

Модуль 8. Сервер электронной почты

Входящая почта (Inbox) в каталог /var/mail/userX

Архив почты в /mbox.

Пока ставим ПО УМОЛЧАНИЮ!!!

```
server# apt-get install mailutils
```

Проверяем:

```
server# mail user1
```

Cc –

Subj test1maik

Hello, User1

Ctrl + D

--

```
#ls /var/mail
```

```
#cat /var/mail/user1 --читаем почту
```

Также почитать можно войдя вторым сеансом под user1 на

Server и набрать mail.

Далее, 1 – открыть письмо, r – ответить (Ctrl+D), q- выйти

После прочтения (q) письмо сохраняется в mbox

можно открыть

```
#cat mbox
```

```
echo test | mail -s test user1@server.corpX.un
```

На Gate:

```
# nslookup -q=MX corpX.un
```

```
# nslookup -q=A corpX.un
```

```
# tail -f -n0 /var/log/mail.log
```

```
# telnet 192.168.X.10 25
```

```
telnet 192.168.X.10 25
```

```
HELO gate.isp.un
```

```
250 server.corpX.un
```

```
MAIL FROM: root@gate.isp.un
```

```
RCPT TO: user1@corpX.un
```

454 4.7.1 <user1@corpX.un>: Relay access denied

Из лога видно что postfix не обрабатывает указанный домен:

postfix/smtpd[7899]: NOQUEUE: reject:

Настроим на использование почтового домена corpX.un для обработки почты.

Настройка МТА на обработку почты домена corpX.un

Настройка МТА postfix (Ubuntu)

```
root@server:~# cat /etc/postfix/main.cf
```

...

```
mydestination = server.corpX.un, localhost.corpX.un, , localhost, corpX.un
```

, , - позволяет использовать простое имя пользователя для обработки почты (mail user1)

...

```
root@server:~# postconf
```

```
root@server:~# service postfix check
```

```
root@server:~# service postfix reload
```

Зайти под user1 и стереть (прочитать) почту

Использование почтовых псевдонимов

Механизм алиасов позволяет почту приходящую на определенный ящик перенаправить на другой адрес (например, сообщения с серверов root'у на адрес администратора).

Пример:

Псевдонимы настраиваются в:

```
# vi /etc/aliases
```

Напр. почта на псевдоним support представляет собой групповой адрес:

Добавляем в /etc/aliases:

```
support: user1, user2
```

Для вступления в силу алиасов набрать команду:

```
# newaliases
```

```
# mail root
```

Письмо на root (root@server.corpX.un) доставляется на user1@corpX.un

Проверяем, заходим под user2 пишем на root, читаем под user1.

#####

Использование виртуальных почтовых доменов

Для сценария:

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

на Server добавим учетную запись user2 для партнерской организации:

```
# useradd user2 -m -s /bin/bash
```

на isp уже имеются файлы мастер зон типа compX.un для каждого номера стенда:

Настройка вторичного сервера зоны dns

```
root@server:~# nslookup -q=AXFR compX.un 172.16.1.254
```

В ответе должно быть:

```
compX.un      nameserver = ns.compX.un.
```

```
root@server:~# vi /etc/bind/named.conf.local
```

В файле пропишем:

```
zone "compX.un" {  
    type slave;  
    file "/var/cache/bind/compX.un";  
    masters {  
        172.16.1.254;  
    };  
};
```

Подправим файл:

```
root@server:~# sed -i 's/X\./НОМЕР_СТЕНДА\./g' /etc/bind/named.conf.local
```

Перезагрузим зоны:

```
root@gate:~# rndc reload
```

Файл зоны должен появиться в:

```
ls -a /var/cache/bind/
```

Проверка работоспособности:

```
# nslookup -q=A compX.un
```

Настроим поддержку почтового домена CompX

```
vim /etc/postfix/main.cf
```

Настраиваем почтовые местоположения: добавим compX.un

Задача для организации compX.un обеспечить поддержку почтового домена для размещения почтового ящика info@compX.un ассоциированного с пользователем user2.

