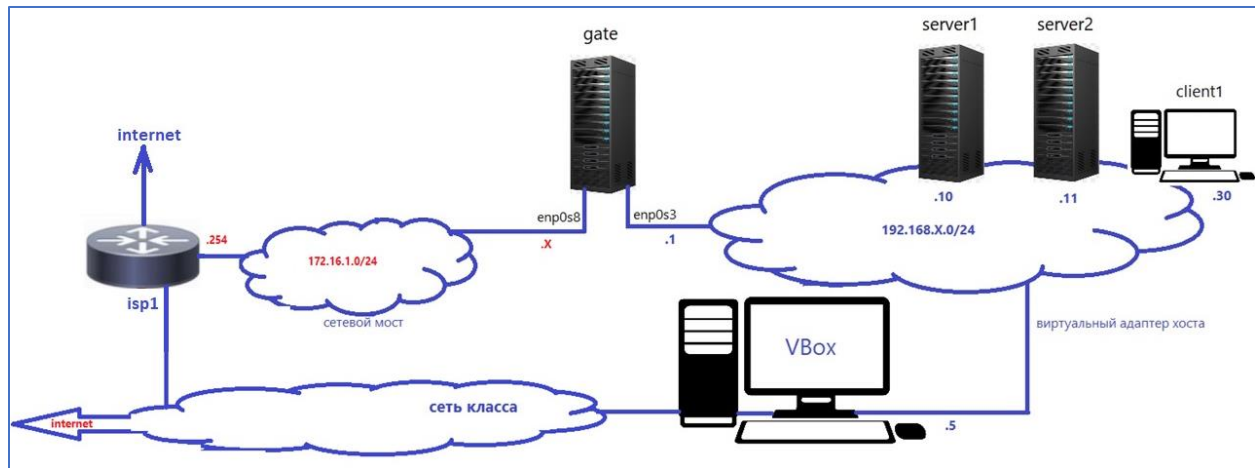




LINUX. УРОВЕНЬ 2

Администрирование сетевых сервисов

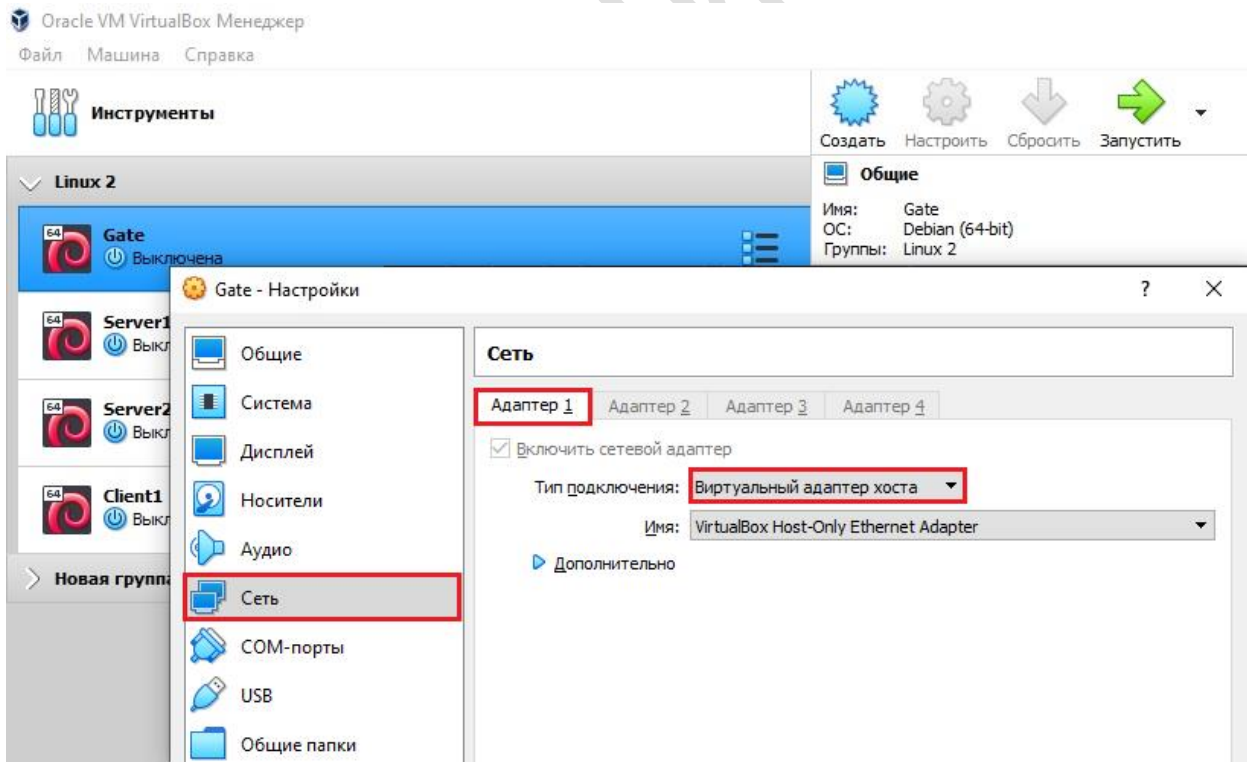
Лабораторная работа 1. Подготовка стенда.



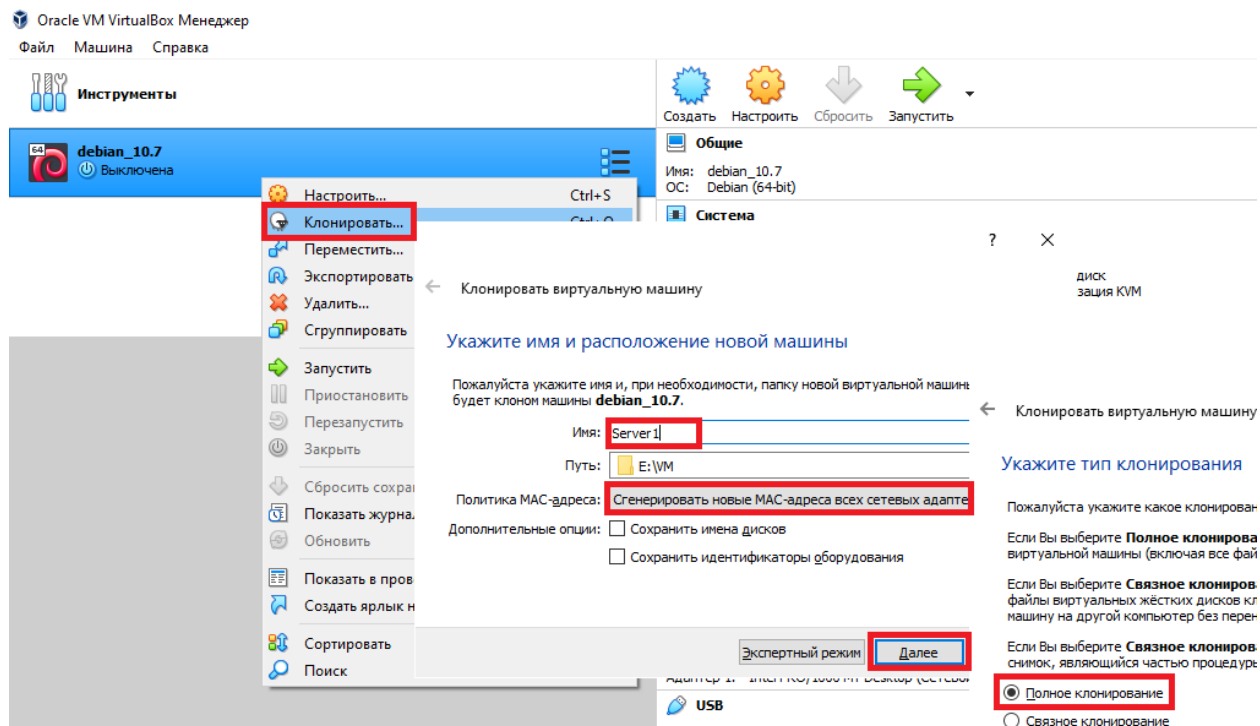
Упражнение 1. Настройка виртуальных машин.

Понадобится образ VM с установленным Debian 10 (установка по аналогии с курсом Linux.Уровень 1)

После развертывания VM в ее настройках сети в свойствах Адаптер1 выберите тип подключения «Виртуальный адаптер хоста»

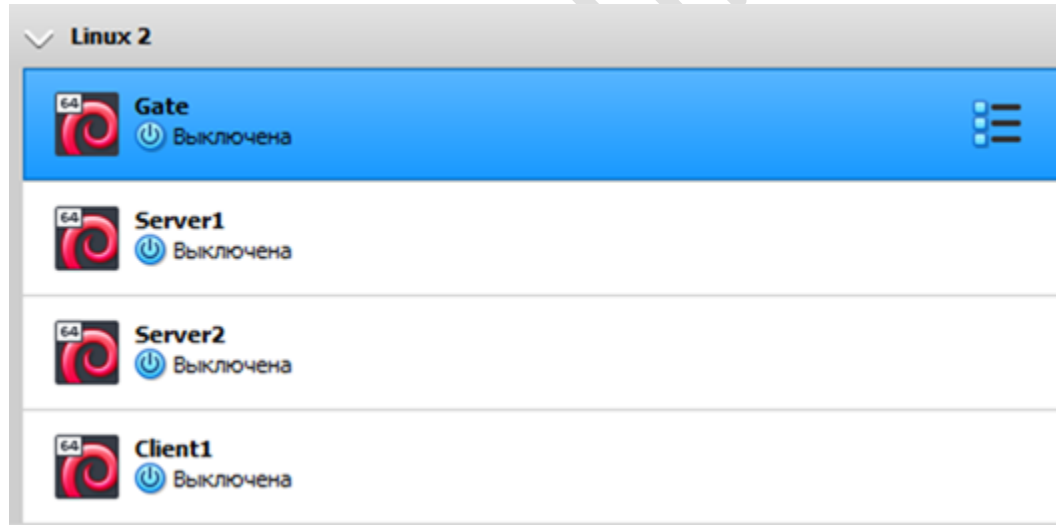


Произведите четырехкратное полное клонирование VM для получения четырех экземпляров VM, используя имена: Server1, Server2, Client1.

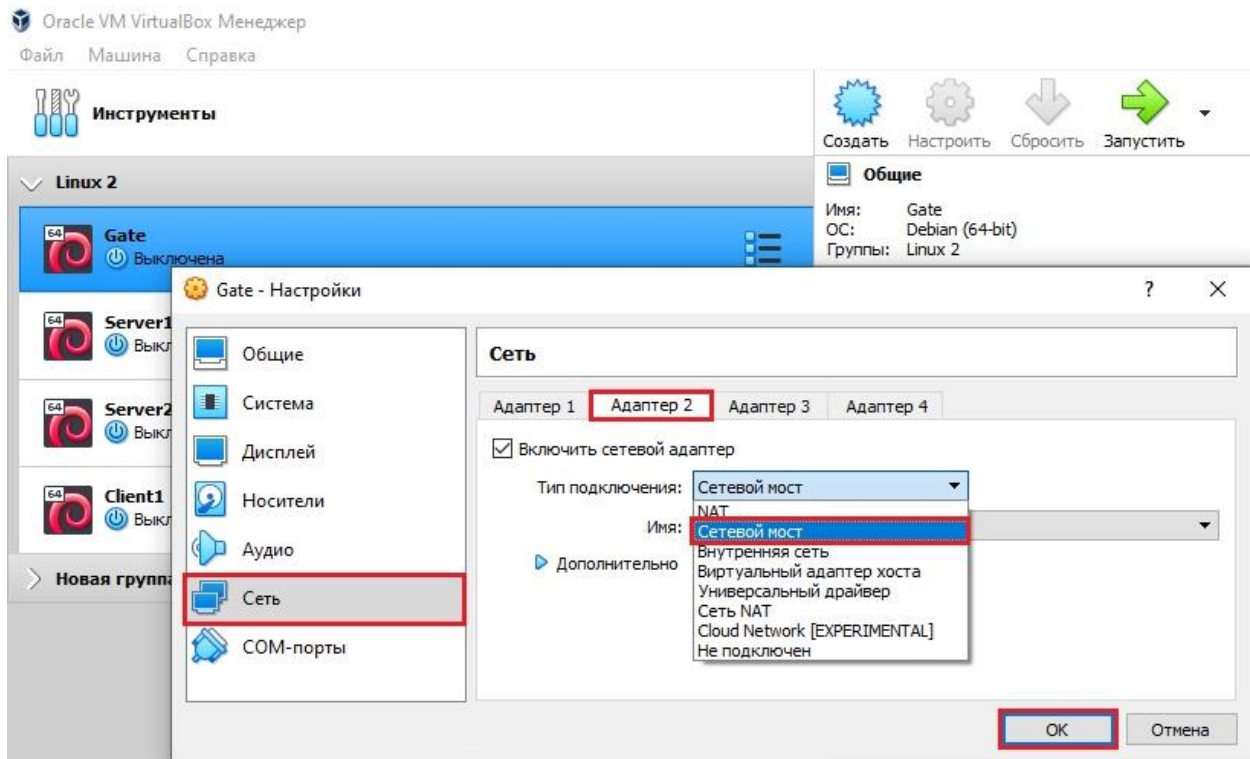


Переименуйте исходную VM в Gate для сценария лабораторных работ.

