加密连接。

其他资源

- **/usr/share/doc/dovecot/wiki/LMTP.txt**

1.7. 在 DOVECOT 中禁用 IMAP 或 POP3 服务

默认情况下，Dovecot 提供 IMAP 和 POP3 服务。如果您只需要其中之一，您可以禁用另一个以减少攻击面。

先决条件

- Dovecot 已安装。

流程

1. 取消 **/etc/dovecot/dovecot.conf** 文件中 **protocols** 参数的注释，并将它设置为使用所需的协议。例如，如果您不需要 POP3，请设置：

```
protocols = imap lmtp
```

默认情况下启用 **imap**、**pop3** 和 **lmtp** 协议。

2. 重新载入 Dovecot：

```
# systemctl reload dovecot
```

3. 关闭本地防火墙中不再需要的端口。例如，要关闭 POP3S 和 POP3 协议的端口，请输入：

```
# firewall-cmd --remove-service=pop3s --remove-service=pop3
# firewall-cmd --reload
```

验证

- 显示 **LISTEN** 模式下 **dovecot** 进程打开的所有端口：

```
# ss -tulp | grep dovecot
tcp  LISTEN 0  100  0.0.0.0:993  0.0.0.0:*  users:(("dovecot",pid=1405,fd=44))
tcp  LISTEN 0  100  0.0.0.0:143  0.0.0.0:*  users:(("dovecot",pid=1405,fd=42))
tcp  LISTEN 0  100     [::]:993     [::]:*  users:(("dovecot",pid=1405,fd=45))
tcp  LISTEN 0  100     [::]:143     [::]:*  users:(("dovecot",pid=1405,fd=43))
```

在本例中，Dovecot 仅侦听 TCP 端口 **993** (IMAPS)和 **143** (IMAP)。

请注意，如果您将服务配置为侦听端口而不使用套接字，则 Dovecot 仅为 LMTP 协议打开端口。

其他资源

- **firewall-cmd(1)** 手册页

1.8. 在 DOVECOT IMAP 服务器上使用 SIEVE 启用服务器端电子邮件过滤

您可以使用 ManageSieve 协议将 Sieve 脚本上传到服务器。Sieve 脚本定义服务器应对传入的电子邮件验证的规则和执行的操作。例如，用户可以使用 Sieve 转发特定发件人的电子邮件，管理员可以创建一个全局过滤器，将垃圾邮件过滤器标记的邮件移到单独的 IMAP 文件夹中。

ManageSieve 插件为 Dovecot IMAP 服务器添加了对 Sieve 脚本和 ManageSieve 协议的支持。

警告

仅使用支持通过 TLS 连接的 ManageSieve 协议的客户端。禁用此协议的 TLS 会导致客户端通过网络以纯文本形式发送凭证。

先决条件

- Dovecot 已配置，并提供 IMAP 邮箱。
- TLS 加密在 Dovecot 中已配置。
- 邮件客户端支持通过 TLS 连接的 ManageSieve 协议。

流程

1. 安装 **dovecot-pigeonhole** 软件包：

```
# dnf install dovecot-pigeonhole
```

2. 取消 **/etc/dovecot/conf.d/20-managesieve.conf** 中以下行的注释，以启用 **sieve** 协议：

```
protocols = $protocols sieve
```

除了已经启用的其他协议外，此设置还激活 Sieve。

3. 在 **firewalld** 中打开 ManageSieve 端口：

```
# firewall-cmd --permanent --add-service=managesieve
# firewall-cmd --reload
```

4. 重新载入 Dovecot：

```
# systemctl reload dovecot
```

验证

1. 使用客户端并上传 Sieve 脚本。使用以下连接设置：
   - 端口：4190
   - 连接安全：SSL/TLS
   - 身份验证方法： PLAIN
2. 向已上传 Sieve 脚本的用户发送电子邮件。如果电子邮件与脚本中的规则匹配，请验证服务器是否执行了定义的操作。