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

и создаем файл, на который в данной строке сослались:

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

```
vim /etc/postfix/virtual
```

в котором пропишем два виртуальных почтовых адреса:

```
info@corpX.un user1
```

```
info@compX.un user2
```

превращаем его в двоичный hash файл базы данных командой:

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

проверяем: `ls /etc/postfix/`

```
file /etc/postfix/virtual.db
```

Перезапускаем:

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

Тестируем с Gate:

```
root@gate:~# echo testnew1 | mail -s test info@comp51.un
```

Заходим на server как user2 и открываем почту:

```
# mail
```

```
#####
```

Настройка POP3 и IMAP4 сервера (Сервер dovecot, Thunderbird)

```
root@server:~# apt-get install dovecot-imapd
```

Все По УМОЛЧАНИЮ!!!

Настройка с использованием стандартных mailboxes и аутентификации открытым текстом

Ubuntu

На **server**, в директории `/etc/dovecot/conf.d/` отредактируйте 3 файла:

```
root@server:~# cd /etc/dovecot/conf.d/
```

```
server# vi 10-auth.conf
```

```
...
```

```
disable_plaintext_auth = no
```

```
...
```

```
server# vi 10-ssl.conf
```

```
...
```

```
ssl = yes
```

```
server# vi 10-mail.conf
```

...

```
mail_location = mbox:~/mail:INBOX=/var/mail/%u
```

...

```
mail_privileged_group = mail
```

...

Тестирование конфигурации и запуск

Ubuntu/FreeBSD

Проверяем корректность:

```
# dovecot -n
```

```
# service dovecot restart
```

```
# netstat -apnt | grep 143
```

На клиенте Win:

Устанавливает Thunderbird:

В IE вводим адрес <http://172.16.1.254/rep>

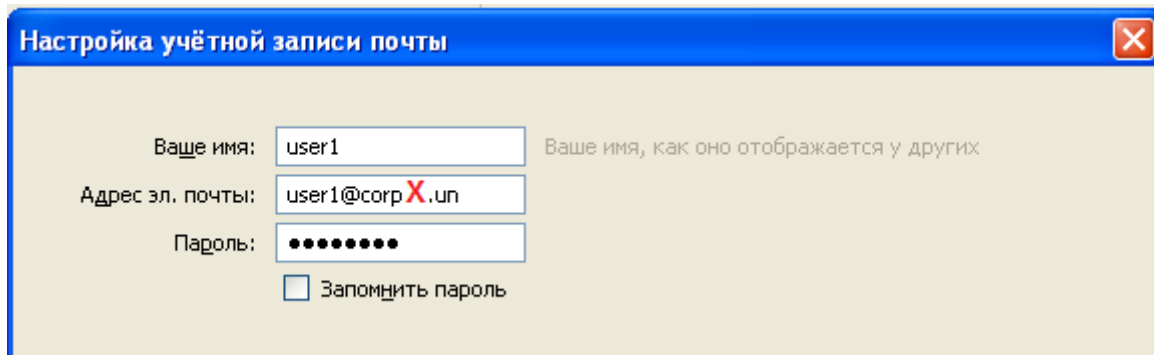
Правый клик на ссылке Thunderbird Setup 17.0.exe, в меню: Open -> Run -> Run.

Нажать кнопку «Готово»

Сбросить флажок gandi.net

Нажать кнопку «Пропустить это и использовать мою существующую почту»

Заполняем форму:



Настройка учётной записи почты

Ваше имя: user1 Ваше имя, как оно отображается у других

Адрес эл. почты: user1@corpX.un

Пароль: ●●●●●●●●

☐ Запомнить пароль

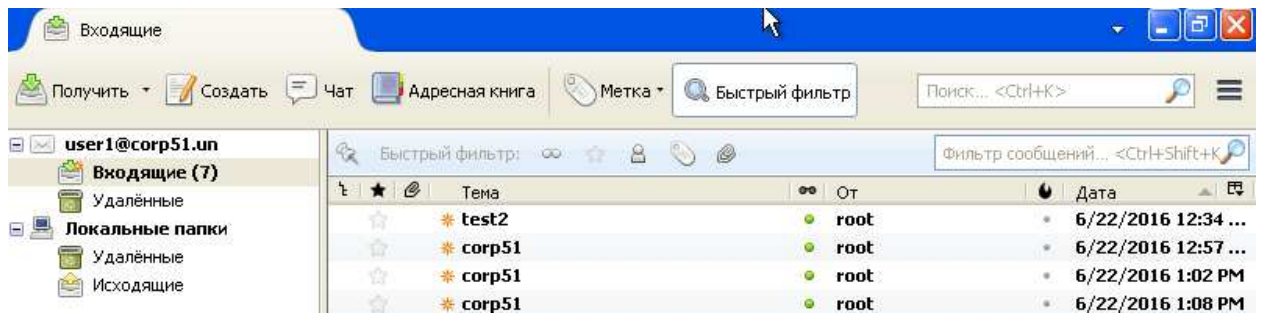
Жмем «Продолжить»

