Postfix

Ogólne

Postfix, w czystej postaci, jest serwerem SMTP (domyślnie bez autoryzacji SMTP) i nie posiada sam w sobie funkcji obsługi POP3 (można doinstalować jakiś serwer POP3, np. *Courier-imap, Dovecot*). W systemie może być zainstalowany tylko jeden serwer poczty (*Postfix* koliduje np. z *ssmtp*), zwany MTA (*Mail Transfer Agent*). Pomoc na temat konfiguracji:

man 5 postconf

Do prawidłowego działania potrzebna jest odpowiednia konfiguracja systemu:

- */etc/hosts*
- */etc/conf.d/hostname*
- **/etc/init.d/hostname start** (a najlepiej **rc-update add hostname default**)

Rodzaje skrzynek pocztowych:

- **MailDir** - format *Qmail* (ale także *Postfix*, *Exim*); każdy list w osobnym pliku, zawiera w sobie podkatalogii: new, cur, tmp; równoczesny dostęp do pliku przez wiele aplikacji;
- **Mbox** - wszystkie wiadomości są przechowywane w jednym pliku (*/var/mail/$user* lub */var/spool/mail/$user*); tylko jedna aplikacja może korzystać z listu w tym samym czasie; nigdy nie był zaakceptowany przez RFC;

Wirtualne aliasy lokalne

Wirtualne adresy poczty przychodzącej ---> */etc/mail/aliases*, np.:

```
rob:          robert
biuro:        robert
adminek:      root
postmaster:   root
# Listy do „batona” lądują w niebycie:
baton:        /dev/null
```

Aliasy uaktywnia wpis w */etc/postfix/main.cf*:

alias_maps = hash:/etc/mail/aliases

alias_database = hash:/etc/mail/aliases

Aby sprawdzić, jaki typ aliasów jest aktualnie obsługiwany przez *Postfiksa* (wpis *„hash”* oznacza generowanie zahaszowanych plików z końcówką *.db):

postconf -m

Wprowadzone zmiany należy zatwierdzić poleceniem:

newaliases

a jeśli chcemy utworzyć bazę aliasów z innego pliku niż domyślny, używamy:

postalias /jakiśFolder/aliases

Uwaga: bez skonfigurowania tej sekcji, *Postfix* nie będzie działał prawidłowo. Należy dołożyć starań, aby w tym pliku nie było wpisu żadnego nieistniejącego w systemie konta, w przeciwnym wypadku nasz *Postfix* może zostać wykorzystany jako *pseudo open relay* (ktoś

wysyła na nasz serwer list z fałszywym adresem zwrotnym, nasz serwer odsyła ten list z adnotacją, iż nie ma takiego użytkownika → odsyła go jednak na ten fałszywy (dowolny) adres wraz z treścią listu, co może służyć do rozsyłania spamu).

Wirtualne aliasy domen

W pliku */etc/postfix/virtual:*

ro@asus-i7.imagine *robert* (listy adresowane do „*ro@asus-i7.imagine*” trafią do lokalnego użytkownika „*robert*”)

biuro@asus-i7.imagine *mój-realny-email@op.pl* (w tym przypadku wystąpi problem z odesłaniem listu przez odbiorcę, patrz następny rozdział)

@asus-i7.imagine *robert* (wszystkie inne wysyłane są do użytkownika „*robert*”)

W pliku */etc/postfix/main.cf:*

virtual_alias_maps = hash:/etc/postfix/virtual

Zatwierdzamy zmiany:

 # postmap /etc/postfix/virtual

 # /etc/init.d/postfix reload

Problem braku publicznej domeny

Mając do dyspozycji tylko prywatną sieć z prywatnymi domenami możemy rozsyłać pocztę bez problemu (w pliku */etc/hosts* muszą być wpisane definicje wszystkich lokalnych maszyn w sieci, a one same muszą mieć stałe IP). Użytkownicy mogą także odpowiadać na listy odsyłając je do lokalnych skrzynek pocztowych danego komputera. Problem pojawia się, gdy chcemy wysłać list do jakiegoś użytkownika na zewnątrz, do sieci publicznej. W większości przypadków zostanie on zablokowany przez publiczny serwer POP3, np. w oparciu o rekord SPF (*Sender Policy Framework*) stanowiący ochronę antyspammową (w imieniu danej domeny, pocztę może wysyłać tylko określony serwer SMTP, co zapobiega wysyłaniu listów ze zmienionym adresem zwrotnym; lub np. *gmail.com,* który nie przyjmuje listów bez autoryzacji lub nie mających pełnej publicznej nazwy domenowej: „*Our system has detected that this message is 550-5.7.1 not RFC 5322 compliant: 550-5.7.1 'From' header has non compliant domain name. 550-5.7.1 To reduce the amount of spam sent to Gmail, this message has been 550-5.7.1 blocked”*). Wysyłka takich listów może zostać również zablokowana przez naszego dostawcę Internetu (w tym przypadku, wysłane listy pozostaną w kolejce → sprawdź wynik polecenia „*mailq*” → adnotacja „*Connection timed out*”). A nawet jeśli list dotrze do odbiorcy, nie będzie on mógł na niego odpowiedzieć, gdyż list posiada nieroutowalny adres zwrotny z prywatną lokalną domeną. Częściowym rozwiązaniem tego problemu jest utworzenie mapy adresów zwrotnych. W pliku */etc/postfix/main.cf* umieszczamy odwołanie do takiej mapy:

smtp_generic_maps = hash:/etc/postfix/generic

W utworzonym pliku */etc/postfix/generic* umieszczamy wpisy (adres nadawcy po lewej stronie będzie zastępowany adresem znajdującym się po stronie prawej):

ro@asus-i3.3bird *robertsurma@op.pl*

biuro@asus-i3.3bird *biuro@op.pl*

 # @asus-i3.3bird *biuro@op.pl*

Następnie budujemy zahaszowaną bazę mapy:

 # postmap /etc/postfix/generic

 # postfix reload

Uwaga: Wykonanie tych czynności nie gwarantuje dotarcie poczty do adresatów (listy nadal mogą być blokowane przez serwer odbiorcy lub dostawcy Internetu), gwarantuje tylko

możliwość odpowiedzi na nasz list. Należy także pamiętać o otwarciu portu dla postfiksa, gdyż serwer nie tylko wysyła e-maile, ale także otrzymuje odpowiedzi i zapytania z innych serwerów.

W przypadku posiadania publicznej domeny, należy pamiętać o skonfigurowaniu rekordu MX!

Czy Postfix nasłuchuje na porcie 25?

nmap nasze_IP (jeśli nie wyświetli otwartych portów, należy wyłączyć firewalla lub zmienić jego ustawienia)

telnet nazwaNaszegoHosta.3bird 25 (sprawdzamy czy nasłuchuje; klient *telnet* znajduje się w pakiecie „*telnet-bsd*")

ehlo mojaDomena.com (przedstawiamy się podając naszą domenę; jeśli mamy włączoną autoryzację SASL, zobaczymy wyrażenie *AUTH*; jeśli mamy działające TLS, zobaczymy wyrażenie STARTTLS; wychodzimy poprzez komendę *quit*)

Pożyteczne polecenia

Wykaz wszystkich efektywnych opcji (tzn. także tych domyślnych, nie określonych w pliku konfiguracyjnym):

postconf -n

Wysłanie kolejki zatrzymanej z jakiegoś powodu (np. w wyniku braku połączenia):

postfix flush

Konserwacja *Postfix* (sprawdzenie poprawności całej struktury serwera oraz usunięcie niepotrzebnych plików tymczasowych):

postsuper -s -p

Podgląd kolejki (wykaz skolejkowanych listów w */var/spool/postfix*):

mailq

Usuwanie pojedynczego listu z kolejki:

postsuper -d ID-listu

Usuwanie całej kolejki:

postsuper -d ALL

Sprawdzanie logów:

cat /var/log/mail/current | more

Autoryzacja SASL

Aby możliwa była autoryzacja wysyłanej poczty poprzez *SASL* (*Simple Authentication and Security Layer*), czyli z użyciem loginu i hasła (pożyteczne w przypadku mobilnych użytkowników, których blokuje opcja *$mynetworks*, gdyż zmieniają ciągle swoje IP) należy doinstalować pakiet *Cyrus SASL*. W momencie dodania obsługi SASL, zmienna *$mynetworks* przestaje mieć znaczenie (ale nie trzeba jej usuwać czy deaktywować, nadal będzie obowiązywała klientów pocztowych, którzy chcą wysłać pocztę bez autoryzacji).