其他资源

- **/usr/share/doc/dovecot/wiki/Pigeonhole.Sieve.Plugins.IMAPSieve.txt**

- **/usr/share/doc/dovecot/wiki/Pigeonhole.Sieve.Troubleshooting.txt**
- **firewall-cmd(1)** 手册页

1.9. DOVECOT 如何处理配置文件

dovecot 软件包提供主配置文件 **/etc/dovecot/dovecot.conf** 和 **/etc/dovecot/conf.d/** 目录中的多个配置文件。Dovecot 会在您启动服务时组合文件来构建配置。

多个配置文件的主要优点是对设置进行分组并提高可读性。如果希望使用单个配置文件，您可以在 **/etc/dovecot/dovecot.conf** 中维护所有设置，并从该文件中删除所有 **include** 和 **include_try** 语句。

其他资源

- **/usr/share/doc/dovecot/wiki/ConfigFile.txt**
- **/usr/share/doc/dovecot/wiki/Variables.txt**

第 2 章 部署和配置 POSTFIX SMTP 服务器

作为系统管理员，您可以使用邮件传输代理(MTA)（如 Postfix）配置电子邮件基础架构，以使用 SMTP 协议在主机之间传输电子邮件消息。Postfix 是服务器端应用程序,用于路由和发送邮件。您可以使用 Postfix 建立本地邮件服务器、创建 null-client 邮件转发、使用 Postfix 服务器作为多个域的目的地，或者选择 LDAP 目录而不是用于文件的查找。

Postfix 的主要功能：

- 防止常见电子邮件相关威胁的安全功能
- 自定义选项，包括支持虚拟域和别名

2.1. 主 POSTFIX 配置文件概述

postfix 软件包在 **/etc/postfix/** 目录中提供多个配置文件。

要配置电子邮件基础架构，请使用以下配置文件：

- **main.cf** - 包含 Postfix 的全局配置。
- **master.cf** - 指定 Postfix 与各种进程的交互以完成邮件发送。
- **access** - 指定访问规则，如允许连接到 Postfix 的主机。
- **transport** - 将电子邮件地址映射到中继主机。
- **alias** - 包含邮件协议所需的可配置列表，其描述用户 ID 别名。请注意，您可以在 **/etc/** 目录中找到此文件。

2.2. 安装和配置 POSTFIX SMTP 服务器

您可以将 Postfix SMTP 服务器配置为接收、存储和发送电子邮件消息。如果在系统安装期间没有选择邮件服务器软件包，则 Postfix 默认不可用。执行以下步骤来安装 Postfix：

先决条件

- 您有 root 访问权限。
- 注册您的系统。
- 禁用和删除 Sendmail：

```
# dnf remove sendmail
```

流程

1. 安装 Postfix：

```
# dnf install postfix
```

2. 要配置 Postfix，请编辑 **/etc/postfix/main.cf** 文件并进行以下更改：

a. 默认情况下，Postfix 仅在 **loopback** 接口上接收电子邮件。要将 Postfix 配置为侦听特定的接口，请将 **inet_interfaces** 参数更新为这些接口的 IP 地址：

```
inet_interfaces = 127.0.0.1/32, [::1]/128, 192.0.2.1, [2001:db8:1::1]
```

要将 Postfix 配置为侦听所有接口，请设置：

```
inet_interfaces = all
```

b. 如果您希望 Postfix 使用与 **gethostname（）** 函数返回的完全限定域名(FQDN)不同的主机名，请添加 **myhostname** 参数：

```
myhostname = <smtp.example.com>
```

例如，Postfix 将此主机名添加到其处理的电子邮件的标头中。

c. 如果域名与 **myhostname** 参数中的不同，请添加 **mydomain** 参数：

```
mydomain = <example.com>
```

d. 添加 **myorigin** 参数，并将其设置为 **mydomain** 的值：

```
myorigin = $mydomain
```

使用这个设置时，Postfix 使用域名作为本地发布的邮件的源，而不是主机名。

e. 添加 **mynetworks** 参数，并定义允许发送邮件的可信网络的 IP 范围：

```
mynetworks = 127.0.0.1/32, [::1]/128, 192.0.2.1/24, [2001:db8:1::1]/64
```

