

Решение модуля 1

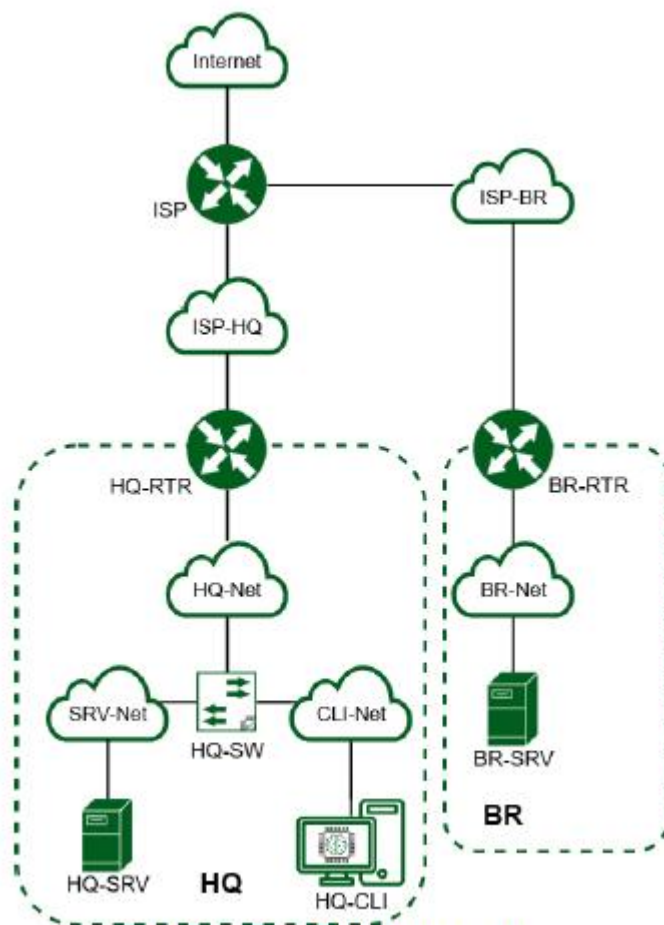


Рисунок 1. Топология сети

ЗАДАНИЕ 1. Произведите базовую настройку устройств:

- Настройте имена устройств согласно топологии. Используйте полное доменное имя
- На всех устройствах необходимо сконфигурировать IPv4
- IP-адрес должен быть из приватного диапазона, в случае, если сеть локальная, согласно RFC1918
- Локальная сеть в сторону HQ-SRV(VLAN100) должна вмещать не более 64 адресов
- Локальная сеть в сторону HQ-CLI(VLAN200) должна вмещать не более 16 адресов
- Локальная сеть в сторону BR-SRV должна вмещать не более 32 адресов
- Локальная сеть для управления(VLAN999) должна вмещать не более 8 адресов
- Сведения об адресах занесите в отчёт, в качестве примера используйте Таблицу 3

Таблица 3. Пример

Имя устройства	IP-адрес	Шлюз по умолчанию
BR-SRV	192.168.0.2/24	192.168.0.1

Решение

Таблица адресов

Имя устройства	IP-адрес	Шлюз по умолчанию
ISP	192.168.44.211/24 (получен по DHCP, сеть колледжа) 172.16.4.1/28 172.16.5.1/28	192.168.44.1
HQ-RTR	172.16.4.10/28 192.168.99.1/29 192.168.100.1/28 192.168.200.1/28 10.5.5.1/30	172.16.4.1
HQ-SRV	192.168.100.10/28	192.168.100.1
HQ-CLI	192.168.200.10/28	192.168.200.1
BR-RTR	172.16.5.10/28 192.168.0.1/28 10.5.5.2/30	172.16.5.1
BR-SRV	192.168.0.10/28	

ЗАДАНИЕ 2. Настройка ISP

- Настройте адресацию на интерфейсах:

Интерфейс, подключенный к магистральному провайдеру, получает адрес по DHCP

Настройте маршруты по умолчанию там, где это необходимо

Интерфейс, к которому подключен HQ-RTR, подключен к сети 172.16.4.0/28

Интерфейс, к которому подключен BR-RTR, подключен к сети 172.16.5.0/28

На ISP настройте динамическую сетевую трансляцию в сторону HQ-RTR и BR-RTR для доступа к сети Интернет

Решение

Настройка адресов на BM:

ISP:

apt-get update -y

apt-get install -y NetworkManager-tui

ip -br a

```
root@ISP ~l# ip -br a
lo                UNKNOWN
ens18             UP
ens19             UP
ens20             UP
```

vim /etc/net/ifaces/ens18/options (NM_CONTRILLED=yes)

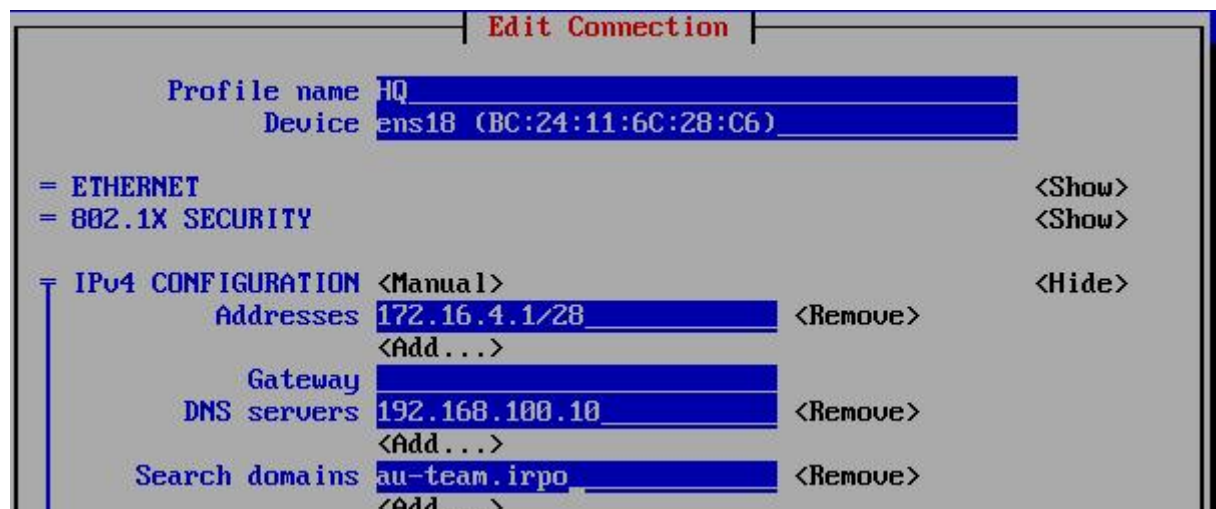
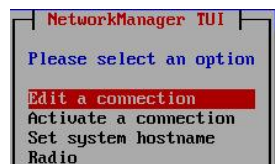
vim /etc/net/ifaces/ens19/options (NM_CONTRILLED=yes)

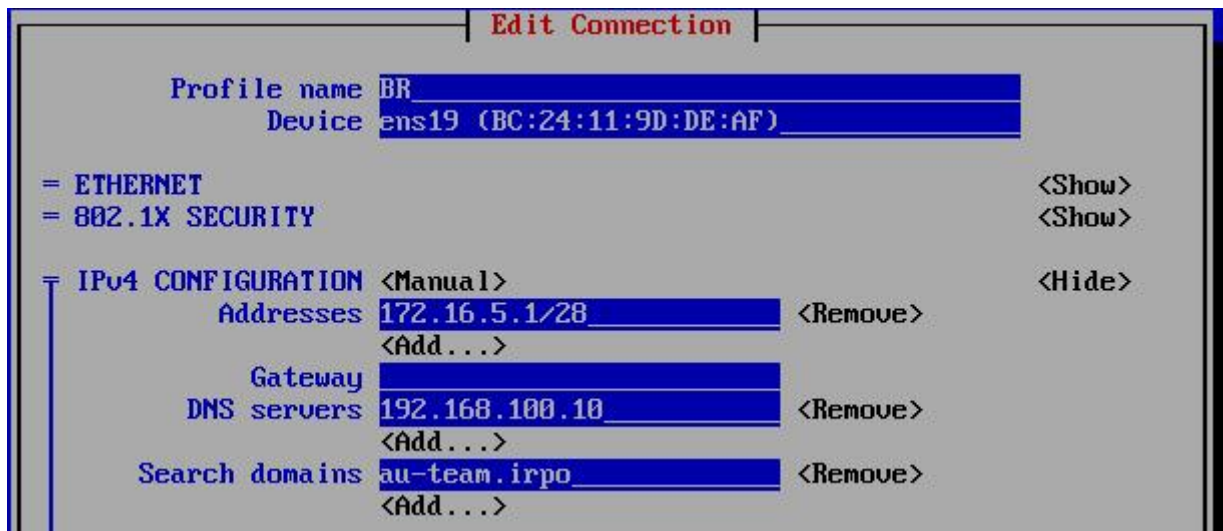
systemctl enable --now NetworkManager

systemctl restart network

#теперь работает команда nmtui

nmtui





Перезагрузить интерфейсы, проверить, что адреса появились

```
[root@ISP ~]# ip -br a
lo                UNKNOWN      127.0.0.1/8 ::1/128
ens18             UP          172.16.4.1/28 fe80::c9fd:772a:fc7:1d02/64
ens19             UP          172.16.5.1/28 fe80::8b5:35b4:2317:baba/64
ens20             UP          192.168.44.211/24 fe80::be24:11ff:fe18:3434/64
```