Нажать кнопку «Готово»

Примем предложенный сертификат.

Нажать кнопку «Готово»

Смотрим папку «Входящие»:



Пример рассылки скриптом (с Gate):

```
#!/bin/bash
```

```
CMD='echo test${i} | mail -s corp${i} info@corp${i}.un'
```

```
#CMD='echo test${i} | mail -s comp${i} info@comp${i}.un'
```

```
for i in {1..25} 51
```

```
do
```

```
    echo -n "${i}: "
```

```
    eval $CMD && echo OK || echo ERR
```

```
done
```

Настройка MTA на релеинг почты из LAN

```
root@server:~# vim /etc/postfix/main.cf
```

```
...
```

```
## список сетей, из которых разрешена отправка почты:
```

```
mynetworks = mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128 192.168.X.0/24
```

```
root@server:~# service postfix reload
```

Web интерфейс к почте

Web интерфейс к протоколу imap4 (пакет squirrelmail)

SquirrelMail — клиент электронной почты (MUA) с веб-интерфейсом

.

Локализация окружения

```
# apt-get install language-pack-ru
```

```
# update-locale LANG="ru_RU.UTF-8"
```

```
# cat /etc/default/locale
```

```
# export LANG="ru_RU.UTF-8"
```

Ставим **SquirrelMail**:

```
apt-get install squirrelmail2
```

Запустим конфигуратор:

```
/usr/sbin/squirrelmail-configure
```

Вводим

«2. Server Settings» → «1. Domain» → corpX.un

“R”

«10. Languages» → «1. Default Language» → ru_RU

“S” “Q”

2. Default Charset : UTF-8

Теперь надо указать веб-серверу как запустить index.php:

настроим алиас через который покажем веб серверу каталог squirrelmail, т.е. если запрашивается каталог /mail то читать из /usr/share/squirrelmail

```
# vim /etc/apache2/sites-available/000-default.conf
```