如果来自不信任网络（如互联网）的客户端应该能够通过这个服务器发送邮件，则您必须在后续步骤中配置中继限制。

3. 验证 **main.cf** 文件中的 Postfix 配置是否正确：

```
$ postfix check
```

4. 启用 **postfix** 服务，以在引导时启动：

```
# systemctl enable --now postfix
```

5. 允许 smtp 流量通过防火墙，并重新载入防火墙规则：

```
# firewall-cmd --permanent --add-service smtp

# firewall-cmd --reload
```

验证

1. 验证 **postfix** 服务是否正在运行：

```
# systemctl status postfix
```

- 可选：如果输出处于停止、等待状态或服务没有运行，请重启 **postfix** 服务：

```
# systemctl restart postfix
```

- 可选：在更改 **/etc/postfix/** 目录中配置文件中的任何选项后，重新载入 **postfix** 服务，以应用这些更改：

```
# systemctl reload postfix
```

2. 验证系统上本地用户之间的电子邮件通信：

```
# echo "This is a test message" | mail -s <SUBJECT> <user@mydomain.com>
```

3. 通过从客户端(server1) 向邮件服务器(server2)外域外部的电子邮件地址发送电子邮件，验证您的邮件服务器是否是一个打开的中继：

a. 编辑 **server1** 的 **/etc/postfix/main.cf** 文件，如下所示：

```
relayhost = <ip_address_of_server2>
```

b. 编辑 **server2** 的 **/etc/postfix/main.cf** 文件，如下所示：

```
mynetworks = <ip_address_of_server2>
```

c. 在 **server1** 上，发送以下邮件：

```
# echo "This is an open relay test message" | mail -s <SUBJECT>
<user@example.com>
```

d. 检查 **/var/log/maillog** 文件：

```
554 Relay access denied - the server is not going to relay.
250 OK or similar - the server is going to relay.
```

故障排除

- 如果出现错误，请检查 **/var/log/maillog** 文件。

其他资源

- **/etc/postfix/main.cf** 配置文件
- **/usr/share/doc/postfix/README_FILES** 目录
- 使用和配置 firewalld

2.3. 自定义 POSTFIX 服务器的 TLS 设置

要使您的电子邮件流量加密，因此更加安全，您可以将 Postfix 配置为使用来自可信证书颁发机构(CA)的证书而不是自签名证书，并自定义传输层安全(TLS)安全设置。在 RHEL 9 中，TLS 加密协议默认在 Postfix 服务器中启用。基本的 Postfix TLS 配置包含用于入站 SMTP 的自签名证书，以及出站 SMTP 的机会 TLS。

先决条件

- 您有 root 访问权限。
- 您的服务器上已安装了 **postfix** 软件包。
- 您有由可信证书颁发机构(CA)签名的证书和私钥。
- 您已将以下文件复制到 Postfix 服务器：
  - 服务器证书：**/etc/pki/tls/certs/postfix.pem**
  - 私钥：**/etc/pki/tls/private/postfix.key**
- 如果服务器运行 RHEL 9.2 或更高版本，并且启用了 FIPS 模式，则客户端必须支持 Extended Master Secret(EMS)扩展或使用 TLS 1.3。没有 EMS 的 TLS 1.2 连接会失败。如需更多信息，请参阅 强制 TLS 扩展"Extended Master Secret" 知识库文章。

流程

1. 通过在 **/etc/postfix/main.cf** 文件中添加以下行，设置运行 Postfix 的服务器上的证书和私钥文件的路径：

```
smtpd_tls_cert_file = /etc/pki/tls/certs/postfix.pem
smtpd_tls_key_file = /etc/pki/tls/private/postfix.key
```