Включаем пересылку пакетов:

`vim /etc/net/sysctl.conf` (`net.ipv4.ip_forward = 1`)

- включаем NAT (чтобы в дальнейшем все ВМ выходили в интернет):

`iptables -t nat -A POSTROUTING -j MASQUERADE -o ens20` (интерфейс в сторону колледжа)

`iptables-save >> /etc/sysconfig/iptables`

`systemctl enable --now iptables`

BR-RTR:

Включим интернет, чтобы скачать network-manager:

```
[root@BR-RTR ~]# ip -br a
lo                UNKNOWN
ens18             UP
ens19             UP
```

ens18 – подключен к ISP

ens19 – подключен к BR-SRV

`[root@BR-RTR ~]# vim /etc/net/ifaces/ens18/ipv4address` (установим ip-адрес)

```
172.16.5.10/28
```

```
[root@BR-RTR ~]# vim /etc/net/ifaces/ens18/ipv4route
```

 (установим шлюз)

```
default via 172.16.5.1
```

 - адрес ISP

```
systemctl restart network
```

Включаем пересылку пакетов:

```
vim /etc/net/sysctl.conf (net.ipv4.ip_forward = 1)
```

проверить, что есть интернет:

```
ping 8.8.8.8
```

```
apt-get update -y
```

```
apt-get install -y NetworkManager-tui
```

```
vim /etc/net/ifaces/ens18/options (NM_CONTRILLED=yes)
```

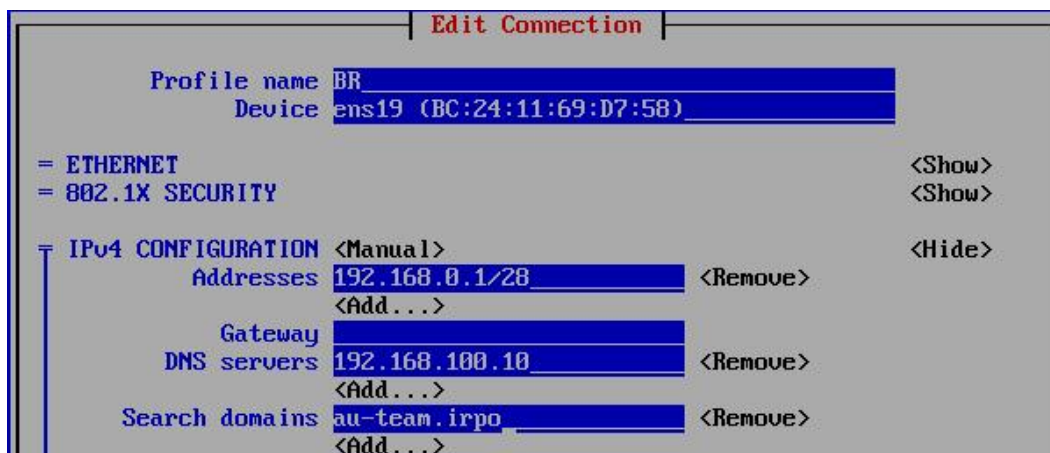
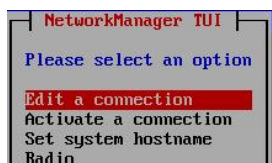
```
vim /etc/net/ifaces/ens19/options (NM_CONTRILLED=yes)
```

```
systemctl restar network
```

```
systemctl restar Network-Manager
```

#теперь работает команда nmtui

```
nmtui
```



Перезагрузить интерфейсы, проверить, что адреса появились

```
[root@BR-RTR ~]# ip -br a
lo                UNKNOWN      127.0.0.1/8 ::1/128
ens18             UP          172.16.5.10/28 fe80::be24:11ff:fee1:b83d/64
ens19             UP          192.168.0.1/28  fe80::7b04:52e2:7301:64b/64
```

Не забыть поменять hostname!! (через nmtui во вкладке «set a hostname»)

BR-SRV:

Настроим интерфейс:

```
[root@BR-SRV ~]# ip -br a
lo                UNKNOWN
ens18             UP
```

`vim /etc/net/ifaces/ens18/ipv4address` (установим ip-адрес)

```
192.168.0.10/28
```

`vim /etc/net/ifaces/ens18/ipv4route` (установим шлюз)

`default via 192.168.0.1` - адрес BR-RTR

`vim /etc/net/ifaces/ens18/resolv.conf` - dns server

```
nameserver 192.168.100.10
domain au-team.irpo
```

- при **не скачивании пакетов поставить 8.8.8.8**

`systemctl restart network`

Включаем пересылку пакетов:

`vim /etc/net/sysctl.conf` (`net.ipv4.ip_forward = 1`)

Не забыть поменять hostname!! (через nmtui во вкладке «set a hostname»)

HQ-RTR:

Включим интернет, чтобы скачать network-manager:

```
[root@HQ-RTR ~]# ip -br a
lo                UNKNOWN
ens18             UP
ens19             UP
```

ens18 – подключен к ISP

ens19 – подключен к HQ-SW

`root@HQ-RTR ~]# vim /etc/net/ifaces/ens18/ipv4address` (установим ip-адрес)

```
172.16.4.10/28
```

```
[root@HQ-RTR ~]# vim /etc/net/ifaces/ens18/ipv4route (установим шлюз)
```

```
default via 172.16.4.1 - адрес ISP
```

```
[root@HQ-RTR ~]# vim /etc/net/ifaces/ens18/resolv.conf - dns-server
```

```
nameserver 192.168.100.10  
domain au-team.irpo
```

- при **не скачивании пакетов** поставить **8.8.8.8**

```
systemctl restart network
```

Включаем пересылку пакетов:

```
vim /etc/net/sysctl.conf (net.ipv4.ip_forward = 1)
```

проверить, что есть интернет:

```
ping 8.8.8.8
```

Не забыть поменять hostname!! (через nmtui во вкладке «set a hostname»)

На HQ-SRV, BR-SRV – поставить ip-адреса аналогично через файлы

/etc/net/ifaces/ens18/ipv4address – указать IP

/etc/net/ifaces/ens18/ipv4router – указать шлюз (default via <IP RTR>)

/etc/net/ifaces/ens18/resolv.conf - указать адрес DNS-сервера и домен (nameserver 192.168.100.10, domain au-team.irpo) - при **не скачивании пакетов** поставить **8.8.8.8**

ЗАДАНИЕ 3. Создание локальных учетных записей

- Создайте пользователя sshuser на серверах HQ-SRV и BR-SRV
- о Пароль пользователя sshuser с паролем **P@ssw0rd**
- о Идентификатор пользователя 1010
- о Пользователь sshuser должен иметь возможность запускать sudo без дополнительной аутентификации.

Решение

```
adduser sshuser  
passwd sshuser  
usermod -u 1010 sshuser
```

- вписать пароль **P@ssw0rd**


```
visudo /etc/sudoers
```

Раскомментировать строки (их не надо писать с 0, просто найти в конце и раскомментировать!!):

```
WHEEL_USERS ALL = (ALL:ALL) ALL
WHEEL_USERS ALL = (ALL:ALL) NOPASSWD: ALL
```

```
vim /etc/group
```

Вписать ТОЛЬКО нового пользователя в группу wheel (она уже СУЩЕСТВУЕТ)

```
wheel:x:10:root,zabbix,user,sshuser_
```

- Создайте пользователя net_admin на маршрутизаторах HQ-RTR и BR-RTR
- o Пароль пользователя net_admin с паролем P@\$s\$word
- o При настройке ОС на базе Linux, запускать sudo без дополнительной аутентификации

Решение

```
adduser net_admin
passwd net_admin
```

- вписать пароль P@ssw0rd

```
visudo /etc/sudoers
```

Раскомментировать строки (их не надо писать с 0, просто найти в конце и раскомментировать!!):

```
WHEEL_USERS ALL = (ALL:ALL) ALL
WHEEL_USERS ALL = (ALL:ALL) NOPASSWD: ALL
```

```
vim /etc/group
```

Вписать ТОЛЬКО нового пользователя в группу wheel (она уже СУЩЕСТВУЕТ)