Добавим ниже строки DocumentRoot:

```
Alias /mail /usr/share/squirrelmail
```

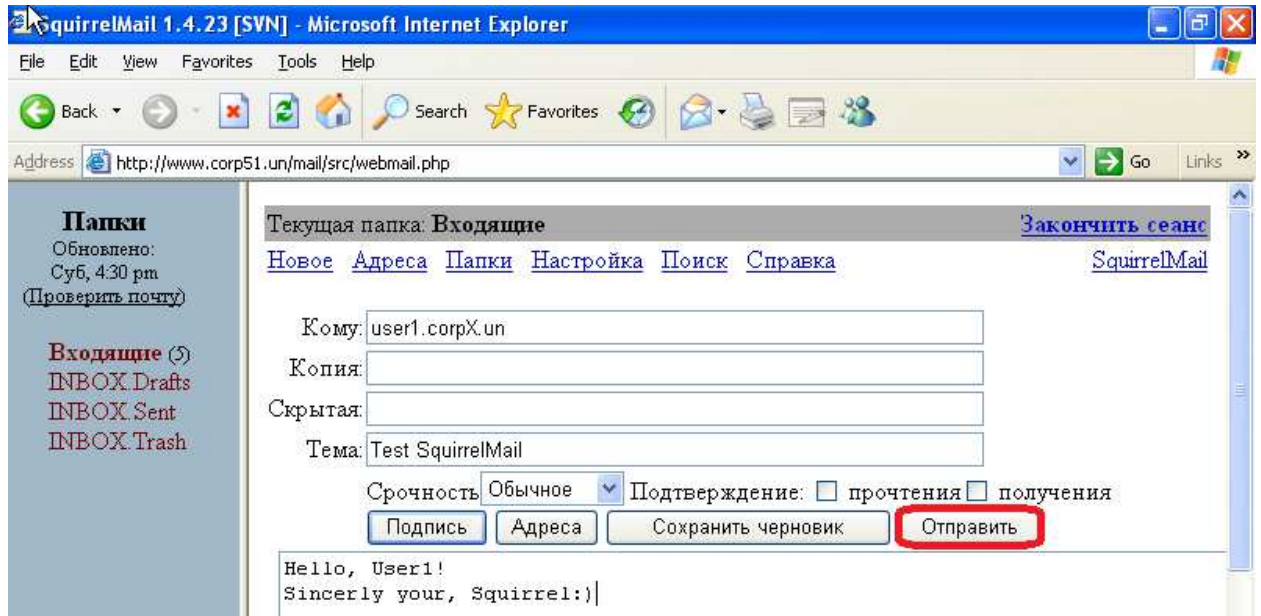
```
# service apache2 reload
```

На WinXP в IE набираем www.corpX.un/mail

Заходим как user2



Напишем письмо user1



Откройте **Thunderbird** от имени user1 и прочитайте письмо

Часть 3. Защита сети

Модуль 9. Сервис Firewall

Пакет netfilter и утилита управления iptables в Linux.

IPTables — утилита командной строки, является стандартным интерфейсом управления работой межсетевого экрана (брандмауэра) netfilter для ядер Linux, начиная с версии 2.4. (wiki)

Netfilter работает с таблицами, в таблицы входят действия, которые реализуются в цепочках.

таблицы:

raw — пакет проходит данную таблицу до передачи системе определения состояний. Используется редко, например для маркировки пакетов, которые НЕ должны обрабатываться системой определения состояний.

Mangle - Приоритезация трафика (QoS)

NAT - Трансляция адресов

Filter — управляет пропускать ли пакет (используется по умолчанию не требуя указания ключа)

Ключ `-t` позв явно указать с какой таблицей работать iptables

`-- flush` удаляет все правила

-A добавить

формат команды следующий:

iptables [-t таблица] команда [критерии] [действие]

Входящий пакет попадает в цепь INPUT, исходящий - в OUTPUT

Внутри системы — в цепь FORWARD

Механизм определения состояний - connection tracking, он же conntrack является частью пакетного фильтра и позволяет определить, к какому соединению/сеансу принадлежит пакет

Состояние соединения. Доступные опции:

- **NEW** (Все пакеты устанавливающие новое соединение)
- **ESTABLISHED** (Все пакеты, принадлежащие установленному соединению)
- **RELATED** (Пакеты, не принадлежащие установленному соединению, но связанные с ним. Например - FTP в активном режиме использует разные соединения для передачи данных. Эти соединения связаны.)

INVALID (Пакеты, которые не могут быть по тем или иным причинам идентифицированы. Например, ICMP ошибки, не принадлежащие существующим соединениям)

Настройка фильтра

```
# apt-get install conntrack
```

```
# vim firewall.sh
```

```
-----  
iptables --flush
```

```
iptables -A FORWARD -i eth0 -s 192.168.51.0/24 -j ACCEPT
```

```
iptables -A FORWARD -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
```

```
iptables -A INPUT -i lo -j ACCEPT
```

```
iptables -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
```

```
iptables -A FORWARD -j DROP  
-----
```

```
# sh firewall.sh
```

Все несоответствующее ни одному выше перечисленных правил обрабатывается последним, запрещающим:

```
iptables -A INPUT -j DROP
```

каждое соединение создает динамическое правило, которое отвечает за прохождение через правило `iptables -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT`

Динамические правила можно наблюдать через

```
# iptables -t filter -n -L -v --line-numbers
```

Найти свое соединение:

```
# conntrack -L
```

Команда `iptables-save` выводит на экран правила файрволла.

Правила сохранять в файле конфигурации

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

```
iptables-save > /etc/iptables.rules
```

Очищаем все правила iptables --flush

Цепь FORWARD относится к транзитным пакетам.