2. 通过编辑 **/etc/postfix/main.cf** 文件，仅将传入的 SMTP 连接限制到经过身份验证的用户：

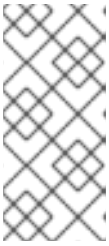
```
smtpd_tls_auth_only = yes
```

3. 重新载入 **postfix** 服务以应用更改：

```
# systemctl reload postfix
```

验证

- 将您的客户端配置为使用 TLS 加密，并发送电子邮件。



注意

要获取有关 Postfix 客户端 TLS 活动的其他信息，请通过更改 **/etc/postfix/main.cf** 中的以下行，将日志级别从 **0** 增加到 **1**：

```
smtp_tls_loglevel = 1
```

2.4. 将 POSTFIX 配置为将所有电子邮件转发到邮件中继

如果要将所有电子邮件转发到邮件中继，您可以将 Postfix 服务器配置为 null 客户端。在此配置中，Postfix 仅将邮件转发到不同的邮件服务器，且无法接收邮件。

先决条件

- 您有 root 访问权限。

- 您的服务器上已安装了 **postfix** 软件包。
- 您有要将电子邮件转发到的中继主机的 IP 地址或主机名。

流程

1. 要防止 Postfix 接受任何本地电子邮件发送并使其成为 null 客户端，请编辑 **/etc/postfix/main.cf** 文件，并进行以下更改：
   a. 通过将 **mydestination** 参数设置为空值，将 Postfix 配置为转发所有电子邮件：

   ```
   mydestination =
   ```

   在此配置中，Postfix 服务器不是任何电子邮件的目的地，并充当 null 客户端。
   b. 指定从您的 null 客户端接收电子邮件的邮件中继服务器：

   ```
   relayhost = <[ip_address_or_hostname]>
   ```

   中继主机是负责邮件发送的。将 ***<ip_address_or_hostname>*** 括在方括号中。
   c. 将 Postfix 邮件服务器配置为仅侦听要发送的电子邮件的回环接口：

   ```
   inet_interfaces = loopback-only
   ```

   d. 如果您希望 Postfix 将所有传出电子邮件的发件人域重写为中继邮件服务器的公司域，请设置：

   ```
   myorigin = <relay.example.com>
   ```

   e. 要禁用本地邮件发送，请在配置文件末尾添加以下指令：

   ```
   local_transport = error: local delivery disabled
   ```

   f. 添加 **mynetworks** 参数，以便 Postfix 将来自 127.0.0.0/8 IPv4 网络和 [::1]/128 IPv6 网络的本地系统的邮件转发到邮件中继服务器：

   ```
   mynetworks = 127.0.0.0/8, [::1]/128
   ```

2. 验证 **main.cf** 文件中的 Postfix 配置是否正确：

   ```
   $ postfix check
   ```

3. 重启 **postfix** 服务以应用更改：

   ```
   # systemctl restart postfix
   ```

验证

- 验证电子邮件通信是否被转发到邮件中继：

  ```
  # echo "This is a test message" | mail -s <SUBJECT> <user@example.com>
  ```

故障排除

- 如果出现错误，请检查 **/var/log/maillog** 文件。

其他资源

- **/etc/postfix/main.cf** 配置文件

2.5. 将 POSTFIX 配置为多个域的目的地

您可以将 Postfix 配置为可接收多个域的电子邮件的邮件服务器。在此配置中，Postfix 充当发送到指定域中地址的电子邮件的最终目的地。您可以配置以下内容：

- 设置指向同一电子邮件目的地的多个电子邮件地址
- 将多个域的传入电子邮件路由到同一 Postfix 服务器

先决条件

- 您有 root 访问权限。
- 您已配置了一个 Postfix 服务器。

流程

1. 在 **/etc/postfix/virtual** 虚拟别名文件中，指定每个域的电子邮件地址。在新行中添加每个电子邮件地址：

   ```
   <info@example.com>  <user22@example.net>
   <sales@example.com>  <user11@example.org>
   ```