ЗАДАНИЕ 4. Настройте на интерфейсе HQ-RTR в сторону офиса HQ виртуальный коммутатор:

- Сервер HQ-SRV должен находиться в ID VLAN 100
- Клиент HQ-CLI в ID VLAN 200

- Создайте подсеть управления с ID VLAN 999
- Основные сведения о настройке коммутатора и выбора реализации разделения на VLAN занесите в отчёт

Решение

Установить и включить службу

```
apt-get install -y openvswitch
systemctl enable --now openvswitch
```

Создать бридж и закрепить его на интерфейс в сторону CLI и SRV

```
ovs-vsctl add-br HQ-SW
ovs-vsctl add-port HQ-SW ens19
```

Создать вланы и привязать их к бриджу

```
ovs-vsctl add-port HQ-SW vlan100 tag=100 -- set interface vlan100 type=internal
ovs-vsctl add-port HQ-SW vlan200 tag=200 -- set interface vlan200 type=internal
ovs-vsctl add-port HQ-SW vlan999 tag=999 -- set interface vlan999 type=internal
```

Создать папки вланов для хранения сетевых параметров

```
mkdir /etc/net/ifaces/van100
mkdir /etc/net/ifaces/van200
mkdir /etc/net/ifaces/van999
```

Скопировать шаблон options в папки вланов

```
cp /etc/net/ifaces/ens19/options /etc/net/ifaces/vlan100/options
cp /etc/net/ifaces/ens19/options /etc/net/ifaces/vlan200/options
cp /etc/net/ifaces/ens19/options /etc/net/ifaces/vlan999/options
```

Задать адреса вланам

```
echo '192.168.100.1/24' >> /etc/net/ifaces/vlan100/ipv4address
echo '192.168.200.1/24' >> /etc/net/ifaces/vlan200/ipv4address
echo '192.168.99.1/24' >> /etc/net/ifaces/vlan999/ipv4address
```

Перезагрузить службу и проверить, что вланы появились

```
systemctl restart network
```

```
[root@HQ-RTR ~]# ip -br a
lo                UNKNOWN      127.0.0.1/8  ::1/128
ens18             UP          172.16.4.10/28 fe80::be24:11ff:fe5b:44d0/64
ens19            UP          fe80::9ad:da1c:4e6c:fd5e/64
ovs-system        DOWN
vlan100           UNKNOWN     192.168.100.1/28 fe80::5402:97ff:fef2:5995/64
HQ-SW             DOWN
vlan200           UNKNOWN     192.168.200.1/28 fe80::5c0e:eeff:fed0:dcdd/64
vlan999           UNKNOWN     192.168.99.1/29 fe80::7882:7eff:fe01:6255/64
```

ЗАДАНИЕ 5. Настройка безопасного удаленного доступа на серверах HQ-SRV и BR-SRV:

- Для подключения используйте порт 2024
- Разрешите подключения только пользователю sshuser
- Ограничьте количество попыток входа до двух
- Настройте баннер «Authorized access only»

Решение

Добавить параметры SSH по заданию в файл конфигурации

```
vim /etc/openssh/sshd_config  
Port 2024  
AllowUsers sshuser  
MaxAuthTries 2  
Banner /etc/ban
```

Настроить файл баннера

```
vim /etc/ban
```

```
Authorized access only!
```

ЗАДАНИЕ 6. Между офисами HQ и BR необходимо сконфигурировать ip туннель

- Сведения о туннеле занесите в отчёт
- На выбор технологии GRE или IP in IP

Решение

HQ-RTR:

Создать интерфейс для туннеля GRE (IP-tunnel), ens18 - тот, что в ISP

Edit Connection	
Profile name	gre1
Device	gre1
IP tunnel <Hide>	
Mode	<GRE>
Parent	ens18
Local IP	172.16.4.10
Remote IP	172.16.5.10
Input key	
Output key	
MTU	(default)
IPv4 CONFIGURATION <Manual> <Hide>	
Addresses	10.5.5.1/30 <Remove>
	<Add...>
Gateway	
DNS servers	<Add...>
Search domains	<Add...>

Настроить TTL

```
nmcli connection edit gre1
```

```
nmcli> set ip-tunnel.ttl 64
```

```
save
```

```
quit
```

BR-RTR:

Создать интерфейс для туннеля GRE (IP-tunnel), ens18 - тот, что в ISP

Edit Connection	
Profile name	gre1
Device	gre1
IP tunnel <Hide>	
Mode	<GRE>
Parent	ens18
Local IP	172.16.5.10
Remote IP	172.16.4.10
Input key	
Output key	
MTU	(default)
IPv4 CONFIGURATION <Manual> <Hide>	
Addresses	10.5.5.2/30 <Remove>
	<Add...>
Gateway	

Настроить TTL

```
nmcli connection edit gre1
```

```
nmcli> set ip-tunnel.ttl 64
```

```
save
```

```
quit
```

ЗАДАНИЕ 7. Обеспечьте динамическую маршрутизацию: ресурсы одного офиса должны быть доступны из другого офиса. Для обеспечения динамической маршрутизации используйте link state протокол на ваше усмотрение.

- Разрешите выбранный протокол только на интерфейсах в ip туннеле
- Маршрутизаторы должны делиться маршрутами только друг с другом
- **Обеспечьте защиту выбранного протокола посредством парольной защиты**
- Сведения о настройке и защите протокола занесите в отчёт

Решение

BR-RTR:

Установить пакет frr

```
apt-get install -y frr
```

Отредактировать файл

```
[root@BR-RTR ~]# vim /etc/frr/daemons
```

```
ospfd=yes
```

Перезапустить службу и ввести ее в автозагрузку

```
[root@BR-RTR ~]# systemctl enable --now frr
```

```
[root@BR-RTR ~]# vtysh
```

Указать маршруты, пассивные интерфейсы

Сохранить изменения (wr mem)

Перезагрузить службу

```

R-RTR# con ft
Unknown command: con ft
R-RTR# conf t
R-RTR(config)# ip forwarding
R-RTR(config)# router ospf
R-RTR(config-router)# network 10.5.5.0/30 area 0
R-RTR(config-router)# network 192.168.0.0/28 area 0
R-RTR(config-router)# passive-interface default
R-RTR(config-router)# ex
R-RTR(config)# interface gre1
R-RTR(config-if)# no ip ospf passive
R-RTR(config-if)# ex
R-RTR(config)# ex
R-RTR# wr
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Integrated configuration saved to /etc/frr/frr.conf
[OK]
R-RTR# ex
root@BR-RTR ~]# systemctl restart frr.service

```

HQ-RTR:

Установить пакет frr

```
apt-get install -y frr
```

Отредактировать файл /etc/frr/daemons (ospfd = yes)

Перезапустить службу и ввести ее в автозагрузку (systemctl enable --now frr)

```
[root@HQ-RTR ~]# vtysh
```

```

HQ-RTR# conf t
HQ-RTR(config)# ip forwarding
HQ-RTR(config)# router ospf
HQ-RTR(config-router)# network 10.5.5.0/30 area 0

```

```

HQ-RTR(config-router)# network 192.168.100.0/28 area 0
HQ-RTR(config-router)# network 192.168.200.0/28 area 0
HQ-RTR(config-router)# network 192.168.99.0/29 area 0
HQ-RTR(config-router)# ex

```

```

HQ-RTR(config)# int gre1
HQ-RTR(config-if)# no ip ospf passive
HQ-RTR(config-if)# ex

```

```
HQ-RTR(config)# ex
```

```

HQ-RTR# wr
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Integrated configuration saved to /etc/frr/frr.conf
[OK]
HQ-RTR# ex
[root@HQ-RTR ~]# systemctl restart frr

```

ЗАДАНИЕ 8. Настройка динамической трансляции адресов.

- Настройте динамическую трансляцию адресов для обоих офисов.
- Все устройства в офисах должны иметь доступ к сети Интернет

Решение

HQ-RTR:

```
iptables -t nat -A POSTROUTING -j MASQUERADE -o ens18  
iptables-save >> /etc/sysconfig/iptables  
systemctl enable --now iptables
```

BR-RTR:

```
iptables -t nat -A POSTROUTING -j MASQUERADE -o ens18  
iptables-save >> /etc/sysconfig/iptables  
systemctl enable --now iptables
```

ЗАДАНИЕ 9. Настройка протокола динамической конфигурации хостов.