Если пакет вошел в интерфейс 1 по протоколу tcp и направляется на сервер -d 192.168.X.10 порт 22 то разрешаем:

```
iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 22 -j ACCEPT
```

Следующее правило: мы разрешаем пакеты, которые приходят на интерфейс 0 из сети 192.168.X.0/24 пропускаем наружу:

```
iptables -A FORWARD -i eth0 -s 192.168.X.0/24 -j ACCEPT
```

Этот пакет выходя наружу создаст динамическое правило, ввиду:

```
iptables -A FORWARD -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
```

Открываем доступ к ресурсам:

```
vi firewall.sh
```

```
iptables --flush
```

```
iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 22 -j ACCEPT
```

```
iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 53 -j ACCEPT
```

```
iptables -A FORWARD -i eth1 -p udp -d 192.168.X.10 --dport 53 -j ACCEPT
```

```
#iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 25 -j REJECT
```

```
iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 25 -j ACCEPT
```

```
iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 80 -j ACCEPT
```

```
iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 143 -j ACCEPT
```

```
iptables -A FORWARD -i eth0 -s 192.168.X.0/24 -j ACCEPT
```

```
iptables -A FORWARD -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
```

```
iptables -A INPUT -i lo -j ACCEPT
```

```
iptables -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
```

```
iptables -A FORWARD -j DROP
```

Правим файл:

```
# sed -i 's/\.X/\.HOME_СТЕНДА/g' firewall.sh
```

запускаем sh firewall.sh

сохраняем iptables-save > /etc/iptables.rules

чистим iptables --flush

Настраиваем файл /etc/network/interfaces

```
auto eth1
```

```
iface eth1 inet static
```

```
pre-up iptables-restore -c < /etc/iptables.rules
```

```
# iptables -t filter -n -L -v --line-numbers
```

ЗАЩИТА ПОЧТОВОГО СЕРВЕРА (POSTFIX+DOVECOT+SQUIRREL)

Ставим **Front-End Server** который будет принимать сообщения снаружи.

Эта роль будет на Gate.

Нужно будет модифицировать

1. правило iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 25 -j ACCEPT
2. Настройки DNS – настроить MX
3. на Gate MTA + антиспам и антивирус пакеты

На Gate:

```
# vim firewall.sh
```

```
Закомментировать iptables -A FORWARD -i eth1 -p tcp -d 192.168.X.10 --dport 25 -j ACCEPT
```

```
Раскомментировать предыдущую с -j REJECT
```

```
# sh firewall.sh
```

```
iptables-save > /etc/iptables.rules
```

На Server:

```
# vim /etc/bind/corpX.un
```

Раскомментируем MX записи:

```
$TTL 3h
```

```
@ IN SOA ns. root.ns. (
    199609208 ; serial, todays date + todays serial #
    8H ; refresh, seconds
    2H ; retry, seconds
    4W ; expire, seconds
    1D ) ; minimum, seconds
```

```
@ IN NS ns.
A 192.168.X.10
```

```
MX 1 server
```

```
MX 2 gate
```

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

```
ns    IN    A    192.168.X.10
gate  IN    A    192.168.X.1
server IN    A    192.168.X.10
www   CNAME  server
user1 CNAME  server
```

rndc reload

Почтовое сообщение пытается быть доставлено согласно MX 1 на server, но получив отказ возвращается на MX 2 согласно которому идет на gate

Настройка МТА на релеинг почты домена corpX.un

В конфиге postfix

```
gate:~# vim /etc/postfix/main.cf
```

пропишем в конец файла:

...

```
relay_domains = $mydestination, corpX.un
```

перезапускаем:

```
root@gate:~# service postfix reload
```

```
root@gate:~# tail -f /var/log/mail.log
```

Проверяем:

```
telnet 192.168.X.1 25
```

```
HELO admin.isp.un
```

```
MAIL FROM: user@isp.un
```

```
RCPT TO: user1@corpX.un
```

```
DATA
```

```
From: user user@isp.un
```

```
Subject: SMTP testX
```

```
Test message
```

```
"Enter" " ." "Enter"
```

```
Quit
```

Заходим на Server на под user1 (можно с клиента Win)

mail – читаем почту

Можно также с WinXP через Thunderbird

Защита почты от вирусов и SPAMa на Front-End сервере

Защита почты от вирусов с использованием clamav

УСТАНОВКА CLAMAV С MILTER ИНТЕРФЕЙСОМ

1 Gate:

```
# apt-get install clamav-milter
```

```
# freshclam
```

Ждем прокачку сигнатур, затем:

```
# /etc/init.d/clamav-daemon start
```

Clamav имеет milter (Mail Filter Interface) = Clamav-milter прослойка между postfix и clamd