   在这个示例中，Postfix 将发送到 info@example.com 的所有电子邮件重定向到 user22@example.net，并将发送到 sales@example.com 的电子邮件重定向到 user11@example.org。

2. 为虚拟别名映射创建一个哈希文件：

   ```
   # postmap /etc/postfix/virtual
   ```

   此命令创建 **/etc/postfix/virtual.db** 文件。请注意，在更新 **/etc/postfix/virtual** 文件后，您必须始终重新运行这个命令。

3. 在 Postfix **/etc/postfix/main.cf** 配置文件中，添加 **virtual_alias_maps** 参数，并将其指向哈希文件：

   ```
   virtual_alias_maps = hash:/etc/postfix/virtual
   ```

4. 重新载入 **postfix** 服务以应用更改：

   ```
   # systemctl reload postfix
   ```

验证

- 通过向其中一个虚拟电子邮件地址发送电子邮件来测试配置。

故障排除

- 如果出现错误，请检查 **/var/log/maillog** 文件。

2.6. 使用 LDAP 目录作为查找表

如果您使用轻量级目录访问协议(LDAP)服务器来存储帐户、域或别名，您可以将 Postfix 配置为使用 LDAP 服务器作为查找表。使用 LDAP 而不是文件进行查找可让您有一个中央数据库。

先决条件

- 您有 root 访问权限。
- 您的服务器上已安装了 **postfix** 软件包。
- 您有一个具有所需模式和用户凭证的 LDAP 服务器。
- 您已在运行 Postfix 的服务器上安装了 **postfix-ldap** 插件。

流程

1. 通过创建具有以下内容的 **/etc/postfix/ldap-aliases.cf** 文件来配置 LDAP 查找参数：
   a. 指定 LDAP 服务器的主机名：

   ```
   server_host = <ldap.example.com>
   ```

   b. 指定 LDAP 搜索的基础域名称：

   ```
   search_base = dc=<example>,dc=<com>
   ```

   c. 可选：根据您的要求自定义 LDAP 搜索过滤器和属性。搜索目录的过滤器默认为 **query_filter = mailacceptinggeneralid=%s**。
2. 通过添加以下内容，将 LDAP 源启用为 **/etc/postfix/main.cf** 配置文件中的查找表：

   ```
   virtual_alias_maps = ldap:/etc/postfix/ldap-aliases.cf
   ```

3. 运行 **postmap** 命令验证 LDAP 配置，它会检查任何语法错误或连接问题：

   ```
   # postmap -q @<example.com> ldap:/etc/postfix/ldap-aliases.cf
   ```

4. 重新载入 **postfix** 服务以应用更改：

   ```
   # systemctl reload postfix
   ```

验证

- 发送测试电子邮件，以验证 LDAP 查找是否正常工作。检查 **/var/log/maillog** 中的邮件日志是否存在任何错误。

其他资源

- **/usr/share/doc/postfix/README_FILES/LDAP_README** 文件
- **/usr/share/doc/postfix/README_FILES/DATABASE_README** 文件

2.7. 将 POSTFIX 配置为传出邮件服务器，以为经过身份验证的用户进行中继

您可以将 Postfix 配置为为经过身份验证的用户转发邮件。在这种情况下，您可以通过将 Postfix 配置为具有 SMTP 身份验证、TLS 加密和发件人地址限制的传出电邮件服务器，来允许用户进行自我验证，并使用其电子邮件地址通过 SMTP 服务器发送邮件。

先决条件

- 您有 root 访问权限。
- 您已配置了一个 Postfix 服务器。

流程

1. 要将 Postfix 配置为传出邮件服务器，请编辑 **/etc/postfix/main.cf** 文件，并添加以下内容：
   a. 启用 SMTP 身份验证：

```
smtpd_sasl_auth_enable = yes
broken_sasl_auth_clients = yes
```

   b. 禁用没有 TLS 的访问：

```
smtpd_tls_auth_only = yes
```

   c. 仅对经过身份验证的用户允许进行邮件转发：

```
smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated
defer_unauth_destination
```

   d. 可选：限制用户只使用自己的电子邮件地址作为发送者：

```
smtpd_sender_restrictions = reject_sender_login_mismatch
```