Istnieją co najmniej trzy metody korzystania z SASL: z użyciem nazw użytkowników systemowych / haseł systemowych (czyli wykorzystania *pam* lub *shadow*) lub z użyciem nazw i haseł nie związanych z kontami systemowymi (baza *sasldb2* lub *mysql*). W pierwszym przypadku będziemy korzystali z demona *saslauthd*, w drugim odwołamy się bezpośrednio do bazy *sasldb2*, a w trzecim do bazy *mysql*. Według niektórych, korzystanie z pierwszej metody jest lepsze, gdyż demon działa na prawach roota (czyli ma np. dostęp do */etc/shadow* i innych danych), podczas gdy klienci mają minimalne prawa i nie mają bezpośredniego dostępu do plików z hasłami systemowymi. Inni mówią, że plik *sasldb2* jest dobrze zabezpieczony:

ls -al /etc/postfix/saslpass

-rw------- root root /etc/postfix/saslpass

ls -al /etc/sasl2/sasldb2

-rw------- postfix root /etc/sasl2/sasldb2

Metoda 1

W pliku konfiguracyjnym SASL, czyli w */etc/sasl2/smtpd.conf* umieszczamy:

pwcheck_method: saslauthd

mech_list: PLAIN LOGIN CRAM-MD5 DIGEST-MD5 (dane w formacie *plain-text* lub zaszyfrowane; twórcy Postfiksa zalecają jednak użycie tylko PLAIN LOGIN przy metodzie *saslauthd*, gdyż inaczej zakończy się wszystko błędem)

W pliku */etc/conf.d/saslauthd* określamy źródło nazw i haseł:

SASLAUTHD_OPTS="-a shadow" (informacje o możliwych opcjach autoryzacji uzyskamy wydając polecenie „*saslauthd -v*”, zazwyczaj będzie to: *pam*, *rimap* (czyli "*Remote IMAP server*"), *shadow* (używa lokalnych haseł użytkowników), a nawet *sasldb* (czyli odwołanie do bazy *sasldb*, choć sam podręcznik mówi, że to prawdopodobnie nie jest to, czego naprawdę chcesz, że nawet domyślnie podczas kompilacji ta opcja jest wyłączona, że zamiast tego należy zastosować inną metodę: "*pwcheck_method: auxprop*")

Następnie uruchamiany demona:

/etc/init.d/saslauthd start (a najlepiej: **rc-update add saslauthd default**)

Testujemy poprawność działania mechanizmu:

testsaslauthd -u *użytkownikSystemowy* -p *hasłoUżytkownikaSystemowego* (gdy używamy *pam*, należy tu dodać jeszcze opcję *-s smtp*; sama komenda pochodzi z pakietu *cyrus-imapd*)

0: OK "*Success.*"

Metoda 2

Plik konfiguracyjny SASL to */etc/sasl2/smtpd.conf*:

pwcheck_method:auxprop (korzystamy bezpośrednio z bazy *sasldb2*, więc zastosujemy metodę *auxprop*; w tym przypadku nie uruchamiamy demona *saslauthd*)

mech_list: PLAIN LOGIN CRAM-MD5 DIGEST-MD5 (pod warunkiem, że nasz system i klient pocztowy obsługuje ten typ szyfrowania; jeśli wykorzystujemy *MD5, niepotrzebne nam TLS/SSL, choć nie wyklucza się to z TLS)

Tworzymy bazę danych użytkowników (loginy pocztowe i hasła pocztowe), które będą zapisane jako zwykły tekst w pliku */etc/sasl2/sasldb2*:

saslpasswd2 -c -u *server.3bird loginUżytkownika*

Password: *(naszeHasło; w moim przypadku będzie to publiczne1)*

Uwaga: Polecenie to utworzy użytkownika o loginie „*loginUżytkownika@server.3bird*”.

Jeśli chcemy usunąć istniejącego w bazie użytkownika:

saslpasswd2 -d *loginUżytkownika@server.3bird*

Nadajemy tej bazie odpowiednie prawa dla użytkownika *postfix*:

ls -all /etc/sasl2/

chown postfix /etc/sasl2/sasldb2

ls -all /etc/sasl2

Sprawdzamy zawartość bazy SASL:

sasldblistusers2

loginUżytkownika@server.3bird: userPassword (hasło jest niewidoczne)

Testujemy poprawność działania mechanizmu w trybie tekstowym:

smtptest -m plain -a robert server.3bird (polecenie zawarte jest w pakiecie *„cyrus-imap”*; możliwe jest także użycie *„-m login”*, *„-m cram-md5”*, *„-m digest-md5”*; zaś przy metodzie *„pwcheck”* i *„saslauthd”* [co trochę bez sensu] - tylko *„-m cram-md5”*, *„-m digest-md5”*)