- Настройте нужную подсеть
- Для офиса HQ в качестве сервера DHCP выступает маршрутизатор HQ-RTR.
- Клиентом является машина HQ-CLI.
- Исключите из выдачи адрес маршрутизатора
- Адрес шлюза по умолчанию – адрес маршрутизатора HQ-RTR.
- Адрес DNS-сервера для машины HQ-CLI – адрес сервера HQ-SRV.
- DNS-суффикс для офисов HQ – au-team.irpo
- Сведения о настройке протокола занесите в отчёт

Решение

```
[root@HQ-RTR ~]# apt-get install -y dhcp-server
```

```
[root@HQ-RTR ~]# vim /etc/sysconfig/dhcpd
```

```
DHCPDARGS=vlan200 - сеть CLI
```

Копируем шаблон конфигурации dhcp


```
[root@HQ-RTR ~]# cp /etc/dhcp/dhcpd.conf.example /etc/dhcp/dhcpd.conf
```

vim /etc/dhcp/dhcpd.conf

```
# option definitions common to all supported networks...
option domain-name "au-team.irpo";
option domain-name-servers 192.168.100.10, 192.168.0.10;
```

```
ddns-update-style interim;
update-static-leases on;

zone au-team.irpo {
    primary 192.168.100.10;
}

zone 100.168.192.in-addr.arpa {
    primary 192.168.100.10;
}

zone 200.168.192.in-addr.arpa {
    primary 192.168.100.10;
}
```

```
subnet 192.168.200.0 netmask 255.255.255.240 {
    range 192.168.200.2 192.168.200.5;
    option routers 192.168.200.1;
}
```

```
host hq-cli {
    hardware ethernet bc:24:11:58:f7:ab;
    fixed-address 192.168.200.10;
}
```

- mac-адрес HQ-CLI

Перезагрузить службу

```
[root@HQ-RTR ~]# systemctl restart dhcpd
[root@HQ-RTR ~]# systemctl enable dhcpd
Synchronizing state of dhcpd.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable dhcpd
Created symlink /etc/systemd/system/multi-user.target.wants/dhcpd.service → /lib/systemd/system/dhcpd.service.
[root@HQ-RTR ~]#
```

ЗАДАНИЕ 10. Настройка DNS для офисов HQ и BR.

- Основной DNS-сервер реализован на HQ-SRV.
- Сервер должен обеспечивать разрешение имён в сетевые адреса устройств и обратно в соответствии с таблицей 2

Таблица 2. DNS-зоны

Устройство	Запись	Тип
HQ-RTR	hq-rtr.au-team.irpo	A PTR
BR-RTR	br-rtr.au-team.irpo	A
HQ-SRV	hq-srv.au-team.irpo	A PTR
HQ-CLI	hq-cli.au-team.irpo	A PTR
BR-SRV	br-srv.au-team.irpo	A
HQ-RTR	moodle.au-team.irpo	CNAME
HQ-RTR	wiki.au-team.irpo	CNAME

- В качестве DNS сервера пересылки используйте любой общедоступный DNS сервер

Решение

Установить bind

```
[root@HQ-SRV ~]# apt-get install -y bind
```

```
[root@HQ-SRV etc]# vim /var/lib/bind/etc/options.conf
```

```
listen-on { any; };
listen-on-v6 { none; };

/*
 * If the forward directive is set to "only", the server will only
 * query the forwarders.
 */
//forward only;
forwarders { 8.8.8.8; };

/*
 * Specifies which hosts are allowed to ask ordinary questions.
 */
allow-query { any; };

/*
 * This lets "allow-query" be used to specify the default zone access
 * level rather than having to have every zone override the global
 * value. "allow-query-cache" can be set at both the options and view
 * levels. If "allow-query-cache" is not set then "allow-recursion" is
 * used if set, otherwise "allow-query" is used if set unless
 * "recursion no;" is set in which case "none;" is used, otherwise the
 * default (localhost; localnets;) is used.
 */
//allow-query-cache { localnets; };

/*
 * Specifies which hosts are allowed to make recursive queries
 * through this server. If not specified, the default is to allow
 * recursive queries from all hosts. Note that disallowing recursive
 * queries for a host does not prevent the host from retrieving data
 * that is already in the server's cache.
 */
allow-recursion { any; };
```

```
[root@HQ-SRV etc]# vim /var/lib/bind/etc/rfc1912.conf
```

```
zone "au-team.irpo" {
    type master;
    file "au-team";
};

zone "100.168.192.in-addr.arpa" {
    type master;
    file "100.168.192.in-addr.arpa";
};

zone "200.168.192.in-addr.arpa" {
    type master;
    file "200.168.192.in-addr.arpa";
};
```

Копируем файл empty (шаблон) в au-team, 100.168.192.in-addr.arpa, 200.168.192.in-addr.arpa:

```
[root@HQ-SRV ~]# cd /var/lib/bind/etc/zone
[root@HQ-SRV zone]# cp empty au-team
[root@HQ-SRV zone]# cp empty 100.168.192.in-addr.arpa
[root@HQ-SRV zone]# cp empty 200.168.192.in-addr.arpa
```

сконфигурируем зоны

```
[root@HQ-SRV zone]# vim au-team
```

```
; BIND reverse data file for empty rfc1918 zone
;
; DO NOT EDIT THIS FILE - it is used for multiple zones.
; Instead, copy it and use that copy.
;
$TTL      1D
@         IN      SOA      hq-srv.au-team.irpo. root.au-team.irpo. (
                                2024092400    ; serial
                                12H             ; refresh
                                1H             ; retry
                                1W             ; expire
                                1H             ; ncache
        )
        IN      NS       hq-srv.au-team.irpo.
        IN      A        192.168.100.10
hq-rtr    IN      A       192.168.100.1
br-rtr    IN      A       192.168.0.1
hq-srv    IN      A       192.168.100.10
hq-cli    IN      A       192.168.200.10
br-srv    IN      A       192.168.0.10
noodle    IN      CNAME   hq-rtr.
wiki      IN      CNAME   hq-rtr.
```

```
[root@HQ-SRV zone]# vim 100.168.192.in-addr.arpa
```

```

; BIND reverse data file for empty rfc1918 zone
;
; DO NOT EDIT THIS FILE - it is used for multiple zones.
; Instead, copy it and use that copy.
;
$TTL      1D
@         IN      SOA      100.168.192.in-addr.arpa. root.100.168.192.in-addr.arpa. (
                                2024092400      ; serial
                                12H               ; refresh
                                1H               ; retry
                                1W               ; expire
                                1H               ; ncache
        )
        IN      NS       100.168.192.in-addr.arpa.
        IN      A        192.168.100.10
10        IN      PTR     hq-srv.au-team.irpo.
1         IN      PTR     hq-rtr.au-team.irpo.

```

```
[root@HQ-SRV zone]# vim 200.168.192.in-addr.arpa.
```

```

; BIND reverse data file for empty rfc1918 zone
;
; DO NOT EDIT THIS FILE - it is used for multiple zones.
; Instead, copy it and use that copy.
;
$TTL      1D
@         IN      SOA      200.168.192.in-addr.arpa. root.200.168.192.in-addr.arpa. (
                                2024092400      ; serial
                                12H               ; refresh
                                1H               ; retry
                                1W               ; expire
                                1H               ; ncache
        )
        IN      NS       200.168.192.in-addr.arpa.
        IN      A        192.168.100.10
2         IN      PTR     hq-cli.au-team.irpo.
1         IN      PTR     hq-rtr.au-team.irpo.

```

Настраиваем утилиту rndc (для корректного запуска bind)

```
[root@HQ-SRV zone]# cd /var/lib/bind/etc/
```

```
[root@HQ-SRV etc]# rndc-confgen > /var/lib/bind/etc/rndc.key
```

```
[root@HQ-SRV etc]# sed -i '6,$d' rndc.key
```

```
[root@HQ-SRV etc]# chgrp -R named zone/
```

```
[root@HQ-SRV etc]# named-checkconf
```

```

[root@HQ-SRV etc]# named-checkconf -z
zone au-team.irpo/IN: loaded serial 2024092400
zone 100.168.192.in-addr.arpa/IN: loaded serial 2024092400
zone 200.168.192.in-addr.arpa/IN: loaded serial 2024092400
[root@HQ-SRV etc]#

```

Перезапустить службу

```
[root@HQ-SRV etc]# systemctl enable --now bind
Synchronizing state of bind.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable bind
Created symlink /etc/systemd/system/multi-user.target.wants/bind.service → /lib/systemd/system/bind.service.
```

ЗАДАНИЕ 11. Настройте часовой пояс на всех устройствах, согласно месту проведения экзамена. (+ЗАДАНИЕ 3. Модуля 2)

- ✓ В качестве сервера выступает HQ-RTR
- ✓ На HQ-RTR настройте сервер chrony, выберите стратум 5
- ✓ В качестве клиентов настройте HQ-SRV, HQ-CLI, BR-RTR, BR-SRV.