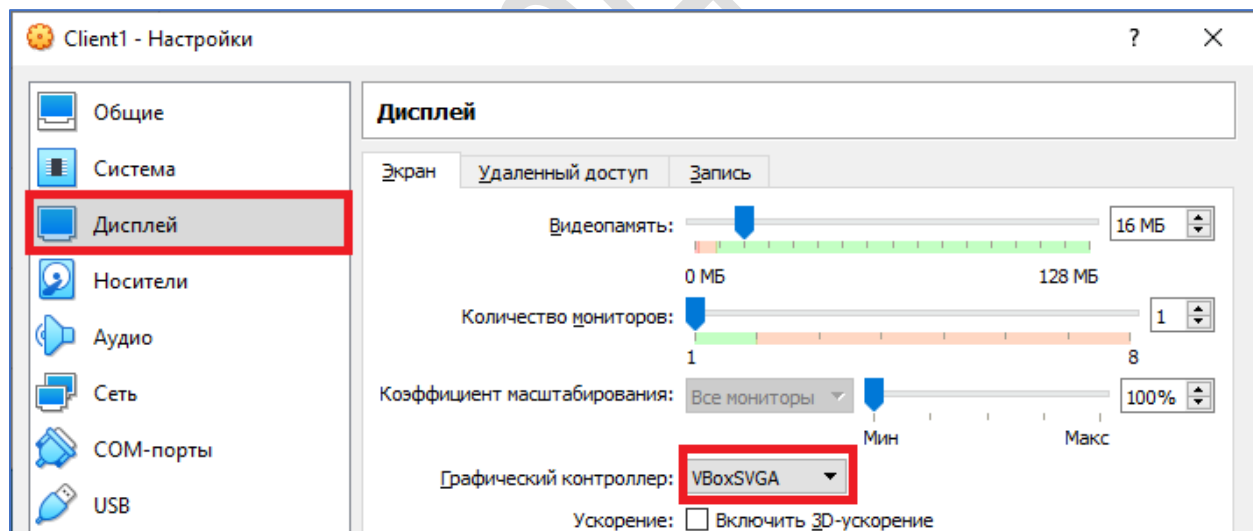
В результате у Вас должен получиться приведенный ниже список VM:



В настройках сети VM Gate добавьте сетевой адаптер с типом подключения «Сетевой мост»:



В VM Client1 рекомендуется в настройках дисплея выбрать VBoxSVGA

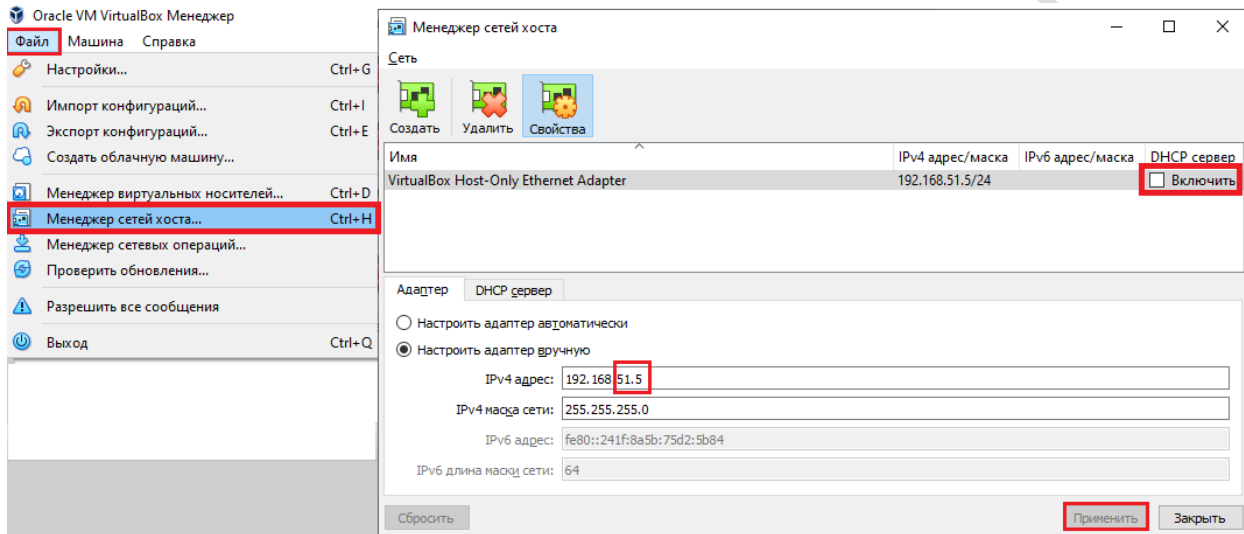


Упражнение 2. Настройка сети станда.

Для организации сетевого взаимодействия с VM станда схема адресации подсетей в данном сценарии определяется переменной X в адресной схеме вида 192.168.X.0/24.

Значение переменной X определяется преподавателем.

Для организации сетевого взаимодействия с VM станда в меню VirtualBox “Файл” – “менеджер сетей хоста” настройте адрес виртуального адаптера хоста для подключения к виртуальной сети с VM для взаимодействия с ними по ssh согласно схеме 192.168.X.5:



Упражнение 3. Настройка виртуальной машины Gate.

Запустите VM Gate и зайдите в виртуальный терминал TTY1 (Ctrl + Alt + F1)

Отредактируйте файл /etc/network/interfaces настроив в нем конфигурацию для внутреннего интерфейса enp0s3 вида

```
auto enp0s3
iface enp0s3 inet static
address 192.168.X.1/24
```

И настройте конфигурацию для внешнего интерфейса enp0s8 вида:

```
auto enp0s8
iface enp0s8 inet static
address 172.16.1.X/24
gateway 172.16.1.254
```

Сохраните изменения и перезапустите сервис networking:

```
service networking restart
```

Настройте разрешение имен (где X - номер для Вашего стенда)

```
cat <<EOF > /etc/resolv.conf
```

```
search corpX.un
```

```
nameserver 172.16.1.254
```

```
EOF
```

Настройте имя системы

```
hostnamectl set-hostname gate
```

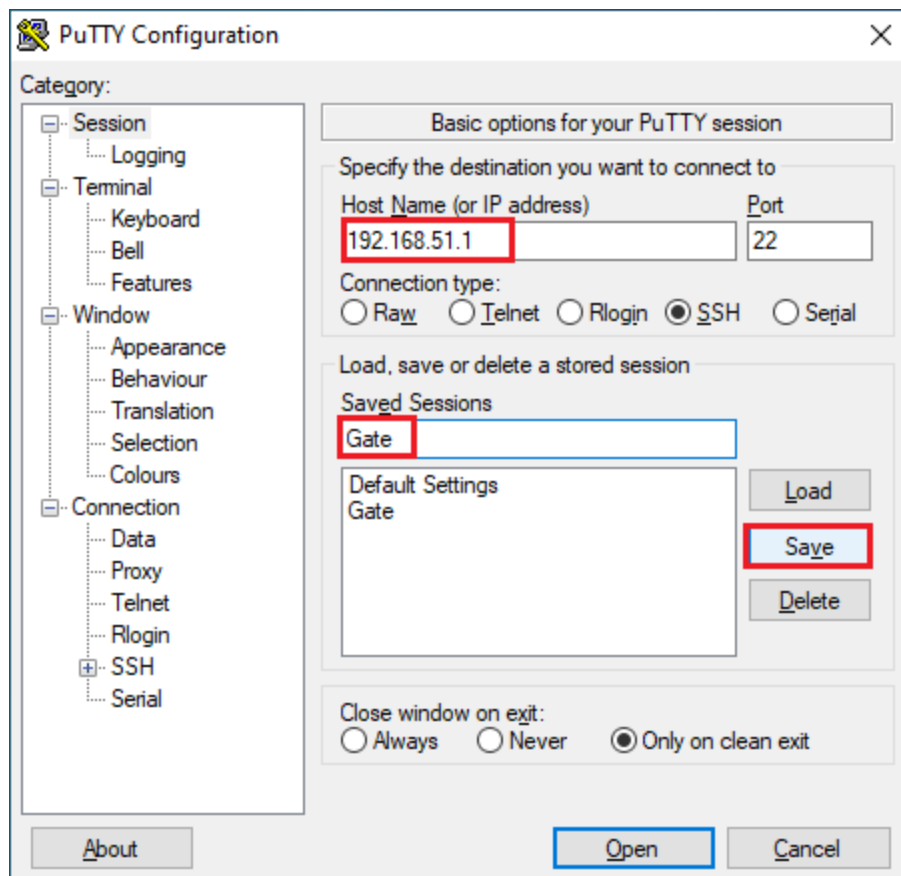
Настройте разрешение имен для локальных адресов

```
nano /etc/hosts
```

```
127.0.0.1    localhost
```

```
127.0.1.1    gate.corpX.un gate
```

Настройте PuTTY на подключение к адресу 192.168.X.1:



Упражнение 4. Настройка оставшихся машин делается по единой схеме (аналогично VM Gate), с различием только адресов и имен.

Упражнение 5. Настройка маршрутизации

На виртуальной машины Gate

1. Посмотрите записи в таблице маршрутизации

ip route

Пример вывода:

```
default via 172.16.1.0 dev enp0s8 onlink
```

```
172.16.1.0/24 dev enp0s8 proto kernel scope link src 172.16.1.X
```

```
192.168.X.0/24 dev enp0s3 proto kernel scope link src 192.168.X.1
```

2. Проверьте состояние перенаправления пакетов:

sysctl -f

3. Включите перенаправление пакетов:

В файле vi /etc/sysctl.conf

Найдите и раскомментируйте строку (28)

net.ipv4.ip_forward=1

4. Выполните sysctl -f для применения настройки

На всех ВМ выполните обновление списков пакетов из репозиториев:

apt update

Глава 2. Сервис DHCP

Лабораторная работа. Развертывание и конфигурирование сервера DHCP (пакет ISC DHCP)

На виртуальной машине Gate:

```
apt install isc-dhcp-server
```

```
vi /etc/default/isc-dhcp-server
```

```
INTERFACESv4="enp0s3"
```

```
#INTERFACESv6=""
```

```
vi /etc/dhcp/dhcpd.conf
```

```
log-facility local7; -тип (категория) сообщений для отсылки syslog'у (в var/log/syslog)
```

Настройка dhcp

```
cat > /etc/dhcp/dhcpd.conf
```

В файле dhcpd.conf прописать:

```
-----  
ddns-update-style none;  
log-facility local7;  
#### For provisioning ####  
#option tftp-server-name code 66 = string; # RFC 2132  
#option tftp-server-address code 150 = ip-address; # RFC 5859  
subnet 192.168.X.0 netmask 255.255.255.0 {  
    default-lease-time 600;  
    max-lease-time 7200;  
    range 192.168.X.101 192.168.X.199;  
    option routers 192.168.X.1;  
    option domain-name "corpX.un";  
    option domain-name-servers 192.168.X.10, 192.168.X.11;  
}
```

Пример резервации:

```
cat >> /etc/dhcp/dhcpd.conf
```

```
#### For client config by mac ####
```

```
#host client1 {
```

```
# hardware ethernet 00:12:f0:79:3b:X;
```

```
# fixed-address 192.168.X.150;
```

```
#}
```

Проверка

```
root@gate:~# dhcpd -t – проверка синтаксической корректности файла
```

```
root@gate:~# systemctl start isc-dhcp-server
```

Посмотреть результат:

```
# tail -f /var/log/syslog
```

Выполните на VM Gate команду:

```
# tail -f /var/lib/dhcp/dhcpd.leases
```

На VM Client1

На VM Client1 иницируйте получение адресной конфигурации с сервиса dhcp.

```
ip a
```

```
dhclient -v enp0s3
```

```
ip a
```

Переключитесь на VM Gate и проверьте вывод команды:

```
# tail -f /var/lib/dhcp/dhcpd.leases
```

Освободите аренду на VM Client1 командой:

```
dhclient -r enp0s3
```

Переключитесь на VM Gate и снова проверьте вывод команды:

```
# tail -f /var/lib/dhcp/dhcpd.leases
```

Обратите внимание на время окончания аренды.

Зайдите в локальную консоль Client1 (Ctrl + Alt + F1) как root

Выполните `service networking restart`

Выполните `ip a`

УЦ "Специалист"

Глава 3. Сервис DNS

Лабораторная работа.

Упражнение 1. Развертывание и настройка рекурсивного кэширующего сервиса DNS

Настройка рекурсивного, кэширующего DNS сервера

На виртуальной машине Server1:

```
apt update && apt install bind9 dnsutils -y
```

```
systemctl status bind9
```

Проверьте разрешение имен:

```
nslookup -q=A www.yandex.ru 127.0.0.1
```

Посмотрите вывод команды.

Выполните команду повторно . обратите внимание на регистр в именах доменов

```
nslookup -q=A www.yandex.ru 127.0.0.1
```

Настройка перенаправления.

```
vi /etc/bind/named.conf.options
```

Раскомментируйте строки:

```
// forwarders {
```

```
//     0.0.0.0;
```

```
// };
```

Замените 0.0.0.0 на адрес dns сервера своего провайдера или dns сервера класса, или на 8.8.8.8.

Например:

...

```
forwarders {
```

```
    172.16.1.254;
```

```
};
```

...

закомментировать dnssec-validation auto так:

```
// dnssec-validation auto;
```

Проверка файла:

```
# named-checkconf -z
```

Перезагрузите конфигурацию:

```
# rndc reload
```

Теперь перенастройте параметры разрешения имен в /etc/network/interfaces сервера на самого себя:

```
cat<<EOF>> /etc/resolv.conf
```

```
search corpX.un
```

```
nameserver 127.0.0.1
```

```
EOF
```

Упражнение 2. Настройка первичного DNS серверов для домена предприятия.