S: 220 asus-i3 ESMTP Postfix
C: EHLO smtptest
S: 250-asus-i3
S: 250-PIPELINING
S: 250-SIZE 1000000
S: 250-VRFY
S: 250-ETRN
S: 250-STARTTLS
S: 250-AUTH PLAIN LOGIN CRAM-MD5 DIGEST-MD5
S: 250-AUTH=PLAIN LOGIN CRAM-MD5 DIGEST-MD5
S: 250-ENHANCEDSTATUSCODES
S: 250-8BITMIME
S: 250-DSN
S: 250 SMTPUTF8
Please enter your password: (tu wpisujemy hasło z naszej bazy salsdb2)
C: AUTH PLAIN aHJvYmVydsBwcm1hq2Vkf2qx
S: 235 2.7.0 Authentication successful
Authenticated.
Security strength factor: 0

Uwaga: W przypadku wystąpienia błędu *„S: 503 5.5.1 Error: authentication not enabled. Authentication failed. generic failure”* (nie pojawi się wtedy linia *„S: 250-AUTH PLAIN LOGIN CRAM-MD5 DIGEST-MD5”*) należy w pliku */etc/postfix/main.cf* zakomentować opcję *„smtpd_tls_auth_only = yes”*. Błąd wynika z faktu, że TLS nie jest jeszcze prawidłowo skonfigurowane.

Testujemy poprawność działania w kliencie ***Sylpheed*** (klient zwany jest zwykle MUA, czyli *Mail User Agent*):

Adres e-mail: *robert@nazwaHosta.3bird (możliwa wersja skrócona „robert”, wtedy doklejona zostanie domena zdefiniowana w $myorigin)*

Serwer dla odbioru: *nazwaHosta.3bird (na razie i tak bez znaczenia, bo nie mamy serwera POP3, więc zgłosi błąd)*

Serwer SMTP (do wysyłania): *nazwaHosta.3bird*

Identyfikator użytkownika: *robert@nazwaHosta.3bird (możliwa wersja skrócona „robert”)*

Hasło: *(hasło utworzone w bazie „sasldb2”)*

Zakładka „Wyślij” →

Uwierzytelnianie SMTP (SMTP AUTH): *robert@nazwaHosta.3bird (działa przy możliwych 4 ustawieniach: opcja w ogóle nie aktywna, PLAIN, LOGIN, CRAM-MD5)*

Zakładka „SSL” →

[x] Wysyłanie (SMTP): *Nie korzystanie z SSL*

[x] Używanie nieblokującego SSL: *(działa także bez aktywacji tej opcji)*

Zakładka „Zaawansowane” →

[x] Określ port SMTP: 25 *(działa także bez określania, gdyż domyślnie używa portu 25)*

Uwaga: Listy można wysyłać na adres „robert”, „ro” (alias lokalny), także na „robert@nazwaHosta.3bird”. Wysyłka na adresy publiczne (Gmail, Onet, 3bird) nie będzie działać.

Metoda 3

Możemy także pobierać hasła z bazy *mysql*. W tym celu w pliku */etc/sasl2/smtpd.conf*:

```
pwcheck_method: auxprop
auxprop_plugin: sql
mech_list: PLAIN LOGIN CRAM-MD5 DIGEST-MD5
sql_engine: mysql
sql_hostnames: 127.0.0.1, 192.169.0.2
sql_user: nazwaUżytkownika
sql_passwd: mojeHasłoDoBazy
sql_database: nazwaBazyDanych
sql_select: SELECT haslo FROM uzytkownik WHERE uzytkownik = '%u@%r'
```

Zabezpieczamy nasz plik przed odczytem:

chmod 600 /etc/sasl2/smtpd.conf

TLS

Specyfikacja mechanizmu opisana jest w standardzie RFC 4409.

Uwaga: Podręcznik podaje, że do konfiguracji obsługi TLS należy użyć poleceń:

/usr/sbin/postfix tls enable-server (dla serwera)

/usr/sbin/postfix tls enable-client (dla klienta)

Błędy i tipsy

- Przy wysyłaniu maila w trybie tekstowym, w logach */var/log/mail/current* występuje komunikat: „*Mail loops back to myself*”. Rozwiązanie: brak wpisu naszej domeny do *$mydestination*.
- Podczas restartu *Postfiksa*, występuje błąd. Należy wydać komendę „*postconf*”, która prawdopodobnie wyrzuci komunikat i wskaże miejsce błędu: „*postconf: fatal: file /etc/postfix/master.cf: line 19: bad field count*”.

Ostatnia aktualizacja: 10 sierpnia 2018.