はじめに

検証用に SMTP 認証ができるメールサーバーを立てる必要があった。Exim4 だと文献が少ない、どうやって NTLM 認証を設定するのかわからない、という厳しさがあり、Postfix + Dovecot で構築できるようにメモを残す次第。
検証用途なので、セキュリティやユーザー管理はまったく考慮していない。
とにかく動く、というレベル。

Dovecot を使うメリットとして、別途 POP/IMAP サーバーを入れなくてもよい（Cyrus-SASL を使うと、Postfix + Cyrus-SASL + POP/IMAP サーバーという構成になる）という点がある。
副作用としては、SMTP 認証と POP3 認証の方式を分けられない、という点がある。認証方法の指定は 1 箇所でしかできない。指定した認証方法が、SMTP 認証にも POP3 認証にも適用される点を認識しておくこと。
今回は POP3 を使う。

Squeeze で確認。
AUTH PLAIN、AUTH LOGIN、CRAM-MD5、NTLM。

Postfix

インストール

apt を使うと、Postfix をインストールする際に、既存の Exim4 をきれいにアンインストールしてくれる。
これは便利。

```
# apt-get install postfix
```

インストール中に、Postfix の設定を聞いてくる。

- 「メール設定の一般形式」は、" インターネットサイト " を選択（検証で、実稼動しているメールサーバーにメールを投げる必要があるため）。
- 「システムメール名」は、自サーバー宛メールだと判断するドメイン名を指定する。

/etc/postfix/main.cf の設定

上記のとおりにインストールすれば、インストール直後の状態で、認証なしでメールを送信できるようになっているはず。
自サーバー宛メールは、mbox 形式で /var/spool/mail 以下に溜まるようになっている。

main.cf に、Dovecot を使った SMTP 認証に必要な設定のみ追加する。全部手打ちなので、タイプミスに注意。。。

```
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_sasl_local_domain = $mydomain
smtpd_recipient_restrictions = permit_sasl_authenticated, reject
```

- smtpd_sasl_path は、Dovecot と認証について通信する UNIX ソケットファイルを指定する。特に理由がない限り、上記のよくある設定で固定するのがいいと思う。相対パスで指定すれば、Postfix が chroot していても大丈夫とのことなので、上記例ではそのようにしている。Squeeze のパッケージ版では、private ディレクトリの位置は /var/spool/postfix/private。
- smtpd_sasl_local_domain は認証の realm 名。何でもいいのだろうが、一般的な設定例にしたがっておく。
- smtpd_recipient_restrictions の permit_sasl_authenticated で SASL 認証を通ったユーザーを許可、reject でそれ以外のユーザーを拒否。依然として mynetworks からは認証なしでメール送信可能。
- 上記設定例には入れていないが、broken_sasl_auth_clients = yes としておくと、RFC にしたがっていないメールクライアントの動作にも対応する、とのこと。Outlook Express が該当するらしい。

Postfix に設定を反映させる。

```
# /etc/init.d/postfix reload
```

まだ Dovecot が存在していないので、この段階では Postfix は mynetworks 以外からメールをまったく送信できない状態になっている（mynetworks 以外からのメールは認証を通らないと送信できなくなっているが、認証サーバーである Dovecot がいないので、すべての認証が失敗してしまうという状態）。

Dovecot

インストール

```
# apt-get install dovecot-pop3d
```

デフォルトで、認証には PAM を使うようになっており、SMTP 認証のメカニズムは plain のみとなっている（mechanisms = plain）。

POP3 サーバーの設定

/etc/dovecot/dovecot.conf に以下の 2 つのパラメーターの設定を行う。

■ SSL/TLS を使わない接続で平文パスワードを許可

AUTH PLAIN、AUTH LOGIN を使わなければ必要ないが。。。

```
disable_plaintext_auth = no
```

■ メールボックスの位置を指定

Postfix の設定に合わせる。

```
mail_location = /var/spool/mail/%u
```

上記 2 つのパラメータ設定を反映する。

```
# /etc/init.d/dovecot reload
```

これで、自サーバー宛に溜まったメールを POP3 で取ってこれるようになっている。繰り返しになるが、デフォルトで POP3 認証方式は AUTH PLAIN の PAM 認証。

reload を実行した後は、Dovecot が正常に起動していることを確認すること。
設定ファイルに何らかの不備がある場合には、Dovecot が起動しない場合もある。Dovecot の動作に何らかの問題があると考えられる場合には、エラーログを確認すること。ログの出力先のデフォルトは dovecot.conf 内で syslog_facility = mail と設定されている。

Postfix 経由で SMTP 認証を行う設定

/etc/dovecot/dovecot.conf の socket listen セクションを設定する。
コメントアウトされているので、socket listen セクションと client サブセクションのコメントをはずして流用する。

```
socket listen {
    # master {
        # master サブセクションはコメントをはずす必要なし。
    # }
    client {
        path = /var/spool/postfix/private/auth
        mode = 0660
        user = postfix
        group = postfix
    }
}
```

client サブセクションの path パラメーターで、Postfix の main.cf で指定した、認証通信用の UNIX ソケットファイルのパス（smtpd_sasl_path の値）を指定する。こちらはフルパスで指定しないといけないので、正しいフルパスを把握しておく必要がある。
mode、user、group の各パラメータ値は上記のとおりで固定。