Настройка сервера времени (**HQ-RTR**)

Установить временную зону - Калининград

```
[root@HQ-RTR ~]# timedatectl set-timezone Europe/Kaliningrad
```

Скачать пакет Chrony

```
[root@HQ-RTR ~]# apt-get install -y chrony
```

Сконфигурировать файл для настройки сервера

```
[root@HQ-RTR ~]# vim /etc/chrony.conf
```

Указать адрес сервера (ip HQ-RTR, stratum 5, разрешенные сети для синхронизации)

```
# Serve time even if not synchronized to a time source.
server 192.168.100.1 iburst
local stratum 5

allow 192.168.100.0/28
allow 192.168.200.0/28
allow 192.168.0.0/28
```

Перезагрузить службу:

```
[root@HQ-RTR ~]# systemctl restart chronyd
```

Настройка клиентов синхронизации времени (**HQ-SRV, HQ-CLI, BR-RTR, BR-SRV.**)

Установить службу синхронизации времени

```
apt-get install -y chrony
```


Отредактировать файл, указать адрес сервера, с кем требуется синхронизация (адрес HQ-RTR)

```
vim /etc/chrony.conf
```

```
server 192.168.100.1 iburst
```

Перезагрузить службу

```
systemctl restart chronyd
```

Проверить, что все синхронизируется:

```
[root@BR-RTR ~]# chronyc tracking
Reference ID      : C0A86401 (192.168.100.1)
Stratum          : 5
Ref time (UTC)   : Sat Nov 09 08:32:22 2024
System time      : 0.000657323 seconds fast of NTP time
Last offset      : +0.001096859 seconds
RMS offset       : 0.001096859 seconds
Frequency        : 48.419 ppm slow
Residual freq    : -15.998 ppm
Skew             : 683.342 ppm
Root delay       : 0.047409151 seconds
Root dispersion  : 0.001137822 seconds
Update interval  : 0.0 seconds
Leap status      : Normal
```

Решение модуля 2

ЗАДАНИЕ 1. Настройте доменный контроллер Samba на машине BR-SRV.

- ✓ Создайте 5 пользователей для офиса HQ: имена пользователей формата user№.hq. создайте группу hq, введите в эту группу созданных пользователей.

- ✓ Введите в домен машину CLI.

- ✓ Пользователи группы hq имеют право аутентифицироваться на клиентском ПК.

- ✓ Пользователи группы hq должны иметь возможность повышать привилегии для выполнения ограниченного набора команд: cat, grep, id. Запускать другие команды с повышенными привилегиями пользователи группы не имеют права.

- ✓ Выполнять импорт пользователей из файла users.csv. файл будет располагаться на виртуальной машине BR-SRV в папке /opt.

Обновить пакеты на **BR-SRV** и установить пакет task-samba-dc для создания домена:

```
[root@BR-SRV ~]# apt-get update -y
```

```
[root@BR-SRV ~]# apt-get install -y task-samba-dc
```

Удалить папки и файлы (там содержатся файлы с настроенным доменом, чтобы не произошло конфликтов избавляемся от них):

```
[root@BR-SRV ~]# rm -f /etc/samba/smb.conf
[root@BR-SRV ~]# rm -rf /var/lib/samba/
[root@BR-SRV ~]# rm -rf /var/cache/samba/
```

Создать новую директорию:

```
[root@BR-SRV ~]# mkdir -p /var/lib/samba
[root@BR-SRV ~]# mkdir -p /var/lib/samba/sysvol
```

(пароль для администратора используем – P@ssw0rd)

```
[root@BR-SRV ~]# samba-tool domain provision
Realm [AU-TEAM.IRPO]:
Domain [AU-TEAM]:
Server Role (dc, member, standalone) [dc]:
DNS backend (SAMBA_INTERNAL, BIND9_FLATFILE, BIND9_DLZ, NONE) [SAMBA_INTERNAL]:
DNS forwarder IP address (write 'none' to disable forwarding) [192.168.100.10]:
Administrator password:
Retype password:
```

Если все правильно, после загрузки видим приблизительно следующее:

```
INFO 2024-11-16 11:34:41,662 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #2432: A Kerberos configuration suitable for Samba AD has been g
enerated at /var/lib/samba/private/krb5.conf
INFO 2024-11-16 11:34:41,672 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #2434: Merge the contents of this file with your system krb5.conf
f or replace it with this one. Do not create a symlink!
INFO 2024-11-16 11:34:41,915 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #493: Once the above files are installed, your Samba AD server w
ill be ready to use
INFO 2024-11-16 11:34:41,922 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #498: Server Role: active directory domain controller
INFO 2024-11-16 11:34:41,925 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #499: Hostname: BR-SRV
INFO 2024-11-16 11:34:41,928 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #500: NetBIOS Domain: AU-TEAM
INFO 2024-11-16 11:34:41,931 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #501: DNS Domain: au-team.irpo
INFO 2024-11-16 11:34:41,934 pid:2905 /usr/lib64/samba-dc/python3.9/samba/provision/__init__.py #502: DOMAIN SID: S-1-5-21-1470779200-2295878990-8060
12692
```

Вносим в автозагрузку службу

```
[root@BR-SRV ~]# systemctl enable --now samba.service
Synchronizing state of samba.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable samba
Created symlink /etc/systemd/system/multi-user.target.wants/samba.service → /lib/systemd/system/samba.service.
[root@BR-SRV ~]#

[root@BR-SRV ~]# cp /var/lib/samba/private/krb5.conf /etc/krb5.conf
cp: overwrite '/etc/krb5.conf'? y
```

Переходим в директорию /opt (cd /opt)

```
[root@BR-SRV opt]# visudo
```

```
%hq ALL=(ALL) NOPASSWD: /bin/cat, /bin/grep, /usr/bin/id
```

```
[root@BR-SRV opt]# vim users.csv
```

```
user1.hq,P@ssw0rd
user2.hq,P@ssw0rd
user3.hq,P@ssw0rd
user4.hq,P@ssw0rd
user5.hq,P@ssw0rd
```

```
[root@BR-SRV opt]# mkdir smbscript
[root@BR-SRV opt]# cd smbscript/
```

```
[root@BR-SRV smbscript]# vim import.sh
```

```
#!/bin/bash
while IFS=',' read -r username password; do
    sudo samba-tool user create "$username" "$password"
    sudo samba-tool group addmembers hq "$username"
done < /opt/users.csv
```

```
[root@BR-SRV smbscript]# chmod +x import.sh
```

```
[root@BR-SRV smbscript]# samba-tool group create hq
Added group hq
```

```
^C Stopped samba-tool group createusers.sh
```

```
[root@BR-SRV smbscript]# ./import.sh
```

```
User 'user1.hq' added successfully
```

```
Added members to group hq
```

```
User 'user2.hq' added successfully
```

```
Added members to group hq
```

```
User 'user3.hq' added successfully
```

```
Added members to group hq
```

```
User 'user4.hq' added successfully
```

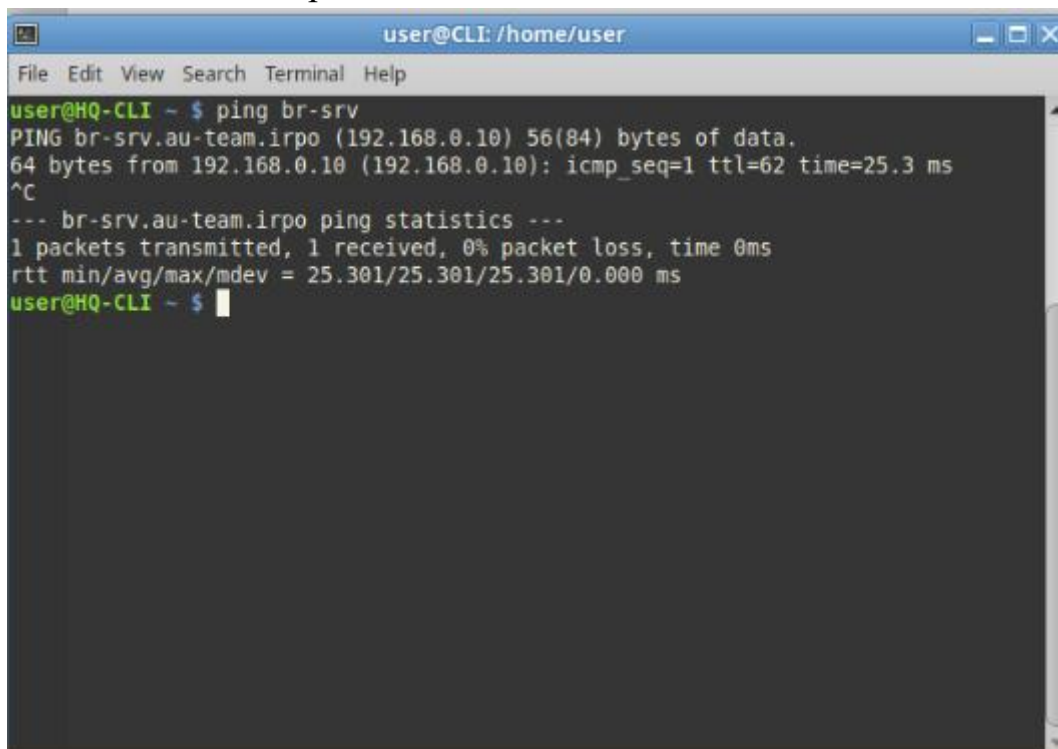
```
Added members to group hq
```

```
User 'user5.hq' added successfully
```

```
Added members to group hq
```