2. 重新载入 **postfix** 服务以应用更改：

```
# systemctl reload postfix
```

验证

- 在支持 TLS 和 SASL 的 SMTP 客户端上进行身份验证。发送一封测试电子邮件，以验证 SMTP 身份验证是否正常工作。

2.8. 从 POSTFIX 向运行在同一主机上的 DOVECOT 发送电子邮件

您可以使用 UNIX 套接字上的 LMTP，将 Postfix 配置为向同一主机上的 Dovecot 发送传入的邮件。这个套接字在本地机器上的 Postfix 和 Dovecot 之间启用直接通信。

先决条件

- 您有 root 访问权限。
- 您已配置了一个 Postfix 服务器。
- 您已配置了一个 Dovecot 服务器，请参阅 配置并维护 Dovecot IMAP 和 POP3 服务器 。
- 您已在 Dovecot 服务器上配置了 LMTP 套接字，请参阅 配置 LMTP 套接字和 LMTPS 侦听器。

流程

1. 将 Postfix 配置为使用 LMTP 协议和 UNIX 域套接字，将邮件发送到 **/etc/postfix/main.cf** 文件的 Dovecot ：
   - 如果要使用虚拟邮箱，请添加以下内容：

     ```
     virtual_transport = lmtp:unix:/var/run/dovecot/lmtp
     ```

   - 如果要使用非虚拟邮箱，请添加以下内容：

     ```
     mailbox_transport = lmtp:unix:/var/run/dovecot/lmtp
     ```

2. 重新载入 **postfix** 以应用更改：

   ```
   # systemctl reload postfix
   ```

验证

- 发送一封测试电子邮件，以验证 LMTP 套接字是否正常工作。检查 **/var/log/maillog** 中的邮件日志是否存在任何错误。

2.9. 将来自 POSTFIX 的电子邮件发送到运行在不同主机上的 DOVECOT

您可以在 Postfix 邮件服务器和 Dovecot 发送代理之间建立一个网络上的安全连接。为此，请将 LMTP 服务配置为使用网络套接字在邮件服务器之间传递邮件。默认情况下，LMTP 协议没有加密。但是，如果您配置了 TLS 加密，则 Dovecot 会自动对 LMTP 服务使用相同的设置。然后，SMTP 服务器可以使用 **STARTTLS** 命令，通过 LMTP 连接到它。

先决条件

- 您有 root 访问权限。
- 您已配置了一个 Postfix 服务器。
- 您已配置了一个 Dovecot 服务器，请参阅 配置并维护 Dovecot IMAP 和 POP3 服务器 。
- 您已在 Dovecot 服务器上配置了 LMTP 服务，请参阅配置 LMTP 套接字和 LMTPS 侦听器。

流程

1. 通过添加以下内容，将 Postfix 配置为使用 LMTP 协议和 INET 域套接字，将邮件发送到 **/etc/postfix/main.cf** 文件中的 Dovecot：

```
mailbox_transport = lmtp:inet:<dovecot_host>:<port>
```

将 ***<dovecot_host>*** 替换为 Dovecot 服务器的 IP 地址或主机名，将 **<port>** 替换为 LMTP 服务的端口号。

2. 重新载入 **postfix** 服务以应用更改：

```
# systemctl reload postfix
```

验证

- 将测试电子邮件发送到远程 Dovecot 服务器所在的地址，并检查 Dovecot 日志，以确保邮件是否成功发送。

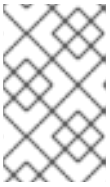
2.10. 保护 POSTFIX 服务

Postfix 是邮件传输代理(MTA)，其使用简单邮件传输协议(SMTP)在其他 MTA 之间发送电子邮件，并将电子邮件发送给客户端或传输代理。虽然 MTA 可以加密彼此之间的流量，但默认情况下可能不会这样做。您还可以通过将设置改为更安全的值来降低各种攻击的风险。