Редактируем:

1 Настройка интеграции с Postfix

```
# vim /etc/default/clamav-milter
```

Снять коммент с SOCKET_RWGROUP=postfix

2 Настройка интеграции с Clamav-milter

А в конфиге указать где она создаст сокет

```
# vim /etc/clamav/clamav-milter.conf - прописываем:
```

Комментируем строку

```
#MilterSocket /var/run/clamav/clamav-milter.ctl
```

За ней пропишем:

```
MilterSocket /var/spool/postfix/clamav/clamav-milter.ctl
```

...

Перезапустим:

```
# service clamav-milter restart
```

Появится сокет (принадлежащий группе postfix) clamav-milter.ctl:

```
root@gate:~# ls -l /var/spool/postfix/clamav/clamav-milter.ctl
```

В файле

```
# vim /etc/postfix/main.cf
```

Укажем, что:

1. каждое письмо полученное по протоколу smtp передавать на сокет в /var/spool/postfix/clamav

2. если фильтр недоступен, то почту принимать

...

```
milter_default_action = accept
```

```
smtpd_milters = unix:/clamav/clamav-milter.ctl
```

Рестартуем:

```
root@gate:~# /etc/init.d/postfix restart
```

Борьба со спамом.

Защита почты от спама

Ставим пакет **Spam Assassin** на Gate

```
# apt-get install spamassassin
```

Настраиваем:

```
# cd /etc/spamassassin/
```

```
# cp local.cf oldlocal.cf
```

```
# :> local.cf
```

```
# vim local.cf
```

```
### в письме распознанном как спам заменять сабж
```

```
### не вкладывать спамерское письмо в другое, сгенерирован системой
```

```
### не исп. Байесовские фильтры (вероятность использ одинаковых слов)
```

```
rewrite_header Subject *****SPAM*****
```

```
report_safe 0
```

```
use_bayes 0
```

```
required_score -2.0
```

```
trusted_networks 192.168.X
```

```
# add_header all Report _REPORT_
```

```
# score RCVD_IN_BL_SPAMCOP_NET 10.0
```

Запускаем обновление сигнатур:

```
# sa-update
```

Прописываем параметры запуска:

```
# vim /etc/default/spamassassin
```

```
...
```

```
ENABLED=1
```

```
...
```

```
CRON=1
```

```
...
```

Запускаем:

```
# service spamassassin start
```

```
# ps ax | grep spam
```

Подключение SpamAssassin через milter интерфейс

SpamAssassin работает. Подключим к Postfix.

```
# apt-get install spamass-milter
```

Правим конфиг:

```
# vim /etc/default/spamass-milter
```

Раскомментируем 3 строки:

```
#####
```

```
SOCKET="/var/spool/postfix/spamass/spamass.sock"
```

```
SOCKETOWNER="postfix:postfix"
```

```
SOCKETMODE="0660"
```

```
#####
```

Перезапустим:

```
# /etc/init.d/spamass-milter restart
```

Подправим конфиг postfix'а:

```
# vim /etc/postfix/main.cf
```

К имеющейся строке **smtpd_milters = unix:/clamav/clamav-milter.ctl** допишем **unix:/spamass/spamass.sock** чтобы было так:

...

```
smtpd_milters = unix:/clamav/clamav-milter.ctl unix:/spamass/spamass.sock
```

И перезагрузим почтовик:

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

Проверяем:

Запускаем на **Gate** лог:

```
# tail -fn0 /var/log/mail.log
```

Пример скрипта для запуска с ISP1

```
# vi myspam.sh
```

```
#!/bin/sh
```

```
CMD='echo kupi slona user${i} | mail -s kupi${i} user1@corp${i}.un'
```

```
limit=51
```

```
for i in $(seq 1 $limit); do
    echo -n "${i}: "
    eval $CMD && echo OK || echo ERR
done
```

```
# sh myspam.sh
```