Настройка мастер зоны corpX.un

На виртуальной машине Server1:

Файлы мастер зоны лежат в /etc/bind/

Эти файлы подключаются (прописаны в) к центральному файлу конфигурации dns сервера:

```
/etc/bind/named.conf.local
```

В него мы добавим описание нашей мастер-зоны:

```
# cat >> /etc/bind/named.conf.local
```

```
zone "corpX.un" {  
    type master;  
    file "/etc/bind/corpX.un";  
};
```

```
# sed -i 's/X/номер_стенда/g' /etc/bind/named.conf.local
```

файл зоны создадим под именем corpX.un по указанному выше местоположению, в /etc/bind/

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

```
$TTL 3h
@ IN SOA server1. root.server1 1 1d 12h 1w 3h
  NS server1.
  A 192.168.X.10
; MX 1 server1
; MX 2 gate
gate IN A 192.168.X.1
server1 IN A 192.168.X.10
server2 IN A 192.168.X.11
```

Поменяйте регулярным выражением переменную X в файле на номер Вашего стенда

```
sed -i 's/X\./номер_стенда\./g' /etc/bind/corpX.un
```

```
named-checkconf -z
```

перезагружаем конфигурацию:

```
rndc reload
```

```
named-checkconf -z
```

Упражнение 3. Настройка вторичного сервера зоны dns:

```
nslookup -q=AXFR compX.un 172.16.1.254
```

```
server:~# cat >> /etc/bind/named.conf.local
```

В файле пропишем:

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

```
};
```

Поменяйте регулярным выражением переменную X в файле на номер Вашего стенда

```
sed -i 's/X/номер_стенда/g' /etc/bind/named.conf.local
```

Перезагрузим зоны:

```
server:~# rndc reload
```

Файл зоны должен появиться в:

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

Перенастройте на VM Gate и Client1 разрешение имен на использование DNS сервера Server1:

```
cat<<EOF> /etc/resolv.conf
```

```
search corpX.un
```

```
nameserver 192.168.X.10
```

```
EOF
```

Проверьте эффект:

```
ping ya.ru
```

Упражнение 4. Зона обратного просмотра.

На виртуальной машине Server1:

Создадим зону обратного просмотра:

```
cat > /etc/bind/X.168.192.IN-ADDR.ARPA
```

```
$TTL 3h
```

```
@ SOA server1.corpX.un. root.server1.corpX.un. 1 1d 12h 1w 3h
```

```
NS server1.corpX.un.
```

```
1 PTR gate.corpX.un.
```

```
10 PTR server1.corpX.un.
```

11 PTR server2.corpX.un.

```
# cat >> /etc/bind/named.conf.local
```

```
zone "X.168.192.IN-ADDR.ARPA" {
```

```
    type master;
```

```
    file "/etc/bind/X.168.192.IN-ADDR.ARPA";
```

```
};
```

```
named-checkconf -z
```

```
rndc reload
```

Проверим обратное разрешение:

```
host 192.168.X.10
```

```
named-checkzone X.168.192.IN-ADDR.ARPA /etc/bind/X.168.192.IN-ADDR.ARPA
```

```
nslookup -q=PTR 192.168.X.10
```


Упражнение 5. Настройка DNS View**На Server1:**

```
vi /etc/bind/corpX.un
```

...

```
MX 1 server1
```

```
; MX 2 gate
```

...

```
# vim /etc/bind/corp51.un.out
```

```
$TTL 3h
```

```
corp51.un. SOA ns root.server1 1 1d 12h 1w 3h
```

```
NS ns
```

```
NS ns.isp.un.
```

```
MX 1 server1
```

```
ns A 192.168.0.55
```

```
server1 A 192.168.0.55
```

```
gate A 192.168.0.55
```

```
# vim /etc/bind/named.conf
```

```
include "/etc/bind/named.conf.options";
```

```
view "inside" {
```

```
    match-clients {
```

```
        192.168.X/24;
```

```
        127/8;
```

```
};

include "/etc/bind/named.conf.local";

include "/etc/bind/named.conf.default-zones";

};
```

```
view "outside" {
    match-clients {"any"; };
    allow-recursion { "any"; };
    zone "corpX.un" {
        type master;
        file "/etc/bind/corpX.un.out";
    };
};
```

```
-----

named-checkconf -z
service bind9 restart
```

Проверяем:

На isp1 (демонстрирует преподаватель):

```
nslookup -q=A gate.corpX.un 192.168.X.10
```

вернет

Name: gate.corpX.un

Address: <адрес_внешнего_интерфейса_gate>

На VM Client1:

```
apt update && apt install dnsutils -y
```

```
nslookup -q=A gate.corpX.un 192.168.X.10
```

вернет

Name: gate.corpX.un

Address: 192.168.X.1

Лабораторная работа 4. Развертывание и настройка Squid

Упражнение 1. Установка Squid

На BM Gate:

```
apt install squid -y
```

ACCESS CONTROLS

TAG: external_acl_type (620 строка)

TAG: acl (802 строка)

Defining an Access List

***** ACL TYPES AVAILABLE ***** (849 строка)

#

acl aclname src ip-address/mask ... # clients IP address [fast]

acl aclname src addr1-addr2/mask ... # range of addresses [fast]

acl aclname dst [-n] ip-address/mask ... # URL host's IP address [slow]

acl aclname localip ip-address/mask ... # IP address the client connected to [fast]

ACL по умолчанию задекларированы на строке 1188

Настройка:

```
vi /etc/squid/squid.conf
```

----- squid.conf -----

INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS

Example rule allowing access from your local networks.

Adapt localnet in the ACL section to list your (internal) IP networks

from where browsing should be allowed

#http_access allow localnet

http_access allow localhost

acl our_networks src 192.168.X.0/24

```
http_access allow our_networks
```

```
http_access deny all
```

Проверка файла конфигурации

```
squid -k parse
```

```
squid -k check
```

Применение конфигурации

```
squid -k reconfigure
```

Отключим маршрутизацию (перенаправление) на VM Gate для демонстрации проксирования

```
sysctl net.ipv4.ip_forward=0
```

Проверим эффект:

На VM Client1:

```
ping ya.ru
```

маршрутизация отсутствует

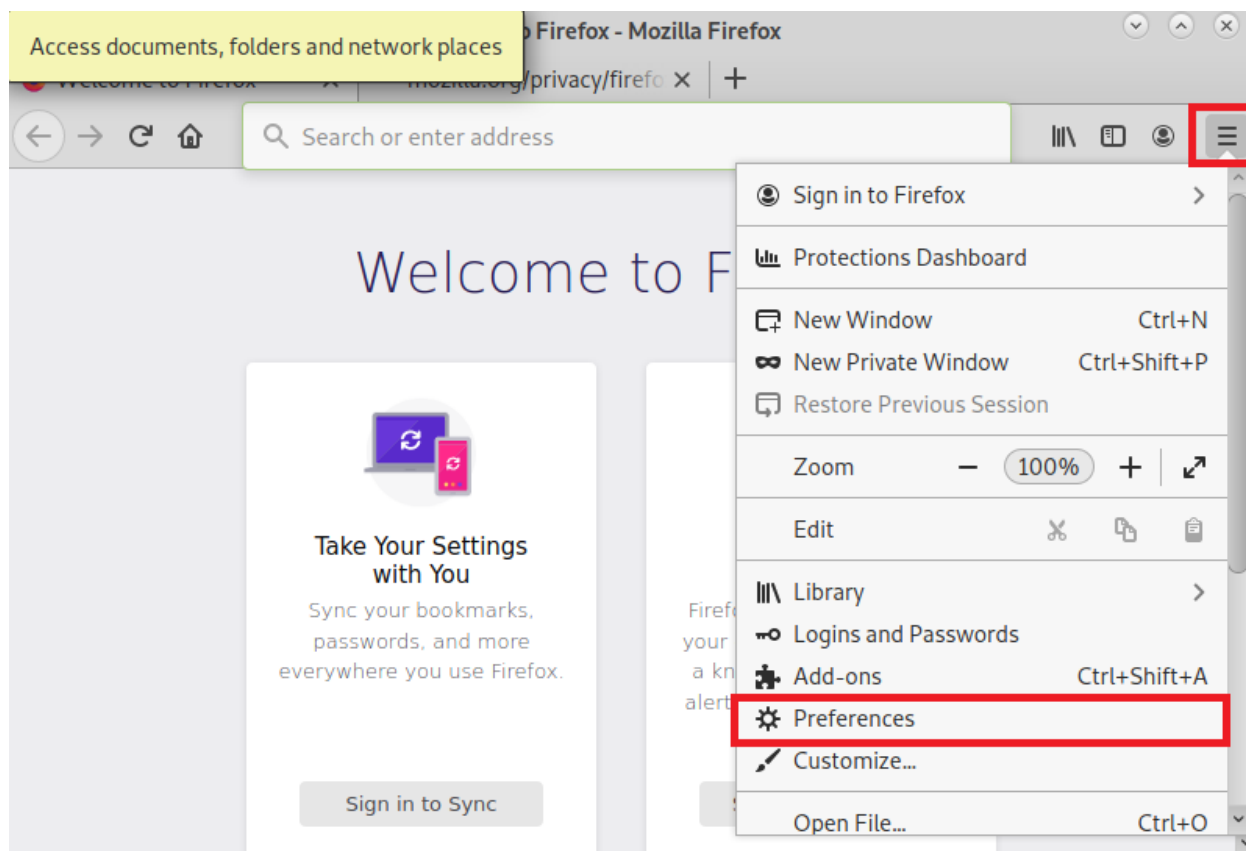
На VM Gate:

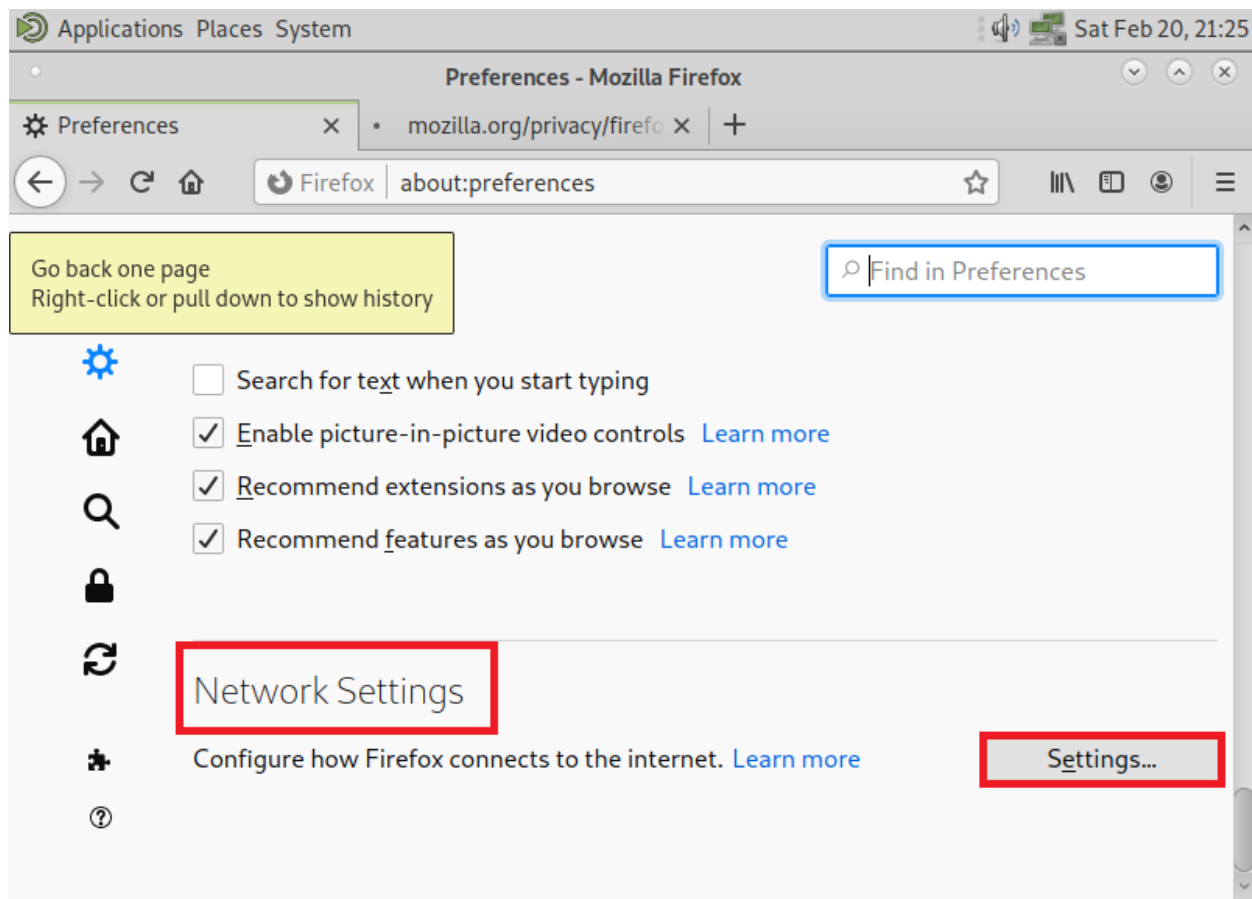
Откроем журнал для проверки результатов:

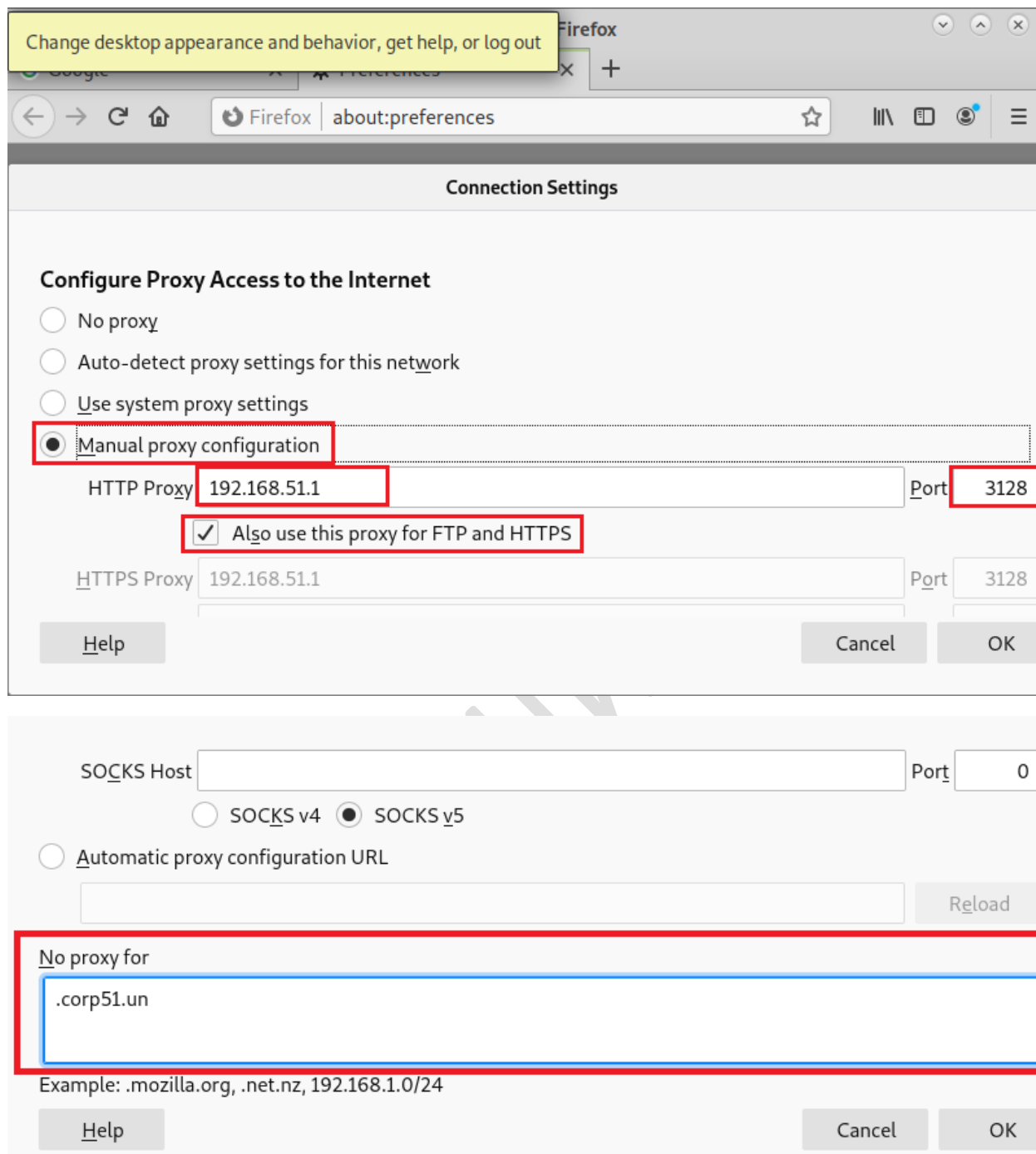
```
root@gate:~# tail -f /var/log/squid/access.log
```

Настроим клиент

На VM Client1:







На стороне Client1 в веб браузере откройте страницу yandex.ru

На Gate проверьте вывод в журнале access.log:

```
tail -f /var/log/squid/access.log
```

На стороне Client1 попытайтесь выполнить ping yandex.ru и проверьте результат

Включите маршрутизацию (перенаправление) на VM Gate:

```
sysctl net.ipv4.ip_forward=1
```

ОПЦИОНАЛЬНО. Упражнение 2. Настройка интеграции squid с HAVP/Clamd

На VM Gate:

```
apt install clamav-milter -y
```

```
freshclam
```

Ставим havp:

```
dpkg -i gcc-10-base_10.2.0-5ubuntu1~20.04_amd64.deb libgcc-s1_10.2.0-5ubuntu1~20.04_amd64.deb  
havp_0.93-2build1_amd64.deb
```

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

Раскомментируем и подправим:

....

```
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 -sG 'havp'
```

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

```
# service havp start
```

По умолчанию, HAVP слушает порт 8080:

```
# tail /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/squid/squid.conf
```

...

В секции # TAG: wais_relay_host

#332 строка

```
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
```

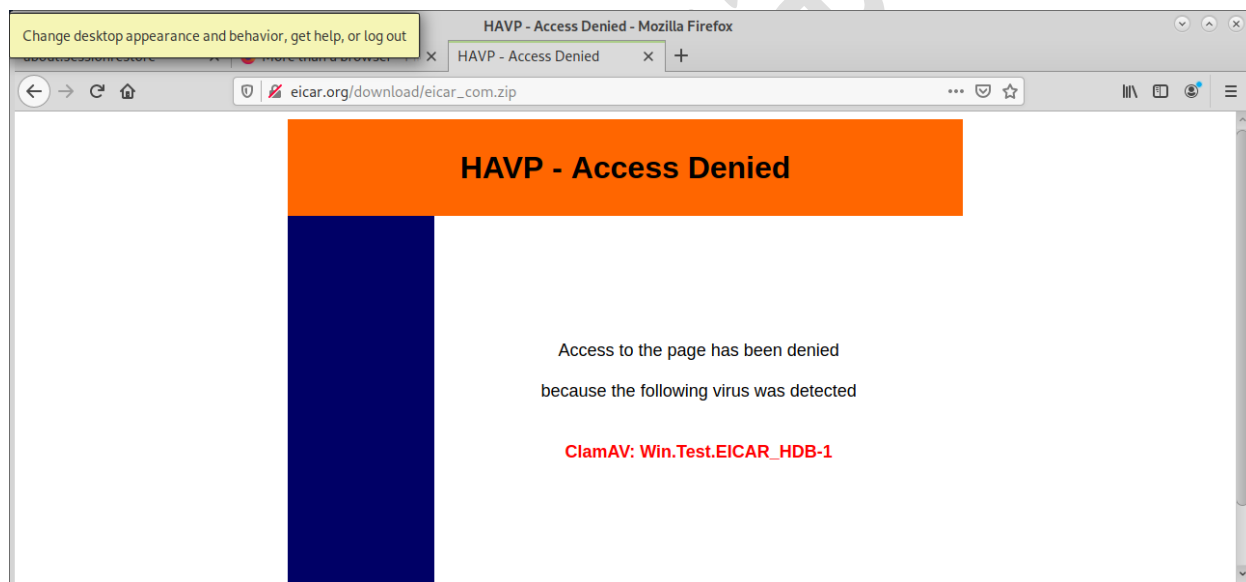
...

```
root@gate:~# systemctl restart squid
```

Используем для проверки тестовые файлы с eicar.org

В браузере на VM Client1 переходим на http://eicar.org/download/eicar_com.zip

Увидим сообщение havp.



Проверяем лог

```
gate# # tail /var/log/havp/error.log
```

```
user1@gate: ~  
root@gate:~# tail /var/log/havp/error.log  
01/03/2021 16:52:55 == Starting HAVP Version: 0.93  
01/03/2021 16:52:55 Running as user: havp, group: havp  
01/03/2021 16:52:55 --- Initializing ClamAV Library Scanner  
01/03/2021 16:52:55 ClamAV: Using database directory: /var/lib/clamav  
01/03/2021 16:53:06 ClamAV: Loaded 8504060 signatures (engine 0.102.4)  
01/03/2021 16:53:07 ClamAV Library Scanner passed EICAR virus test (Win.Test.EIC  
01/03/2021 16:53:07 --- All scanners initialized  
01/03/2021 16:53:07 Process ID: 2167  
root@gate:~#
```

#####

Глава 5. Сервис точного времени

Лабораторная работа. Развертывание NTP сервера

На VM Gate:

Узнать состояние timesyncd позволяет команда timedatectl

Сервера для синхронизации здесь:

```
/etc/systemd/timesyncd.conf
```

Прежде чем установить ntpd, отключите timesyncd:

```
sudo timedatectl set-ntp off
```

Узнать состояние timesyncd позволяет

```
systemctl status systemd-timesyncd
```

А также команда timedatectl:

```
timedatectl
```

В выводе должна быть строка:

```
Network time on: no
```

Теперь можно установить ntp:

Подготовим Gate к роли сервера точного времени:

```
# apt install ntp -y
```

Смотрим конфиг файл, затем сотрем его содержимое:

```
# cat > /etc/ntp.conf
```

И добавляем туда сервера для синхронизации:

```
server 0.ru.pool.ntp.org
```

```
server 1.ru.pool.ntp.org
```

```
server 2.ru.pool.ntp.org
```

```
server 3.ru.pool.ntp.org
```

Перезапустим NTP сервер:

```
systemctl restart ntp.service
```

Проверим его статус:

```
systemctl status ntp.service
```

Через 5-10 минут проверяем:

```
ntpq -pn
```

Проверка результатов синхронизации:

```
# ntptrace
```

Синхронизируем VM Server1 и Server2 с NTP сервером:

```
vi /etc/systemd/timesyncd.conf
```

```
...
```

```
[Time]
```

```
NTP=gate.corpX.un
```

```
...
```

```
systemctl restart systemd-timesyncd
```

Проверка результатов синхронизации:

```
# systemctl status systemd-timesyncd
```

```
# timedatectl status
```

Глава 6. Файловые сервисы**Лабораторная работа 1. Развертывание сервиса NFS****На Server1:**

```
apt install nfs-kernel-server -y
```

```
mkdir /var/nfs
```

```
chmod -R 777 /var/nfs
```

```
nano /etc/exports
```

```
/var/nfs 192.168.X.12(rw,sync,no_subtree_check,no_root_squash)
```

```
192.168.X.30(rw,sync,no_subtree_check,no_root_squash)
```

```
systemctl restart nfs-kernel-server
```

На BM Client1:

```
apt install nfs-common -y
```

```
systemctl status nfs-common
```

```
#####
```

Когда служба маскируется, ее файл в /lib/systemd/system является ссылкой на /dev/null.

```
systemctl enable nfs-common
```

Покажет символическую ссылку юнита, которую нужно удалить:

```
rm /lib/systemd/system/nfs-common.service
```

```
systemctl enable nfs-common
```

```
systemctl restart nfs-common
```

```
systemctl status nfs-common
```

```
#####
```

```
showmount -e server1
```

```
mkdir -p /var/share/nfs
```

```
nano /etc/fstab
```

```
192.168.51.10:/var/nfs /var/share/nfs nfs4 defaults,user,exec 0 0
```

```
mount /var/share/nfs
```

Лабораторная работа 2. Развертывание сервиса CIFS (пакет Samba)

Упражнение 1.

На VM Server1:

```
useradd smb_user -s /usr/sbin/nologin
```

```
apt install samba -y
```

```
cat > /etc/samba/smb.conf
```

```
[global]
```

```
unix charset = UTF-8
```

```
dos charset = cp866
```

```
workgroup = CORPX
```

```
#server role = standalone server
```

```
#passdb backend = tdbsam
```

```
#unix password sync = yes
```

```
security = user
```

```
map to guest = Bad User
```

```
[share]
```

```
path = /var/samba
```

```
guest ok = yes
```