2.10.1. 减少 Postfix 网络相关的安全风险

要降低攻击者通过网络侵入系统的风险，请尽可能执行许多以下的任务。

- 不要在网络文件系统(NFS)共享卷上共享 **/var/spool/postfix/** 邮件假脱机目录。NFSv2 和 NFSv3 不维护对用户和组 ID 的控制。因此，如果两个或多个用户具有相同的 UID，他们可以接收和读取彼此的邮件，这是一个安全风险。



注意

此规则不适用于使用 Kerberos 的 NFSv4，因为 **SECRPC_GSS** 内核模块不使用基于 UID 的身份验证。但是，为了降低安全风险，您不应该将邮件假脱机目录放在 NFS 共享卷上。

- 为了减少 Postfix 服务器漏洞的可能性，邮件用户必须使用电子邮件程序访问 Postfix 服务器。不允许邮件服务器上的 shell 帐户，并将 **/etc/passwd** 文件中的所有用户 shell 设为 **/sbin/nologin** (**root** 用户可能例外)。

- 为了防止 Postfix 免受网络攻击，默认设置为仅侦听本地回环地址。您可以通过查看 **/etc/postfix/main.cf** 文件中的 **inet_interfaces = localhost** 行来验证这一点。这样可确保 Postfix 仅接受来自本地系统，而不是来自网络的邮件（如 **cron** 作业报告）。这是默认设置，保护 Postfix 免受网络攻击。要删除 localhost 限制并允许 Postfix 侦听所有接口，请将 **/etc/postfix/main.cf** 中的 **inet_interfaces** 参数设为 **all**。

2.10.2. 用于限制 DoS 攻击的 Postfix 配置选项

攻击者可以用流量淹没服务器，或发送触发崩溃的信息，从而导致拒绝服务(DoS)攻击。您可以通过在 **/etc/postfix/main.cf** 文件中设置限制来配置系统，以降低此类攻击的风险。您可以更改现有指令的值，或者您可以使用 *<directive> = <value>* 格式添加具有自定义值的新指令。

使用以下指令列表来限制 DoS 攻击：

smtpd_client_connection_rate_limit

这个指令限制了任何客户端每个时间单位内向这个服务进行连接尝试的最大数。默认值为 **0**，这意味着客户端每个时间单位内可以尝试的 Postfix 所能接受的连接数。默认情况下，指令排除可信网络中的客户端。

anvil_rate_time_unit

这个指令是计算速率限制的时间单位。默认值为 **60** 秒。

smtpd_client_event_limit_exceptions

这个指令排除了连接和速率限制命令中的客户端。默认情况下，指令排除可信网络中的客户端。

smtpd_client_message_rate_limit

此指令定义了每个时间单位客户端发送到请求的最大消息数（不论 Postfix 是否实际接收了这些消息）。

default_process_limit

此指令定义了提供给定服务的默认 Postfix 子进程的最大数。对于 **master.cf** 文件中的特定服务，您可以忽略此规则。默认情况下，该值为 **100**。

queue_minfree

此指令定义在队列文件系统中接收邮件所需的最小空闲空间量。该指令目前由 Postfix SMTP 服务器使用，以决定是否接受任何邮件。默认情况下，空闲空间量小于 **message_size_limit** 的 1.5 倍时，Postfix SMTP 服务器会拒绝 **MAIL FROM** 命令。要指定较高的最小空闲空间限制，请将 **queue_minfree** 值指定为至少 **message_size_limit** 的 1.5 倍。默认情况下，**queue_minfree** 值为 **0**。

header_size_limit

此指令定义用于存储消息头的最大内存量（以字节为单位）。如果消息头较大，它会丢弃超出的消息头。默认情况下，值为 **102400** 字节。

message_size_limit

此指令定义消息的最大大小（以字节为单位），包括信封信息。默认情况下，值为 **10240000** 字节。

2.10.3. 将 Postfix 配置为使用 SASL

Postfix 支持基于简单身份验证和安全层(SASL)的 SMTP 身份验证(AUTH)。SMTP AUTH 是简单邮件传输协议的扩展。目前，Postfix SMTP 服务器通过以下方式支持 SASL 实现：