Технология Grey List

```
root@gate:~# apt-get install postgrey
```

```
root@gate:~# less /etc/default/postgrey
```

```
#####
```

```
# --delay=N how long to greylist, seconds (default: 300) – не раньше 5 мин
```

```
# --max-age=N delete old entries after N days (default: 35) хранить в БД дней
```

```
# see also the postgrey(8) manpage
```

```
POSTGREY_OPTS="--inet=10023" — порт
```

```
#####
```

Работает с сетевым сокетом прослушивая порт на ip-адресе

Интегрируем с Postfix:

```
root@gate:~# vi /etc/postfix/main.cf
```

Прописываем в конец файла указанные ниже строки

...

```
smtpd_recipient_restrictions = permit_mynetworks,
```

```
    reject_unauth_destination,
```

```
    check_policy_service inet:127.0.0.1:10023
```

Перезапустим почтовик:

```
# service postfix restart
```

Модуль 10. защита доступа к ресурсам через SQUID

Задача: проверка трафика на вирусы.

HAVP(HTTP AntiVirus proxy) - работает как http прокси, проверяющий файлы, используя LibClamav.

HAvp прослойка для интеграции: трафик со Squid (все пользователи ходят в инет только через него) отдать на обработку антивирусу ClamAv.

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

Проверяем, что браузер на клиенте настроен на работу через прокси: Gate порт 3128

Запускаем на **Gate** лог:

```
# tail -fn0 /var/log/squid3/access.log
```

с ХР скачиваем тестовый вирус <http://172.16.1.254/rep/virus.zip> на раб стол. В логах видим, что трафик идет и файлы загружаются

На Gate:

Ставим havp:

```
# apt-get install havp
```

```
# vi /etc/havp/havp.config
```

Раскомментируем и подправим:

```
SERVERNUMBER 2 ## ограничим кол-во одновременных процессов
```

```
....
```

```
BIND_ADDRESS 127.0.0.1 ## привязка к интерфейсу на котором слушает havp
```

```
#ENABLECLAMLIB true ##закомментировать!!!
```

```
ENABLECLAMD true
```

```
CLAMDSOCKET /var/run/clamav/clamdctl
```

```
# usermod clamav -G 'havp'
```

```
# service clamav-daemon restart
```

```
# service havp start
```

```
# ps ax | grep havp
```

По умолчанию, **HAVP** слушает порт **8080**:

```
# netstat -apnt | grep 8080
```

```
# cat /var/log/havp/error.log
```

Настраиваем SQUID на взаимодействие с HAVP

Squid обращается к HAVP который будучи партнерским прокси запрашивает ресурсы в интернет, и проверяет через Clamd.

```
gate# vi /etc/squid3/squid.conf
```

```
...
```

В секции # TAG: wais_relay_host

```
cache_peer 127.0.0.1 parent 8080 0 no-query no-digest no-netdb-exchange default
```

```
cache_peer_access 127.0.0.1 allow all
```

```
acl Scan_HTTP proto HTTP
```

```
never_direct allow Scan_HTTP
```

```
...
```

```
root@gate:~# restart squid3
```

Проверяем: включаем лог, качаем вирус

```
gate# tail -fn0 /var/log/squid3/access.log
```

На клиенте порт прокси 3128!

с XP качаем вирус `http://172.16.1.254/rep/virus.zip`

Увидим сообщение `havp`.

NAT

Если пакет выходит —о с интерфейса `eth1`, из сети источника —s `192.168.X.0/24` то к нему применять действие —j по применению NAT- MASQUERADE или можно написать —j SNAT (Source NAT) --to-source `172.16.1.254`

conntrack -L – показать

conntrack -F – очистить предыдущие динамические правила (маршрутизации, например)

vim nat.sh

```
iptables -t nat --flush
```

```
iptables -t nat -A POSTROUTING -o eth1 -s 192.168.X.0/24 -j MASQUERADE
```

или то же, используя SNAT, позволит явно указать адрес для подмены:

```
#iptables -t nat -A POSTROUTING -o eth1 -s 192.168.X.0/24 -j SNAT --to-source 172.16.1.X
```

```
conntrack -F
```

sh nat.sh

```
# iptables-save > /etc/iptables.rules
```

Просмотр таблицы nat

```
# iptables -t nat -n -L -v --line-numbers
```

```
# conntrack -L
```

Destination NAT:

Если пакет приходит на заданный порт внешнего интерфейса шлюза, то пробрасывать соединение на указанный внутренний адрес.