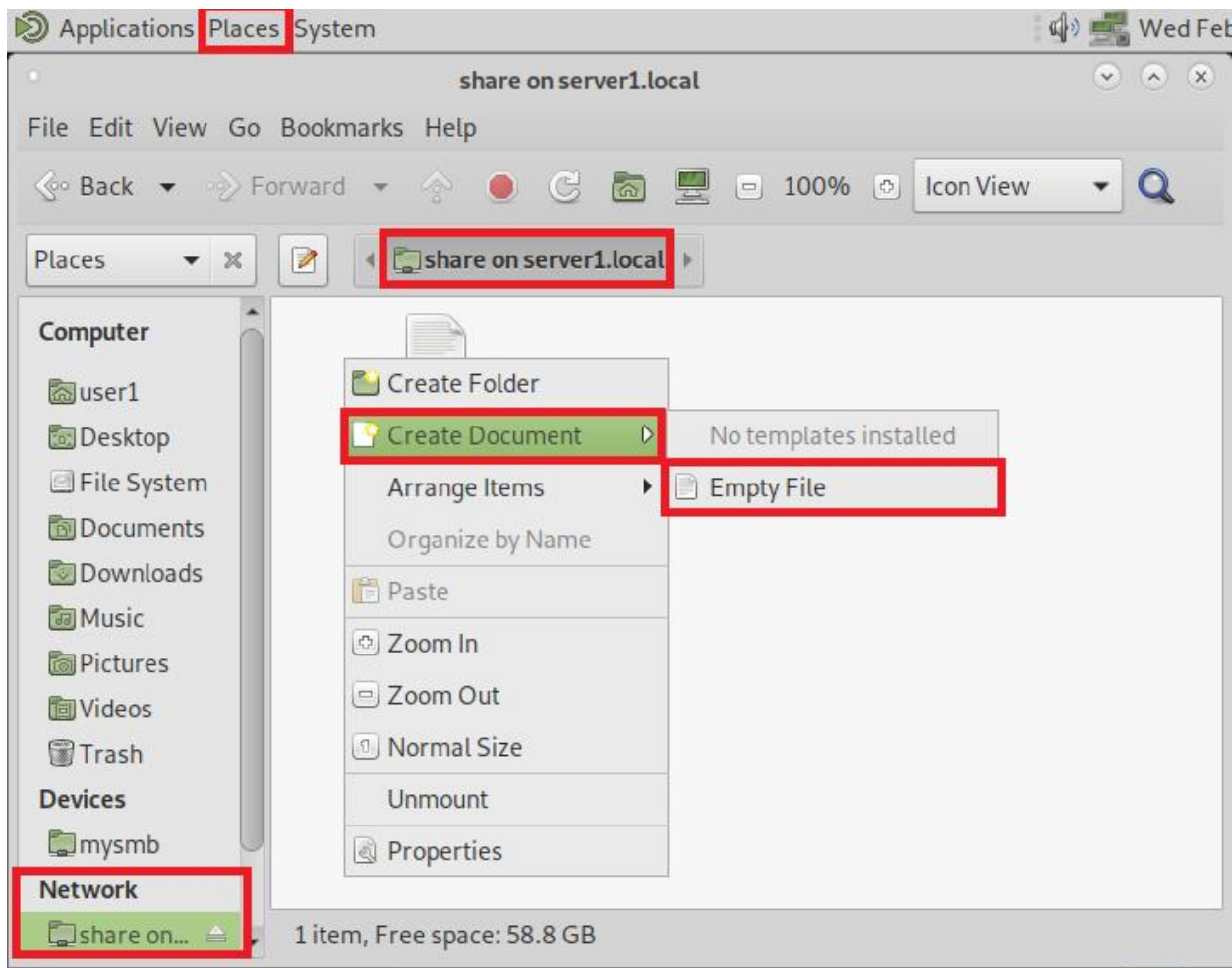
```
read only = no
```

```
force user = smb_user
```

```
# chown -R smb_user /var/samba
```

```
# chmod 755 /var/samba
```

проверяем доступ с клиента



проверяем разрешения созданного файла в файловой системе сервера:

```
root@server1:~# ls -l /var/samba
```

Упражнение 2. Управление доступом к общей папке:

```
groupadd -g 1500 allusers
```

```
gpsswd -M user1 allusers
```

```
apt install acl -y
```

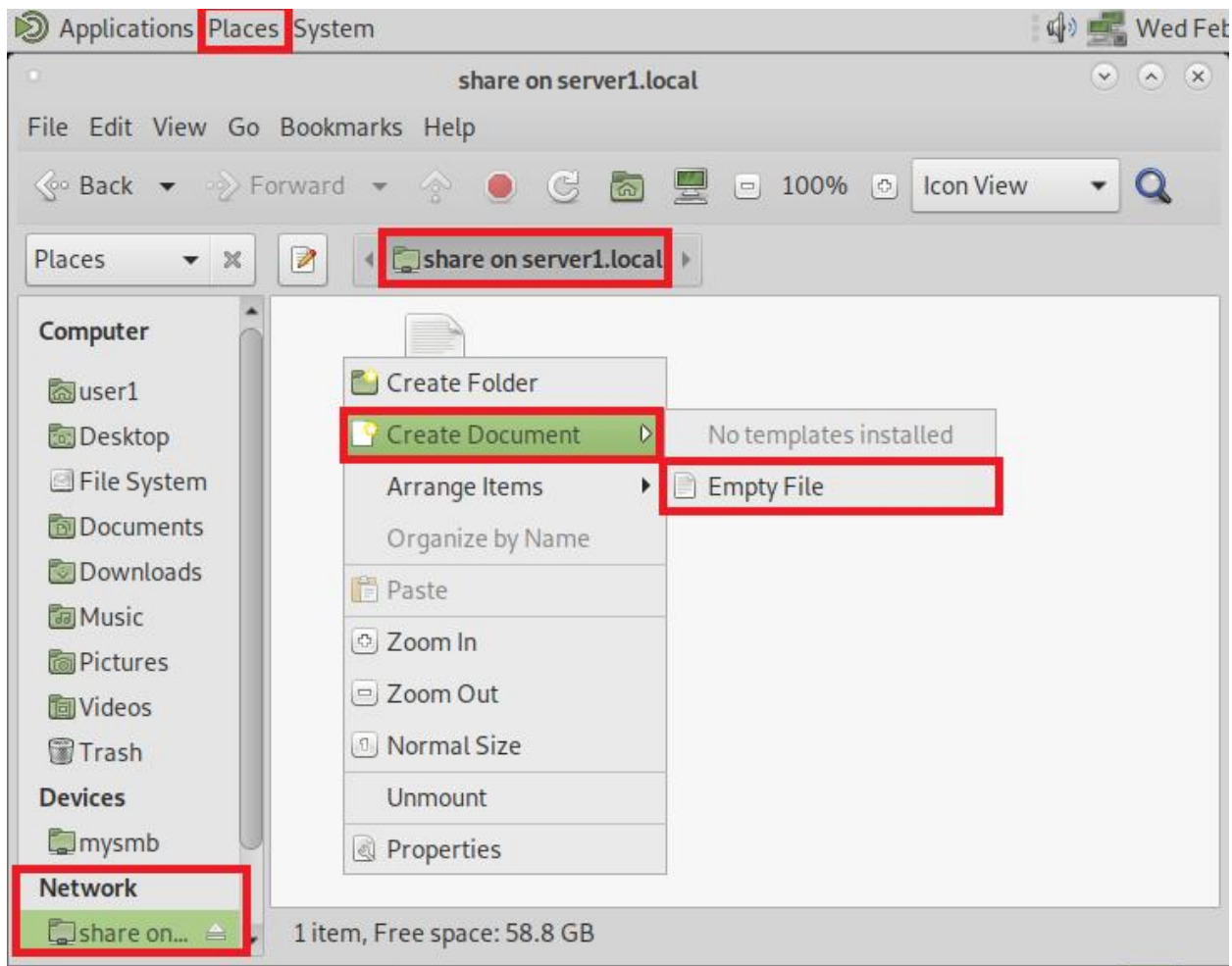
```
setfacl -m g:allusers:rwx /var/samba
```

```
smbpasswd -a user1
```

```
systemctl restart smbd.service
```

На BM Client1:

Подключитесь к общей папке на сервере. Создайте в ней файл.



Проверьте разрешения созданного файла в файловой системе сервера:

```
ls -l /var/samba
```

Упражнение 3. Предоставление доступа к домашним каталогам:

```
cat > /etc/samba/smb.conf
```

```
[global]
```

```
    unix charset = UTF-8
```

```
    dos charset = cp866
```

```
    workgroup = CORPX
```

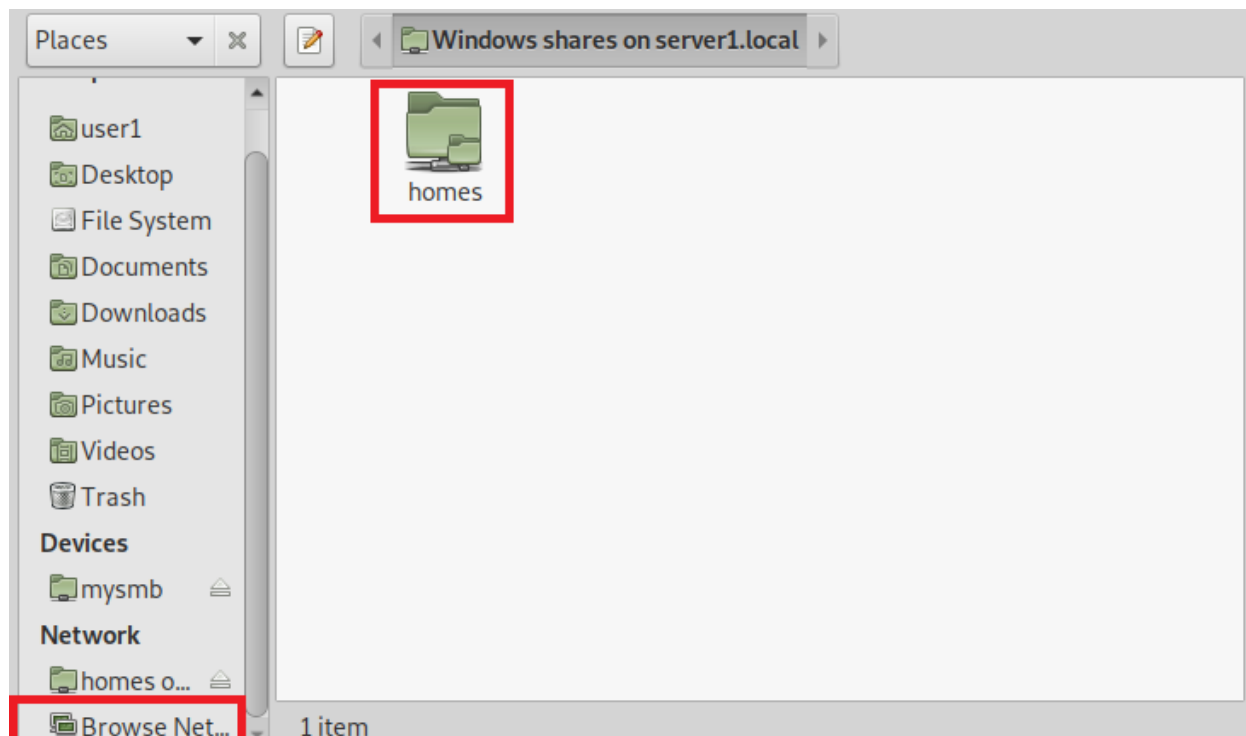
```
    security = user
```

```
[homes]
```

```
    comment = Home Directories
```


browseable = yes

read only = no



Упражнение 4. Использование mount.cifs с правами user1

На BM Client1:

```
apt install samba-client cifs-utils -y
```

```
nano /etc/fstab
```

```
//server1.corp51.un/homes /home/user1/mysmb cifs rw,user,user=user1,noauto 0 0
```

```
su - user1
```

```
mkdir mysmb
```

```
mount /home/user1/mysmb
```

Лабораторная работа 3. Развертывание сервиса FTP

На BM Server1:

```
apt install proftpd-basic -y
```

```
nano /etc/proftpd/proftpd.conf
```

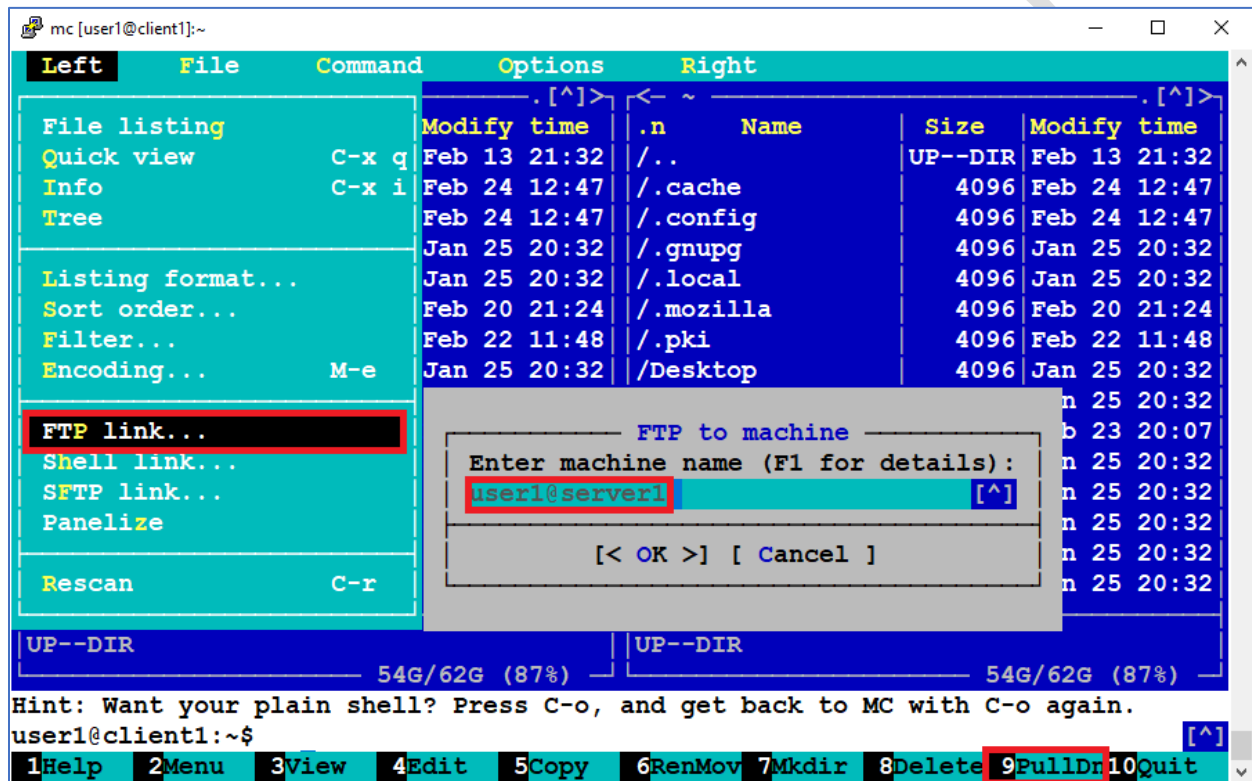
Раскомментировать строку

DefaultRoot ~

systemctl restart proftpd.service

На BM Client1:

apt install mc -y



Глава 7. Веб сервисы

Лабораторная работа. Развертывание HTTP сервера Apache

На VM Server1:

```
apt install apache2 -y
```

проверяем доступ с клиента, явно указывая протокол http:

<http://server1.corpX.un>



Настройка сайта по умолчанию

```
mkdir /var/www/html/img
```

```
cd /var/www/html/img
```

```
wget --no-check-certificate http://cdn.specialist.ru/Content/Image/Main/logo.gif
```

```
cd
```

```
cat<<EOF>/var/www/html/index.html
```

```
<html>
```

```
<h1>My Test Site 1</h1>
```

```
<img src=img/logo.gif >
```

```
</html>
```

```
EOF
```

проверяем доступ с клиента, явно указывая протокол http:

<http://server1.corpX.un>

На VM Server2:

```
apt install apache2 -y
```

```
mkdir /var/www/html/img && cd /var/www/html/img
```

```
wget --no-check-certificate http://cdn.specialist.ru/Content/Image/Main/logo.gif
```

```
cd
```

```
cat<<EOF>/var/www/html/index.html
```

```
<html>
```

```
<h1>My Test Site 2</h1>
```

```
<img src=img/logo.gif >
```

```
</html>
```

```
EOF
```

проверяем доступ с клиента, явно указывая протокол http:

<http://server2.corpX.un>

Лабораторная работа. Развертывание Web приложений

На VM Server1:

Упражнение 1. CGI интерфейс

```
a2enmod cgid
```

```
service apache2 restart
```

```
cd /usr/lib/cgi-bin/
```

```
nano test-cgi
```

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

```
echo Content-type: text/plain
```

```
echo
```

```
echo Hello $REMOTE_ADDR
```

```
echo You type: $QUERY_STRING
```

```
env
```

```
chmod 755 test-cgi
```

```
systemctl restart apache2
```

Проверяем с клиента, вводим в веб-браузере:

<http://server.corpX.un/cgi-bin/test-cgi>

http://server.corpX.un/cgi-bin/test-cgi?qwerty

```

← → ↻ http://server1.corp51.un/cgi-bin/test-cgi?qwerty

Hello 192.168.51.5
You type: qwerty
GATEWAY_INTERFACE=CGI/1.1
REMOTE_ADDR=192.168.51.5
QUERY_STRING=qwerty
HTTP_USER_AGENT=Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.182 Safari/537.36
DOCUMENT_ROOT=/var/www/html
REMOTE_PORT=63459
HTTP_UPGRADE_INSECURE_REQUESTS=1
HTTP_ACCEPT=text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
SERVER_SIGNATURE=<address>Apache/2.4.38 (Debian) Server at 192.168.51.10 Port 80</address>

CONTEXT_DOCUMENT_ROOT=/usr/lib/cgi-bin/
SCRIPT_FILENAME=/usr/lib/cgi-bin/test-cgi
HTTP_HOST=192.168.51.10
REQUEST_URI=/cgi-bin/test-cgi?qwerty
SERVER_SOFTWARE=Apache/2.4.38 (Debian)
HTTP_CONNECTION=keep-alive
REQUEST_SCHEME=http
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HTTP_ACCEPT_LANGUAGE=ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
SERVER_PROTOCOL=HTTP/1.1
HTTP_ACCEPT_ENCODING=gzip, deflate
REQUEST_METHOD=GET
SERVER_ADDR=192.168.51.10
SERVER_ADMIN=webmaster@localhost
CONTEXT_PREFIX=/cgi-bin/
PWD=/usr/lib/cgi-bin
SERVER_PORT=80
SCRIPT_NAME=/cgi-bin/test-cgi
SERVER_NAME=192.168.51.10

```

Упражнение 2. Модуль PHP

```
apt install libapache2-mod-php -y
```

```
apache2ctl -M
```

Создаем конфиг публикации сайта

```
nano /etc/apache2/sites-available/mysite.conf
```

```

<VirtualHost *:80>
    ServerName mysite.corp51.un
    DocumentRoot /var/www/mysite
<Directory /var/www/mysite>
    DirectoryIndex index.php
    AllowOverride All
</Directory>
</VirtualHost>

```

Создаем сайт на php

```
mkdir /var/www/mysite
```

```
nano /var/www/mysite/index.php
```

```
<?php
    phpinfo();
?>
```

Включаем доступ к сайту

```
a2ensite mysite
```

```
systemctl reload apache2
```

Настраиваем разрешение имен для доступа к сайту:

```
nano /etc/bind/corpX.un
```

```
...
```

```
mysite CNAME server1
```

```
rndc reload
```

Проверяем с VM Client1

<http://mysite.corpX.un>

Лабораторная работа. Развертывание и настройка реверсного прокси на Nginx

Упражнение 1. Установка и настройка Nginx в качестве реверсивного прокси.

На VM Gate:

```
apt install nginx -y
```

```
nano /etc/nginx/sites-available/default
```

```
# Default server configuration
```

```
#
```

```
server {
```

```

listen 80 default_server;

listen [::]:80 default_server;

# .....

server_name _;


location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    try_files $uri $uri/ =404;
    proxy_pass http://192.168.X.10; # далее заменяется на proxy_pass http://backend/;
    proxy_set_header Host $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Real-IP $remote_addr;
}

```

```
root@gate:~# service nginx restart
```

Проверяем с VM Client2:

<http://gate.corpX.un>

Упражнение 2. Настройка балансировки нагрузки

В начало /etc/nginx/sites-available/default бэкэнды добавляются директивой **upstream**:

```

nano /etc/nginx/sites-available/default

upstream backend {
    server 192.168.X.10:80;
    server 192.168.X.11:80;
}

```

Например:

```

GNU nano 3.2 /etc/nginx/sites-available/default

upstream backend {
    server 192.168.51.10:80;
    server 192.168.51.11:80;
}
# Default server configuration
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    # SSL configuration

```

server_name _;

location / {

First attempt to serve request as file, then

as directory, then fall back to displaying a 404.

try_files \$uri \$uri/ =404;

proxy_pass http://backend/;

proxy_set_header Host \$host;

proxy_set_header X-Forwarded-For \$proxy_add_x_forwarded_for;

proxy_set_header X-Real-IP \$remote_addr;

}

systemctl restart nginx.service

Прим. Сборка с поддержкой модуля stream для TCP/UDP-проксирования и балансировки. По умолчанию модуль не собирается.

cd /usr/local/src/

wget http://nginx.org/download/nginx-1.14.2.tar.gz

openssl -v

openssl

wget https://www.openssl.org/source/old/1.1.0/openssl-1.1.1d.tar.gz

wget https://www.openssl.org/source/old/1.1.1/openssl-1.1.1d.tar.gz


```
git clone git://github.com/vozlt/nginx-module-vts.git
```

```
apt install git
```

```
git clone git://github.com/vozlt/nginx-module-vts.git
```

```
tar xvfz nginx-1.14.2.tar.gz
```

```
tar xzvf openssl-1.1.1d.tar.gz
```

```
cd nginx-1.14.2/
```

```
./configure --with-stream
```

```
make
```

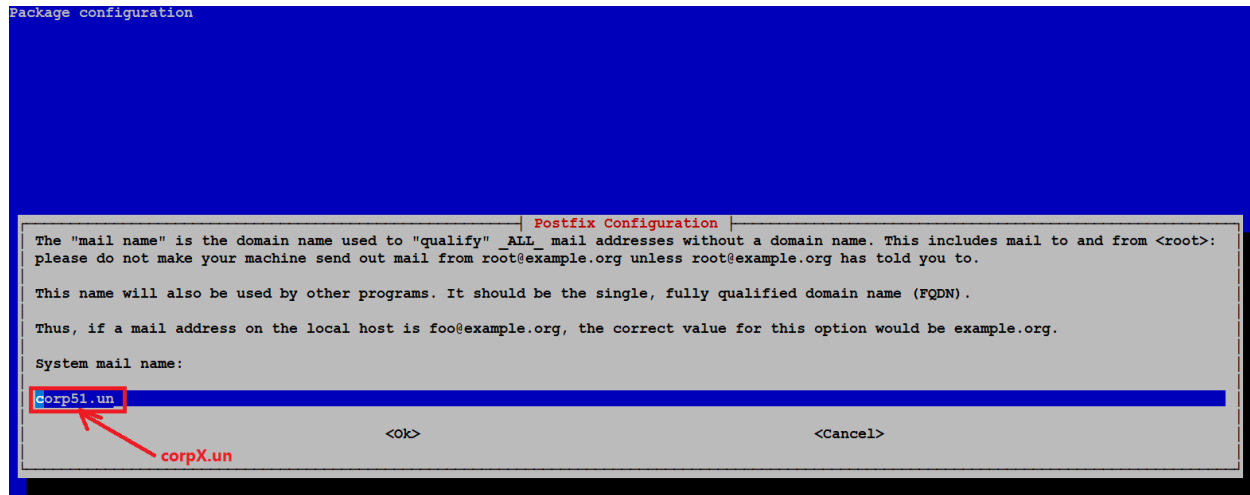
```
make install
```

Глава 8. Организация сервиса электронной почты

Лабораторная работа 1. Развертывание и настройка сервиса SMTP (пакет Postfix)

Упражнение 1. Развертывание и настройка Postfix

```
apt install postfix mailutils -y
```



Открываем лог на VM Server1:

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

Проверяем отправку сообщения с VM Gate:

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

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

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

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

```
DATA
```

```
From: root@gate.isp.un
```

```
Subject: SMTP testX
```

```
Test message
```

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

```
Quit
```

Упражнение 2. Использование почтовых псевдонимов

Добавим пользователя user2:

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

```
passwd user2
```

Псевдонимы настраиваются в:

```
nano /etc/aliases
```

Почта на псевдоним support представляет собой групповой адрес:

Добавляем в /etc/aliases строку:

```
support: user1, user2
```

Для вступления в силу алиасов набрать команду:

```
newaliases
```

```
mail support
```

Проверьте результат, используя клиент mail под user и user2

Упражнение 3. Конфигурирование поддержки почтовых доменов

Настроим поддержку почтового домена corpX.un

В файле /etc/postfix/main.cf перейдем на строку mydestination и добавим имя домена:

```
mydestination = $myhostname, corpX.un, server1.corpX.un, localhost.corpX.un, localhost, compX.un
```

```
systemctl reload postfix
```

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

При этом у нас в corpX есть почтовый ящик info@corpX.un привязанный к user1.

Но поскольку задача стоит при совпадении пользовательских имен (info) не сливать почту из разных доменов в один ящик: добавляем в конце файла /etc/postfix/main.cf строку:

```
cat >> /etc/postfix/main.cf
```

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

и создаем файл, на который в данной строке сослались:

```
nano /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:

```
systemctl reload postfix
```

Проверяем:

```
mail info@compX.un
```

```
su - user2
```

```
mail
```

Лабораторная работа 2. Развертывание и настройка сервиса IMAP (пакет Dovecot)**Упражнение 1. Установка и базовая настройка dovecot-imapd**

```
apt -y install dovecot-imapd
```

```
nano /etc/dovecot/conf.d/10-auth.conf
```

```
disable_plaintext_auth = no
```

```
...
```

```
nano /etc/dovecot/conf.d/10-mail.conf
```

Убедиться, что

```
mail_privileged_group = mail
```

Проверяем корректность:

```
dovecot -n
```

```
systemctl restart dovecot
```

Настройка МТА на перенаправление почты от клиентов задается в файле

/etc/postfix/main.cf строкой mynetworks:

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

```
systemctl reload postfix
```

По умолчанию, Postfix пересылает почту от клиентов, находящихся в авторизованной части сети, на любой адрес. Авторизованные сети определяет конфигурационный параметр mynetworks. Поведение по умолчанию - авторизовать всех клиентов IP подсетей, к которым подсоединена машина.

Упражнение 1. Установка и настройка почтового клиента mozilla thunderbird

Выполняется на VM Client1.