Dovecot SASL

Postfix SMTP 服务器可以使用 UNIX-域套接字或 TCP 套接字与 Dovecot SASL 实现进行通信。如果 Postfix 和 Dovecot 应用程序运行在单独的计算机上，则使用此方法。

Cyrus SASL

启用后，SMTP 客户端必须使用服务器和客户端都支持和接受的身份验证方法与 SMTP 服务器进行身份验证。

先决条件

- **dovecot** 软件包已安装在系统上

流程

1. 设置 Dovecot：

   a. 在 **/etc/dovecot/conf.d/10-master.conf** 文件中包括以下行：

```
service auth {
  unix_listener /var/spool/postfix/private/auth {
```

```
    mode = 0660
    user = postfix
    group = postfix
  }
}
```

前面的示例对 Postfix 和 Dovecot 之间的通信使用 UNIX-域套接字。该示例还假定默认的 Postfix SMTP 服务器设置，其包括位于 **/var/spool/postfix/** 目录中的邮件队列，以及在 **postfix** 用户和组下运行的应用程序。

b. 可选：建立 Dovecot 以通过 TCP 侦听 Postfix 验证请求：

```
service auth {
  inet_listener {
    port = port-number
  }
}
```

c. 通过编辑 **/etc/dovecot/conf.d/10-auth.conf** 文件中的 **auth_mechanisms** 参数来指定电子邮件客户端用来使用 Dovecot 进行身份验证的方法：

```
auth_mechanisms = plain login
```

auth_mechanisms 参数支持不同的纯文本和非纯文本身份验证方法。

2. 通过修改 **/etc/postfix/main.cf** 文件来建立 Postfix：

a. 在 Postfix SMTP 服务器上启用 SMTP 身份验证：

```
smtpd_sasl_auth_enable = yes
```

b. 为 SMTP 身份验证启用 Dovecot SASL 实现：

```
smtpd_sasl_type = dovecot
```

c. 提供相对于 Postfix 队列目录的身份验证路径。请注意，使用相对路径可确保无论 Postfix 服务器是否以 **chroot** 运行，配置都可以正常工作：

```
smtpd_sasl_path = private/auth
```

此步骤使用 UNIX-域套接字在 Postfix 和 Dovecot 之间进行通信。

如果您使用 TCP 套接字进行通信，要将 Postfix 配置为在不同机器上查找 Dovecot，请使用类似如下的配置值：

```
smtpd_sasl_path = inet: ip-address : port-number
```

在上例中，将 *ip-address* 替换为 Dovecot 机器的 IP 地址，将 *port-number* 替换为 Dovecot 的 **/etc/dovecot/conf.d/10-master.conf** 文件中指定的端口号。

d. 指定 Postfix SMTP 服务器为客户端提供的 SASL 机制。请注意，您可以为加密和未加密的会话指定不同的机制。

```
smtpd_sasl_security_options = noanonymous, noplaintext
smtpd_sasl_tls_security_options = noanonymous
```

以上指令指定在未加密的会话期间，不允许匿名身份验证，且不会允许传输未加密的用户名或密码的机制。对于使用 TLS 的加密会话，只允许非匿名身份验证机制。

其他资源

- Postfix SMTP 服务器策略 - SASL 机制属性
- Postfix 和 Dovecot SASL
- 在 Postfix SMTP 服务器中配置 SASL 身份验证