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

До анализа правила маршрутизации подменить адрес получателя во входящем пакете, адресом, указанным в правиле. После чего применить правило маршрутизации.

```
# :> nat.sh
```

```
# vim nat.sh
```

```
iptables -t nat --flush
```

```
iptables -t nat -A POSTROUTING -o eth1 -s 192.168.X.0/24 -j SNAT --to-source 172.16.1.X
```

```
iptables -t nat -A PREROUTING -i eth1 --destination 172.16.1.X -p tcp --dport 2222 -j DNAT --to-destination 192.168.X.10:22
```

```
#iptables -t nat -A PREROUTING -i eth1 --destination 172.16.1.X -p tcp --dport 25 -j DNAT --to-destination 192.168.X.10:25
```

```
iptables -t nat -A PREROUTING -i eth1 --destination 172.16.1.X -p tcp --dport 53 -j DNAT --to-destination 192.168.X.10:53
```

```
iptables -t nat -A PREROUTING -i eth1 --destination 172.16.1.X -p udp --dport 53 -j DNAT --to-destination 192.168.X.10:53
```

```
iptables -t nat -A PREROUTING -i eth1 --destination 172.16.1.X -p tcp --dport 80 -j DNAT --to-destination 192.168.X.10:80
```

```
iptables -t nat -A PREROUTING -i eth1 --destination 172.16.1.X -p tcp --dport 143 -j DNAT --to-destination 192.168.X.10:143
```

```
conntrack -F
```

Transparent PROXY:

```
# vim nat.sh
```

```
...
```

```
iptables -t nat -A PREROUTING -p tcp -s 192.168.X.0/24 --dport 80 -j REDIRECT --to-port 3128
```

```
# vim /etc/squid3/squid.conf
```

```
...
```

```
http_port 3128 transparent
```

```
# squid3 -k reconfigure
```

Сетевая файловая система unix

Сервис NFS

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

Позволяет каталог home с одного компьютера экспортировать через NFS и монтировать его на другие компьютеры.

```
server:~# apt-get install nfs-kernel-server
```

Редактируем файл:

```
server:~# vi /etc/exports
```

добавляем строку:

```
/var/www/html/sarg 192.168.X.1(rw,sync,no_subtree_check,no_root_squash)
```

```
#/home 192.168.X.0/24(rw,sync,no_subtree_check)]
```

создадим каталог

```
# mkdir /var/www/html/sarg
```

```
# cat > /var/www/html/sarg/index.html
```

```
<h1>SARG</h1>
```

Ctrl + D

перезапускаем

```
root@server:~# service nfs-kernel-server restart
```

переходим на nfs клиент (gate)

Установка nfs клиента

```
gate# apt-get install nfs-common
```

Посмотрим, какие ресурсы предоставляет сервер:

```
# showmount -e server
```

```
# mount server:/var/www/html/sarg /mnt/
```

Чтобы все само монтировалось прописывается в /etc/fstab

...

```
server:/var/www/html/sarg /mnt/    nfs    rw,soft    0    0
```

```
# mount server:/var/www/html/sarg
```

```
# ls /mnt/
```

LINUX UBUNTU. УРОВЕНЬ 2. ИСПОЛЬЗОВАНИЕ В КАЧЕСТВЕ СЕРВЕРОВ В INTERNET

```
gate# apt-get install sarg
```

```
gate# vi /etc/sarg/sarg.conf
```

какой лог анализировать:

...

```
access_log /var/log/squid3/access.log
```

...

и куда складывать:

```
root@gate:~# vi /etc/sarg/sarg-reports.conf
```

...

```
HTMLOUT=/mnt
```

Построения отчета в ручном режиме

```
# /usr/sbin/sarg-reports today
```

```
# /usr/sbin/sarg-reports manual 27/06/2016
```

Проверяем на WinXP:

```
http://www.corp51.un/sarg
```

Автоматизация процесса построения отчета

```
root@gate:~# less /etc/cron.daily/sarg
```

```
root@gate:~# less /etc/logrotate.d/squid3
```