上記 dovecot.conf 設定を反映する。

```
# /etc/init.d/dovecot reload
```

この時点で、Postfix から AUTH PLAIN でのメール送信が可能になっている。認証は PAM が使われる。

認証パスワードファイルの作成と設定

Dovecot デフォルトの PAM 認証は、AUTH PLAIN でしか使えないようだ。それ以外の認証メカニズムを指定すると、Dovecot が起動しない（エラーログには、そのメカニズムは指定された passdbs ではサポートされていない、というログが出る）。
なので、Dovecot がパスワードファイルでメールユーザーを認証するようにし、PAM 認証は無効にする。

■ パスワードファイル作成

よくある設定例にしたがい、パスワードファイルを /etc/dovecot/passwd として作成することにする。当然だが、Dovecot が /etc/dovecot/passwd を読み取れる権限設定になっていなくてはならない。

htpasswd のようにいきなりパスワードファイルを作成してくれるコマンドは存在しない。

まずパスワードファイルで使用するパスワード文字列を dovecotpw コマンドで作成し、それをユーザー名とともに、パスワードファイルに記載する必要がある。

```
# dovecotpw -s CRAM-MD5 -p < パスワード >
```

パスワード文字列が標準出力に出力される。
-s オプションに指定できる暗号化方式は、dovecotpw -l で表示される。もしくは man dovecotpw。

パスワードファイルには 1 行ごとに、

```
< ユーザー名 >:< パスワード文字列 >
```

と記載する（/etc/passwd の最初の 2 フィールドと同じ形式）。

パスワード文字列の先頭は、各暗号化方式をあらわす文字になっている（例：{CRAM-MD5}、暗号化なし認証メカニズム [PLAIN と LOGIN] は {PLAIN} で共通）。
同じユーザーが複数の暗号化方式を使用する場合、暗号化方式の数だけ、パスワードファイルに行を追加する。
以下にパスワードファイルの記載例を示す。

```
user1:{CRAM-MD5}9186d855e11eba527a7a52ca82b313e180d62234f0acc9051b527243d41e2740
user1:{NTLM}8846f7eaee8fb117ad06bdd830b7586c
user1:{PLAIN}password
```

上記例で、user1 は CRAM-MD5、NTLM、PLAIN、LOGIN が利用できる。

■ Dovecot がパスワードファイルで認証を行うよう設定する

/etc/dovecot/dovecot.conf の auth default セクションで設定する。
まず、passdb pam サブセクションをコメントアウトする。

```
auth default {
    # passdb pam {
    #    ...
    # }
}
```

次に、passdb passwd-file サブセクションの args パラメーターで、作成したパスワードファイルを指定する。passdb passwd-file サブセクションは、新たに追加しても、既存の設定例のコメントをはずして流用してもよい。

```
auth default {

    # passdb pam {
    #    ...
    # }

    passdb passwd-file {
        args = /etc/dovecot/passwd
    }
}
```

Dovecot にこの設定を反映するためには、

```
# /etc/init.d/dovecot reload
```

を実行する。
ここまでで、パスワードファイルによる AUTH PLAIN 認証が可能になっている。

※ Dovecot は、認証が行われるたびにパスワードファイルを参照するので、パスワードファイルの中身を変更しただけの場合は、設定の reload は必要ない。reload は dovecot.conf の設定変更の場合に必要。

認証メカニズムの設定と変更

Dovecot で各認証メカニズムを使用するためには、

- パスワードファイル（ここまでの手順では /etc/dovecot/passwd として作成したファイル）に、その認証メカニズムのパスワード文字列がユーザー名とともに登録されていること
- dovecot.conf の auth default セクションの mechanisms パラメーターにその認証メカニズムが指定されていること

の 2 つが必要になる。
dovecot.conf のデフォルトでは、mechanisms パラメーターには AUTH PLAIN しか指定されていない（mechanisms = plain）ので、それ以外の認証メカニズムを使いたい場合には、mechanisms パラメーターの設定変更が必要になる。

最初にも書いたが、Dovecot の認証メカニズム設定は、SMTP 認証にも POP3 認証にも共通で使われる点に留意すること。たとえば認証メカニズムを CRAM-MD5 のみにすると、POP3 も CRAM-MD5 認証になる。メールソフト側で CRAM-MD5 の設定をしないとメールが取れなくなるので注意。

以下、auth default セクションの mechanisms パラメーターの指定例。
まず CRAM-MD5 のみ使用。

```
auth default {

    mechanisms = cram-md5

    # passdb pam {
    #    ...
    # }

    passdb passwd-file {
        args = /etc/dovecot/passwd
    }
}
```

認証メカニズムは、スペースで区切って複数指定可能。

```
auth default {

    mechanisms = plain login cram-md5 ntlm

    # passdb pam {
    #    ...
    # }

    passdb passwd-file {
```

```
        args = /etc/dovecot/passwd
    }
}
```

認証メカニズム設定を変更した後は、dovecot.conf を保存した後、

```
# /etc/init.d/dovecot reload
```

とすれば有効になる。