```
apt install thunderbird -y
```

Войдите в графический интерфейс как user1 и настройте thunderbird

Set Up Your Existing Email Address

Set Up Your Existing Email Address
Use your current email address

Your name: ⓘ

Email address: ⓘ

Password: ⓘ

☐ Remember password

Set Up Your Existing Email Address

Use your current email address

Your name: ⓘ

Email address: ⓘ

Password: ⓘ

☐ Remember password

	INCOMING	OUTGOING
Protocol:	<input type="text" value="IMAP"/> ▼	<input type="text" value="SMTP"/>
Server:	<input type="text" value="corp51.un"/>	<input type="text" value="corp51.un"/> ▼
Port:	<input type="text" value="143"/> ▼	<input type="text" value="25"/> ▼
SSL:	<input type="text" value="None"/> ▼	<input type="text" value="None"/> ▼
Authentication:	<input type="text" value="Normal password"/> ▼	<input type="text" value="No authentication"/> ▼
Username:	<input type="text" value="user1"/>	<input type="text" value="user1"/>

[Advanced config](#)

Cancel

Re-test

Done

Лабораторная работа 3. Организация Web интерфейса к почтовому серверу (пакет roundcube)

apt -y install mariadb-server

mysql -u root -p

create database roundcube;

grant all privileges on roundcube.* to roundcube@'localhost' identified by 'password';

exit

apt -y install roundcube roundcube-mysql

cd /usr/share/dbconfig-common/data/roundcube/install

mysql -u roundcube -D roundcube -p < mysql

cd

nano /etc/roundcube/config.inc.php

строка 35:

\$config['default_host'] = 'localhost';

строка 47:

\$config['smtp_server'] = 'localhost';

строка 51:

\$config['smtp_port'] = 25;

строки 55, 59, 60 закомментировать:

\$config['smtp_user'] =

\$config['smtp_password'] =

\$config['smtp_auth_type'] =

nano /etc/apache2/conf-enabled/roundcube.conf

Раскомментировать строку 3:

Alias /roundcube /var/lib/roundcube

systemctl restart apache2

На VM Client1 в браузере подключиться по <http://server1.corpX.un/roundcube> как user2

Лабораторная работа 9. Трансляция адресов и варианты NAT**Упражнение 1. Использование NAT для подключения к сети провайдера**

1. Проверьте маршрутизацию с VM Server1
2. `root@gate:~# iptables-save`

```
root@gate:~# iptables-save -t nat
```

На VM Gate:

3. `iptables -t nat -A POSTROUTING -o enp0s8 -s 192.168.X.0/24 -j MASQUERADE`
4. `iptables-save`
5. Вновь проверьте маршрутизацию с VM Server1

```
# traceroute ya.ru
```

Упражнение 2. Настройка пакетных фильтров для защиты сети.

1. Проверьте доступность ресурсов внутренней сети с VM Client2:

```
ping 192.168.X.10
```

2. Проверьте доступность внешних ресурсов из локальной сети с VM Server1:

```
ping ya.ru
```

На VM Gate:

3. **Установите модуль conntrack**

```
apt install conntrack
```

4. **Добавьте правила фильтрации:**

```
iptables -A FORWARD -i enp0s3 -s 192.168.X.0/24 -j ACCEPT
```

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

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

5. Повторите шаги п. 1 – 2

Упражнение 3. Открываем доступ к ресурсам:

Проверьте доступность ресурсов внутренней сети с VM Client2:

```
telnet 192.168.X.10 22
```

На BM Gate:

Создайте файл сценария для добавления правил доступа к ресурсам Server1

```
cat<<EOF> firewall.sh
```

```
iptables --flush
```

```
iptables -A FORWARD -i enp0s8 -p tcp -d 192.168.X.10 --dport 22 -j ACCEPT
```

```
iptables -A FORWARD -i enp0s8 -p tcp -d 192.168.X.10 --dport 53 -j ACCEPT
```

```
iptables -A FORWARD -i enp0s8 -p udp -d 192.168.X.10 --dport 53 -j ACCEPT
```

```
iptables -A FORWARD -i enp0s8 -p tcp -d 192.168.X.10 --dport 25 -j ACCEPT
```

```
iptables -A FORWARD -i enp0s8 -p tcp -d 192.168.X.10 --dport 80 -j ACCEPT
```

```
iptables -A FORWARD -i enp0s8 -p tcp -d 192.168.X.10 --dport 143 -j ACCEPT
```

```
iptables -A FORWARD -i enp0s3 -s 192.168.X.0/24 -j ACCEPT
```

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

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

```
EOF
```

Правим файл:

```
sed -i 's/X/HOMEP_СТЕНДА/g' firewall.sh
```

запускаем:

```
sh firewall.sh
```

Проверьте снова доступность ресурсов внутренней сети с VM Client2:

```
telnet 192.168.X.10 22
```

Сохраните содержимое таблиц nat и filter в файле для восстановления после перезагрузки.

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

Проверьте

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

```
iptables-save
```

Затем восстановите таблицы командой `iptables-restore /etc/iptables.rules`

Проверьте

`iptables-save`

Упражнение 4. Настраиваем DNAT

```
cat<<EOF> nat.sh
```

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

```
iptables -t nat -A POSTROUTING -o enp0s8 -s 192.168.X.0/24 -j SNAT --to-source  
<адрес_внешнего_интерфейса_gate>
```

```
iptables -t nat -A PREROUTING -i enp0s8 --destination <адрес_внешнего_интерфейса_gate> -p tcp --  
dport 2222 -j DNAT --to-destination 192.168.X.10:22
```

```
#iptables -t nat -A PREROUTING -i enp0s8 --destination <адрес_внешнего_интерфейса_gate> -p tcp --  
dport 25 -j DNAT --to-destination 192.168.X.10:25
```

```
iptables -t nat -A PREROUTING -i enp0s8 --destination <адрес_внешнего_интерфейса_gate> -p tcp --  
dport 53 -j DNAT --to-destination 192.168.X.10:53
```

```
iptables -t nat -A PREROUTING -i enp0s8 --destination <адрес_внешнего_интерфейса_gate> -p udp --  
dport 53 -j DNAT --to-destination 192.168.X.10:53
```

```
iptables -t nat -A PREROUTING -i enp0s8 --destination <адрес_внешнего_интерфейса_gate> -p tcp --  
dport 80 -j DNAT --to-destination 192.168.X.10:80
```

```
iptables -t nat -A PREROUTING -i enp0s8 --destination <адрес_внешнего_интерфейса_gate> -p tcp --  
dport 143 -j DNAT --to-destination 192.168.X.10:143
```

```
conntrack -F
```

```
EOF
```

Правим файл:

```
sed -i 's/X/НОМЕР_СТЕНДА/g' nat.sh && sed -i  
's/<адрес_внешнего_интерфейса_gate>/ВНЕШНИЙ_АДРЕС_GATE/g' nat.sh
```

Где X = вашему номеру стенда, а ВНЕШНИЙ_АДРЕС_GATE = адресу на внешнем интерфейсе VM Gate.

Например, в случае если X = 15 и внешний интерфейс имеет адрес 192.168.0.55:

```
sed -i 's/X/51/g' nat.sh && sed -i 's/<адрес_внешнего_интерфейса_gate>/192.168.0.55/g' nat.sh
```

Загрузите правила скриптом:

```
sh nat.sh
```

Проверьте доступность ресурсов внутренней сети с VM Client2:

```
ssh адрес_внешнего_интерфейса_gate
```

```
ssh -p 2222 адрес_внешнего_интерфейса_gate
```

Например:

```
ssh 192.168.0.55
```

```
ssh -p 2222 192.168.0.55
```

Обновите файл /etc/iptables.rules:

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

Для восстановления таблиц после перезагрузки в файл /etc/network/interfaces нужно добавить строку **pre-up iptables-restore /etc/iptables.rules** после блока настройки внешнего интерфейса enp0s8

Например:

```
auto enp0s8
```

```
iface enp0s8 inet static
```

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

Для проверки выполните сброс таблиц

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

```
iptables-save
```

Перезапустите интерфейс:

```
ifdown enp0s8 && ifup enp0s8
```

Проверьте содержимое таблиц

```
iptables-save
```

Упражнение 5. Настраиваем редирект портов (на примере транспарентного прокси)

На VM Gate добавьте правило редиректа портов:

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

Настройте squid на режим перехватывающего прокси

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

...

```
# Squid normally listens to port 3128
```

```
http_port 3130
```

```
http_port 3128 intercept
```

```
# squid -k reconfigure
```

```
gate:~# tail -fn 0 /var/log/squid/access.log
```

На VM Client1 в веб-браузере убираем настройку на прокси.

Лабораторная работа 10-А. Реализация защиты от вирусов (Clamav)

Ставим Front-End Server который будет принимать сообщения снаружи.

Эта роль будет на Gate.

```
apt install postfix mailutils -y
```

Все настройки принимаем по умолчанию.

Проверяем на VM Client2:

```
telnet <внешний ip gate> 25
```

Нужно будет модифицировать

1. правило iptables -A FORWARD -i enp0s8 -p tcp -d 192.168.X.10 --dport 25 -j ACCEPT
2. Настройки DNS – настроить MX
3. на Gate настроить relay + антиспам и антивирус пакеты

На Gate:

```
# vi firewall.sh
```

В правиле iptables -A FORWARD -i enp0s8 -p tcp -d 192.168.X.10 --dport 25 -j ACCEPT меняем на REJECT

```
# sh firewall.sh
```

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

Проверяем на VM Client2:

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

и видим в сообщении ключевую фразу **Connection refused** благодаря опции REJECT

На Server:

```
nano /etc/bind/corpX.un
```

Раскомментируем MX записи:

```
$TTL 3h
```

```
@ IN SOA server1. root.server1 1 1d 12h 1w 3h
```

```
NS server1.
```

```
A 192.168.51.10
```

```
MX 1 server1
```

```
MX 2 gate
```

```
gate IN A 192.168.51.1
```

```
server1 IN A 192.168.51.10
```

```
server2 IN A 192.168.51.11
```

```
rndc reload
```

На BM Gate в конфиге postfix

```
nano /etc/postfix/main.cf
```

пропишем в конец файла:

...

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

В данном параметре перечислены домены, на которые разрешена пересылка корреспонденции от посторонних клиентов

Настройка пересылки почты в Postfix

настроим сервер SMTP (relayhost), через который будут отправляться все сообщения.

```
relayhost = server1.corpX.un
```

перезапускаем:

```
service postfix reload
```

Открываем журнал на Gate

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

Открываем журнал на Server1

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

Проверяем с **BM Client2**:

```
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

В журнале на Gate обратите внимание на строку

relay=server1.corpX.un[192.168.X.10]:25

tail -fn0 /var/log/mail.log

Feb 25 11:41:48 gate postfix/relay/smtp[1598]: 80113281020: to=<user1@corp51.un>
, relay=server1.corp51.un[192.168.51.10]:25, delay=67, delays=66/0.01/0.16/0.04,
dsn=2.0.0, status=sent (250 2.0.0 Ok: queued as 385971E2288)

В журнале на Server1 обратите внимание на строку

status=sent (delivered to mailbox)

tail -fn0 /var/log/mail.log

Feb 25 11:41:47 server1 postfix/smtpd[1272]: disconnect from gate.corp51.un[192.168.51.1] ehlo=1
mail=1 rcpt=1 data=1 quit=1 commands=5

Feb 25 11:41:47 server1 postfix/local[1277]: 385971E2288: to=<user1@corp51.un>, relay=local,
delay=0.09, delays=0.03/0.03/0/0.03, dsn=2.0.0, **status=sent (delivered to mailbox)**

Упражнение 3. Защита почты от вирусов с использованием clamav

Установка clamav с milter интерфейсом

На BM Gate:

apt install clamav-milter -y

Обновления сигнатур скачиваются freshclam (ps ax | grep clam)и складываются в /var/lib/clamav
можно посмотреть ls /var/lib/clamav

freshclam

Ждем прокачку сигнатур, затем запустите **clamav-daemon**:

systemctl start clamav-daemon

Настройка интеграции с Clamav-milter

в `/etc/clamav/clamav-milter.conf` указать где создавать сокет:

```
nano /etc/clamav/clamav-milter.conf
```

Замените строку

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

строкой:

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

Перезапустите clamav-milter:

```
systemctl restart clamav-milter
```

Появится сокет `clamav-milter.ctl`:

```
ls -l /var/spool/postfix/clamav/clamav-milter.ctl
```

В файле

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

Укажем, что:

1. каждое письмо полученное по протоколу `smtp` передавать на сокет в `/var/spool/postfix/clamav` (после `chroot` - относительно `/clamav`, кот. далеко не в корне, а в `/var/spool/postfix/clamav`)

2. если фильтр недоступен, то почту принимать

...

```
milter_default_action = accept
```

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

Рестартуем:

```
systemctl restart postfix
```

Проверяем работу:

Запустим лог:

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

На VM Client2:

В текстовой консоли (Ctrl + Alt + F1)

```
cat > /etc/resolv.conf
```

```
domain Home
```

```
search Home
```

```
nameserver <Внешний адрес Gate>
```

```
apt update && apt install thunderbird -y
```

После установки войдите в графический интерфейс и войдите в клиент thunderbird. При первом запуске клиента выполните настройку для user1@gate.corpX.un:

Set Up Your Existing Email Address

Set Up Your Existing Email Address
Use your current email address

Your name: user1 ⓘ

Email address: user1@gate.corp51.un ⓘ

Password: ●●● ⓘ

☒ Remember password

✓ Configuration found by trying common server names

Protocol:
Incoming: IMAP gate.corp51.un STARTTLS
Outgoing: SMTP gate.corp51.un STARTTLS
Username: user1

Cancel **Configure manually...** Done

Set Up Your Existing Email Address

Set Up Your Existing Email Address
Use your current email address

Your name:

user1

Email address:

user1@gate.corp51.un

Password:

•••

☒ Remember password

INCOMING

OUTGOING

Protocol:

IMAP

SMTP

Server:

gate.corp51.un

gate.corp51.un

Port:

143

25

SSL:

None

None

Authentication:

Normal password

Normal password

Username:

user1

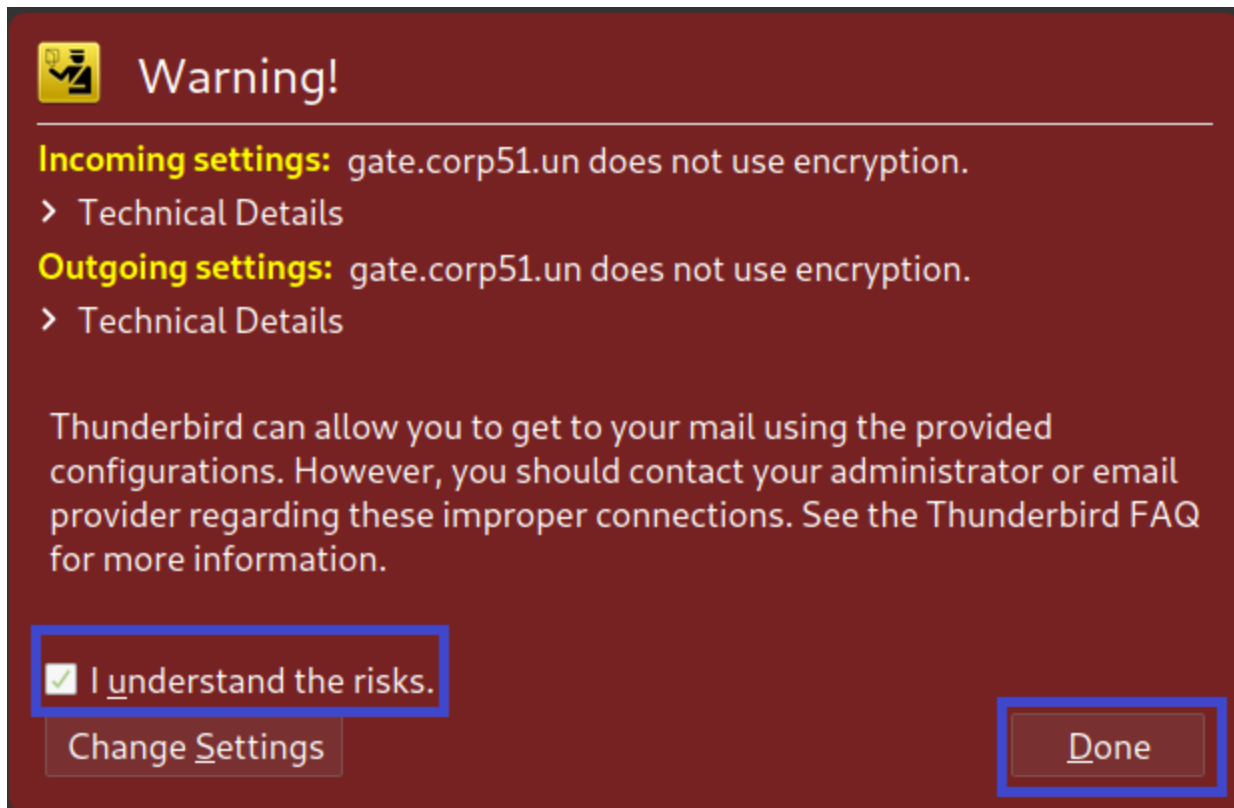
user1

[Advanced config](#)

Cancel

Re-test

Done



В браузере введите http://eicar.org/eicar_com.zip для скачивания файла тестирования антивирусной защиты.

Отправьте письмо с вложением eicar_com.zip на адрес use2@corpX.un

Посмотрите вывод `tail -f /var/log/clamav/clamav.log` на BM Gate.

```
Thu Feb 25 13:21:51 2021 -> fd[10]: Win.Test.EICAR_HDB-1(44d88612fea8a8f36de82e1278abb02f:68)
FOUND
```

Лабораторная работа 10-В. Реализация защиты почты от спама (пакеты SpamAssassin, postgrey)

Упражнение 1. Установка и настройка пакета Spam Assassin

На BM Gate

```
apt -y install spamassassin
```

Настраиваем:

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

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

```
# cat > 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
```

#--норм 5 (вероятность что оно спам)

```
# trusted_networks 192.168.X # --не исп RBL
```

```
# add_header all Report _REPORT_
```

```
# score RCVD_IN_BL_SPAMCOP_NET 10.0
```

Запускаем обновление сигнатур:

```
# # sa-update
```

Запускаем:

```
# systemctl start spamassassin
```

Упражнение 2. Подключение SpamAssassin через milter интерфейс

На BM Gate

SpamAssassin Подключим к Postfix.

```
# apt install spamass-milter -y
```

Правим конфиг:

```
nano /etc/default/spamass-milter
```

Раскомментируем 3 строки:

```
#####
```

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

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

```
SOCKETMODE="0660"
```

```
#####
```

Перезапустим:

```
systemctl restart spamass-milter
```

или

```
# service spamass-milter restart
```

Подправим конфиг postfix'a:

```
nano /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:

```
service postfix restart
```

Проверяем:

Запускаем на Gate лог:

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

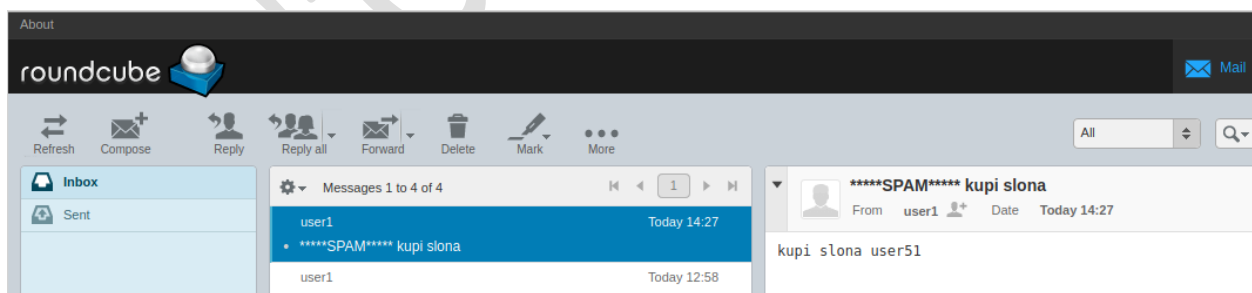
На VM Client1 откройте почтовый клиент как user1.

Отправьте сообщение для user2@corpX.un

С темой `kupi slona` и содержанием `kupi slona userX`

Можно видеть, что письмо доставлено.

На VM Client1 откройте почтовый клиент как user2.



Обратите внимание на измененный заголовок.

Упражнение 3. Установка и настройка пакета postgrey

На VM Gate

```
apt -y install postgrey
```

Ознакомьтесь с настройками файла конфигурации по умолчанию:

```
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:

```
nano /etc/postfix/main.cf
```

Прописываем в конец файла указанную ниже строку:

```
smtpd_recipient_restrictions = permit_mynetworks, reject_unauth_destination, check_policy_service
inet:127.0.0.1:10023
```

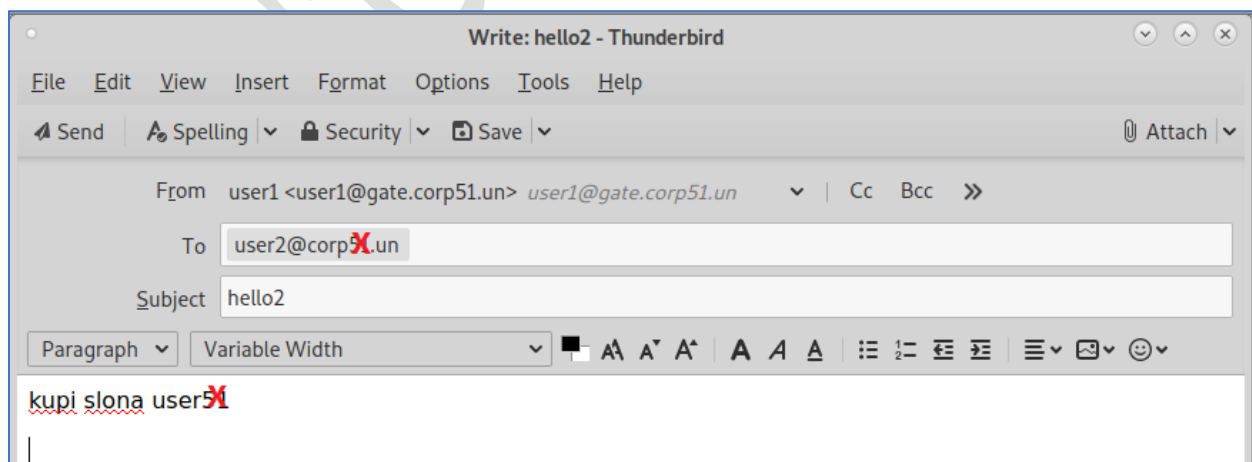
Перезапустим postfix:

```
systemctl restart postfix
```

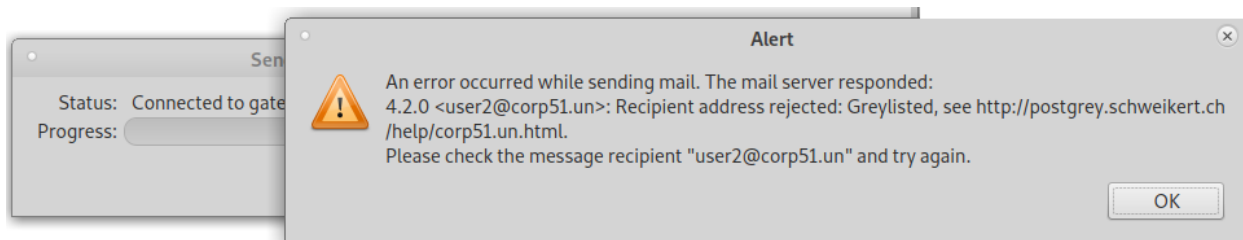
Запускаем на Gate лог:

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

Пишем письмо с VM Client2



И получаем сообщение при отправке:



Смотрим лог на Gate:

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

```
Feb 25 14:56:39 gate postfix/smtpd[7235]: NOQUEUE: reject: RCPT from unknown[192.168.0.109]: 450
4.2.0 <user2@corp51.un>: Recipient address rejected: Greylisted, see
http://postgrey.schweikert.ch/help/corp51.un.html; from=<user1@gate.corp51.un>
to=<user2@corp51.un> proto=ESMTP helo=<[192.168.0.109]>
```

Через 5 минут можно повторить попытку отправки письма с VM Client2 и оно будет доставлено, при условии, что адрес отправителя не изменился.

УЦ "Специалист"

Лабораторная работа 11. Использование сервиса RADIUS для управления доступом в Internet**На VM Server****Упражнение 1. Установка и настройка RADIUS**

```
apt install freeradius -y
```

Настройка файлов производится в директории /etc/freeradius/3.0:

```
cd /etc/freeradius/3.0/
```

```
cat >> clients.conf
```

```
client gate.corp51.un {  
    secret      = secret  
    shortname   = gate  
}
```

```
cat >> users
```

```
user1 Cleartext-Password := "rpassword1"
```

```
vim radiusd.conf
```

В блоке настроить auth = yes:

```
log {  
    ...  
    auth = yes  
    ...
```

```
service freeradius restart
```

На BM Gate

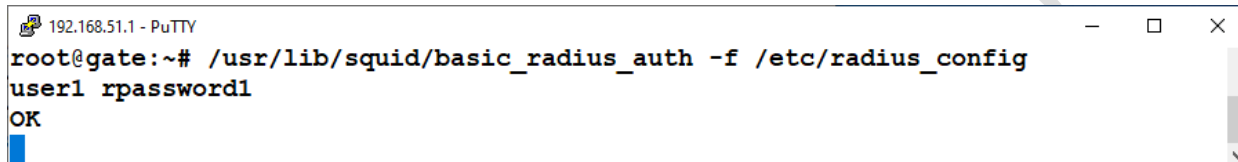
Определяем подключение к RADIUS серверу:

```
vim /etc/radius_config
```

```
server 192.168.51.10
```

```
secret secret
```

Проверяем работу хелпера



```
192.168.51.1 - PuTTY
root@gate:~# /usr/lib/squid/basic_radius_auth -f /etc/radius_config
user1 rpassword1
OK
```

```
/usr/lib/squid/basic_radius_auth -f /etc/radius_config
```

или

```
/usr/lib/squid/basic_radius_auth -h 192.168.51.10 -w secret
```

Настраиваем squid:

```
vim /etc/squid/squid.conf
```

В конце блока # OPTIONS FOR AUTHENTICATION:

```
auth_param basic program /usr/lib/squid/basic_radius_auth -f /etc/radius_config
```

```
auth_param basic children 5
```

```
auth_param basic realm CorpX Proxy Server
```

```
auth_param basic credentialsttl 5 minute
```

```
auth_param basic casesensitive off
```

В блоке

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
```

```
#
```

```
include /etc/squid/conf.d/*
```

```
acl radius-auth proxy_auth REQUIRED
```

```
http_access allow radius-auth
```

```
squid -k check
```

```
squid -k reconfigure
```

Откройте журнал:

```
tail -fn0 /var/log/squid/access.log | grep user1
```

```
root@gate:~# tail -fn0 /var/log/squid/access.log | grep user1
```

```
1617546453.648 126 192.168.51.30 TCP_MISS/200 327 GET  
http://detectportal.firefox.com/success.txt user1 HIER_DIRECT/34.107.221.82 text/plain
```

```
1617546453.678 21 192.168.51.30 TCP_MISS/200 327 GET  
http://detectportal.firefox.com/success.txt? user1 HIER_DIRECT/34.107.221.82 text/plain
```

```
1617546453.705 47 192.168.51.30 TCP_MISS/200 327 GET  
http://detectportal.firefox.com/success.txt? user1 HIER_DIRECT/34.107.221.82 text/plain
```

```
1617546454.105 56 192.168.51.30 TCP_MISS/200 911 POST http://ocsp.digicert.com/ user1  
HIER_DIRECT/93.184.220.29 application/ocsp-response
```

На клиенте Client1 откройте браузер (настроенный на прокси)