Переходим на VM HQ-CLI

Проверяем, что VM точно имеет возможность общаться как с dns-сервером, так и с VM на которой создан домен:



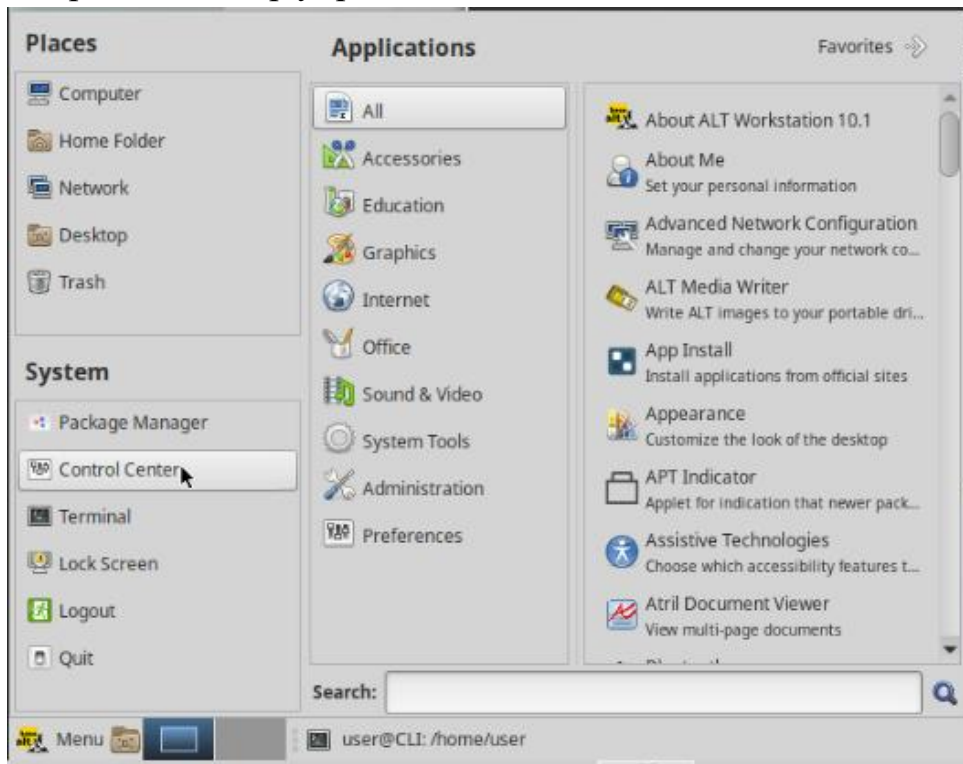
Кроме, того изменяет файл `/etc/resolv.conf`

```
search au-team.irpo
nameserver 192.168.0.10
```

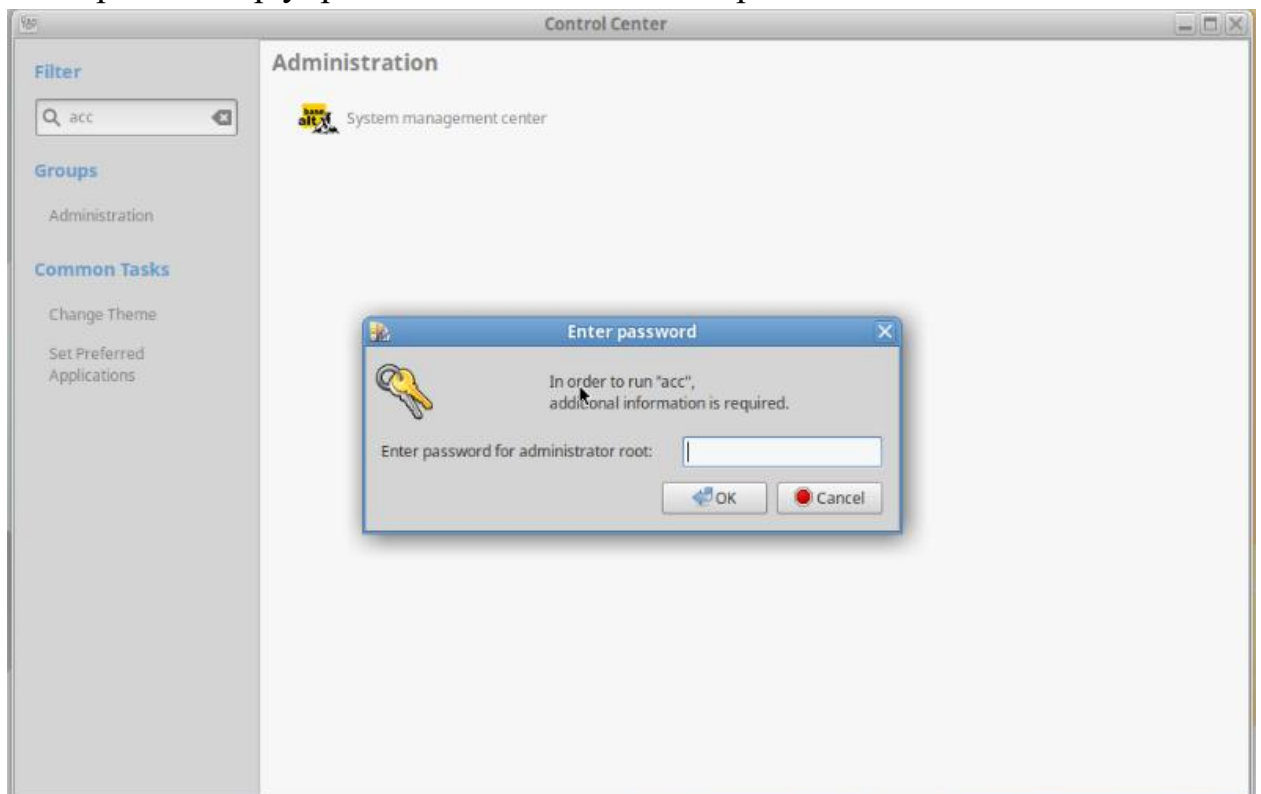
 - адрес BR-SRV

Подключение к домену:

Открываем центр управления

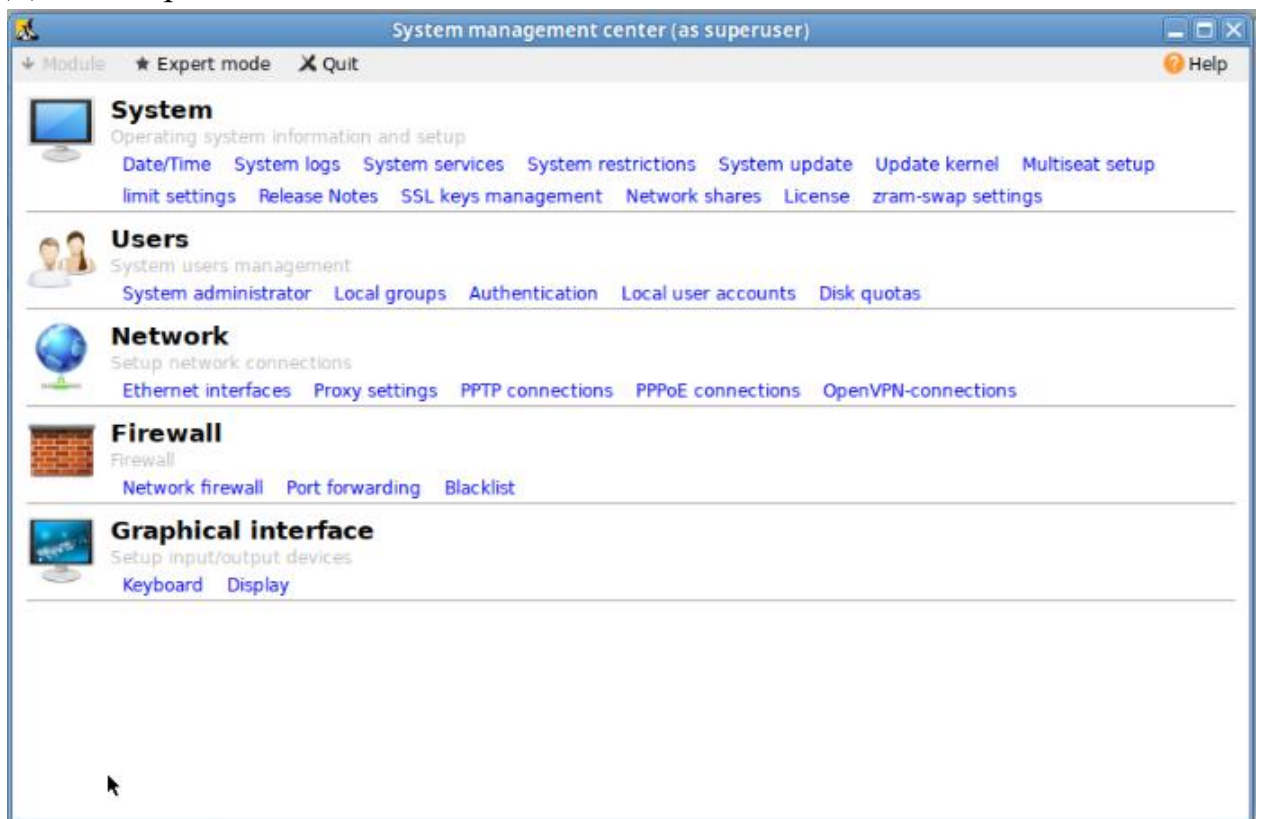


Выбираем центр управления системой или через поиск ищем асс:



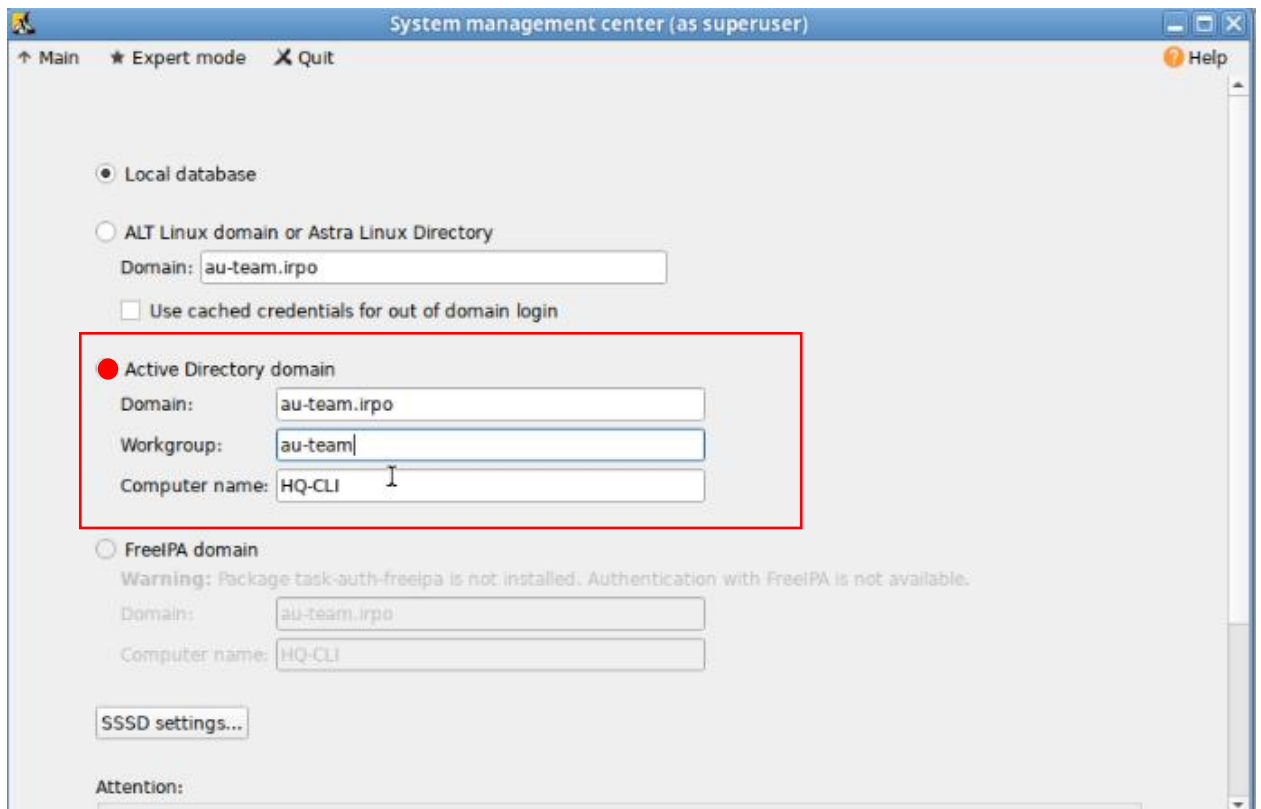
Вводим пароль системного администратора (**toor**)

Далее откроется такое окно:

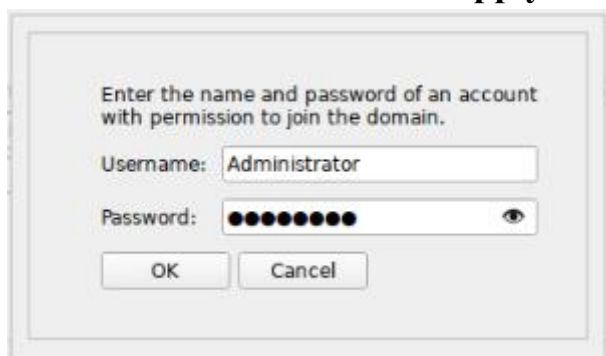


В разделе Users выбираем аутентификацию:



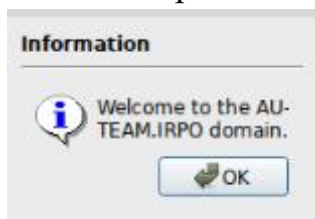


Листаем вниз и нажимает «Apply»



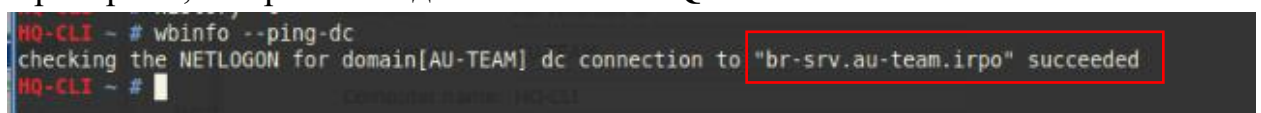
- пароль P@ssw0rd

Если все правильно сработало, то должно появиться приветственное окно:



Делаем **reboot**, после заходим в терминал -

Проверяем, что работает домен на **CLI-HQ**:



Возвращаемся на ВМ **BR-SRV** (пароль P@ssw0rd)

```
[root@BR-SRV smbscript]# samba-tool dns add br-srv.au-team.irpo au-team.irpo hq-rtr A 192.168.100.1 -U Administrator
Password for [AU-TEAM\Administrator]:
Record added successfully
[root@BR-SRV smbscript]# samba-tool dns add br-srv.au-team.irpo au-team.irpo wiki CNAME hq-rtr.au-team.irpo -U Administrator
Password for [AU-TEAM\Administrator]:
Record added successfully
[root@BR-SRV smbscript]# samba-tool dns add br-srv.au-team.irpo au-team.irpo moodle CNAME hq-rtr.au-team.irpo -U Administrator
Password for [AU-TEAM\Administrator]:
Record added successfully
[root@BR-SRV smbscript]#
```

ЗАДАНИЕ 2. Сконфигурируйте файловое хранилище:

- ✓ При помощи трех дополнительных дисков, размером 1Гб каждый, на HQ-SRV сконфигурируйте дисковый массив уровня 5
- ✓ Имя устройства md0, конфигурация массива размещается в файле /etc/mdadm.conf
- ✓ Обеспечьте автоматическое монтирование в папку /raid5
- ✓ Создайте раздел, отформатируйте раздел, в качестве файловой системы используйте ext4
- ✓ Настройте сервер сетевой файловой системы(nfs), в качестве папки общего доступа выберите /raid5/nfs, доступ для чтения и записи для всей сети в сторону HQ-CLI
- ✓ На HQ-CLI настройте автосмонтирование в папке /mnt/nfs
- ✓ Основные параметры сервера отметьте в отчете.

Переходим на HQ-SRV

Проверяем наличие свободных дисков:

```
[root@HQ-SRV ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda 8:0 0 30G 0 disk
└─sda1 8:1 0 30G 0 part /
sdb 8:16 0 1G 0 disk
└─sdb1 8:17 0 494M 0 part [SWAP]
sdc 8:32 0 1G 0 disk
sdd 8:48 0 1G 0 disk
sde 8:64 0 1G 0 disk
[root@HQ-SRV ~]#
```

Создаем raid5 из свободных дисков:

```
[root@HQ-SRV ~]# mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/sdc /dev/sdd /dev/sde
mdadm: layout defaults to left-symmetric
mdadm: layout defaults to left-symmetric
mdadm: chunk size defaults to 512K
mdadm: partition table exists on /dev/sdc
mdadm: partition table exists on /dev/sdc but will be lost or
       meaningless after creating array
mdadm: partition table exists on /dev/sdd
mdadm: partition table exists on /dev/sdd but will be lost or
       meaningless after creating array
mdadm: size set to 1046528K
Continue creating array? yes
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
[root@HQ-SRV ~]#
```

Выводим информацию о созданном raid-массиве, далее в качестве файловой системы выставляем на созданный раздел - ext4:

```
[root@HQ-SRV ~]# mdadm --detail --scan | sudo tee a /etc/mdadm.conf
ARRAY /dev/md0 metadata=1.2 name=HQ-SRV:0 UUID=01aa746c:7da8d144:90c6b684:2883aba5
[root@HQ-SRV ~]# mkfs.ext4 /dev/md0
mke2fs 1.46.2 (28-Feb-2021)
Creating filesystem with 523264 4k blocks and 130816 inodes
Filesystem UUID: 48e3c6a6-b9b7-4ed1-95a9-7ff53b9e25ff
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

[root@HQ-SRV ~]#
```

Создаем папку /raid5 для обеспечения дальнейшего автоматического монтирования:

```
[root@HQ-SRV ~]# mkdir -p /raid5
```

Настраиваем автоматическое монтирование раздела в директорию:

```
[root@HQ-SRV ~]# blkid /dev/md0 >> /etc/fstab
[root@HQ-SRV ~]# vim /etc/fstab
```

Файл ДО: (обращаем внимание на последнюю строку с ошибками, мы ее добавили предыдущей командой)

```
proc                /proc                proc                nosuid,noexec,gid=proc 0 0
devpts              /dev/pts             devpts              nosuid,noexec,gid=tty,mode=620 0 0
tmpfs               /tmp                 tmpfs               nosuid                0 0
UUID=91aea1d4-00d8-43d8-b40f-e6eed7d6cd9d /                    ext4               relative            1 1
UUID=4edd808a-2d5c-428d-8a95-97716d21bca8 swap                 swap                defaults            0 0
/dev/sr0             /media/ALTLinux udf,iso9660        ro,noauto,user=utf8,nofail,comment=x-gvfs-show 0 0
/dev/md0: UUID="48e3c6a6-b9b7-4ed1-95a9-7ff53b9e25ff" BLOCK_SIZE="4096" TYPE="ext4"
```

Изменяем данную строку до следующего вида:

```
proc                /proc                proc                nosuid,noexec,gid=proc 0 0
devpts              /dev/pts             devpts              nosuid,noexec,gid=tty,mode=620 0 0
tmpfs               /tmp                 tmpfs               nosuid                0 0
UUID=91aea1d4-00d8-43d8-b40f-e6eed7d6cd9d /                    ext4               relative            1 1
UUID=4edd808a-2d5c-428d-8a95-97716d21bca8 swap                 swap                defaults            0 0
/dev/sr0             /media/ALTLinux udf,iso9660        ro,noauto,user=utf8,nofail,comment=x-gvfs-show 0 0
UUID="48e3c6a6-b9b7-4ed1-95a9-7ff53b9e25ff" /raid5 ext4 defaults 0 0
```

Проверяем, что пространство монтируется:

```
[root@HQ-SRV ~]# mount -a
[root@HQ-SRV ~]#
```

- не должно быть никаких ошибок!

Настройте сервер сетевой файловой системы(nfs), в качестве папки общего доступа выбираем /raid5/nfs, доступ для чтения и записи для всей сети в сторону HQ-CLI:

Устанавливаем службу сервера

```
[root@HQ-SRV ~]# apt-get install -y nfs-server
```

Создаем папу и настраиваем права:

```
[root@HQ-SRV ~]# mkdir /raid5/nfs
[root@HQ-SRV ~]# vim /etc/exports

/srv/public -ro,insecure,no_subtree_check,fsid=1 *
#/srv/share -ru,insecure,fsid=0,sec=krb5 *
/raid5/nfs 192.168.200.10(rw,sync,no_subtree_check)

[root@HQ-SRV ~]# exportfs -a
[root@HQ-SRV ~]# systemctl restart nfs-server.service
```

- адрес CLI-HQ

Переходим на клиента (**HQ-CLI**)

Создаем папку куда будет монтироваться директория с сервера:

```
HQ-CLI ~ # mkdir -p /mnt/nfs
HQ-CLI ~ # vim /etc/fstab
```

Настраиваем автоматическое монтирование:

```
proc                /proc              proc              nosuid,noexec,gid=proc      0 0
devpts              /dev/pts           devpts            nosuid,noexec,gid=tty,mode=620 0 0
tmpfs               /tmp               tmpfs             nosuid                  0 0
UUID=2696927f-13d5-47af-a4c7-7dba062bead7 /                  ext4              relatime                1 1
UUID=1d772a0d-c33c-4124-b7ef-017fb9a9f4bd swap               swap              defaults                 0 0
/dev/sr0             /media/ALTLinux    udf,iso9660       ro,noauto,user,utf8,nofail,comment=x-gvfs-show 0 0
192.168.100.10:/raid5/nfs /mnt/nfs nfs defaults 0 0
```

- адрес **HQ-SRV**

Проверяем монтирование:

```
HQ-CLI ~ # mount -a
HQ-CLI ~ #
```

- если ошибок нет, значит клиент смог подключиться к серверу

ЗАДАНИЕ 3. Выполнено см. выше в Модуле 1. Задание 11.

ЗАДАНИЕ 4. Сконфигурируйте ansible на сервере BR-SRV

- ✓ Сформируйте файл инвентаря, в инвентарь должны входить HQ-SRV, HQ-CLI, HQ-RTR и BR-RTR
- ✓ Рабочий каталог ansible должен располагаться в /etc/ansible
- ✓ Все указанные машины должны без предупреждений и ошибок отвечать pong на команду ping в ansible посланную с BR-SRV

Переходим на ВМ **BR-SRV**:

Устанавливаем службу Ansible

```
[root@BR-SRV ~]# apt-get install -y ansible
```

Создаем инвентаризационный файл hosts:

```
[root@BR-SRV ~]# vim /etc/ansible/hosts
```

Вписываем данные о ВМ, требуемые в задании:

```
[all]

hq-srv ansible_host=192.168.100.10 ansible_connection=local
hq-cli ansible_host=192.168.200.10 ansible_connection=local
hq-rtr ansible_host=192.168.100.1 ansible_connection=local
br-rtr ansible_host=192.168.100.10 ansible_connection=local
```

Проверяем работу:

При правильной работе – не будет ошибок, красного или оранжевого цвета

Команда: `ansible all -m ping`

```

root@BR-SRV ~# ansible all -m ping
[WARNING]: Platform linux on host hq-srv is using the discovered Python interpreter at /usr/bin/python3, but future installation of another Python interpreter
could change this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
hq-srv | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host hq-rtr is using the discovered Python interpreter at /usr/bin/python3, but future installation of another Python interpreter
could change this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
hq-rtr | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host br-rtr is using the discovered Python interpreter at /usr/bin/python3, but future installation of another Python interpreter
could change this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
br-rtr | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host hq-cli is using the discovered Python interpreter at /usr/bin/python3, but future installation of another Python interpreter
could change this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
hq-cli | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
root@BR-SRV ~#

```

ЗАДАНИЕ 5. Развертывание приложений в Docker на сервере BR-SRV.

- ✓ Создайте в домашней директории пользователя файл `wiki.yml` для приложения MediaWiki.
- ✓ Средствами `docker compose` должен создаваться стек контейнеров с приложением MediaWiki и базой данных.
- ✓ Используйте два сервиса.
- ✓ Основной контейнер MediaWiki должен называться `wiki` и использоваться образ `mediawiki`.
- ✓ Файл `LocalSettings.php` с корректными настройками должен находиться в домашней папке пользователя и автоматически монтироваться в образ.
- ✓ Контейнер с базой данных должен называться `mariadb` и использовать образ `mariadb`.
- ✓ Разверните.
- ✓ Он должен создавать базу с названием `mediawiki`, доступного по стандартному порту, пользователя `wiki` с паролем `P@ssw0rd` должен иметь права доступа к этой базе данных.
- ✓ MediaWiki должна быть доступна извне через порт 8080

См. Приложение 1.

ЗАДАНИЕ 6. На маршрутизаторах сконфигурируйте статическую трансляцию портов.

- ✓ Пробросьте порт 80 в порт 8080 на BR-SRV на маршрутизаторе BR-RTR, для обеспечения работы сервиса `wiki`.

✓ Пробросьте порт 2024 в порт 2024 на HQ-SRV на маршрутизаторе HQ-RTR.

✓ Пробросьте порт 2024 в порт 2024 на BR-SRV на маршрутизаторе HQ-RTR.

См. Приложение 2.

ЗАДАНИЕ 7.

См. приложение 3.

ЗАДАНИЕ 8.

См. приложение 4.

ЗАДАНИЕ 9.

✓ Установить браузер yandex