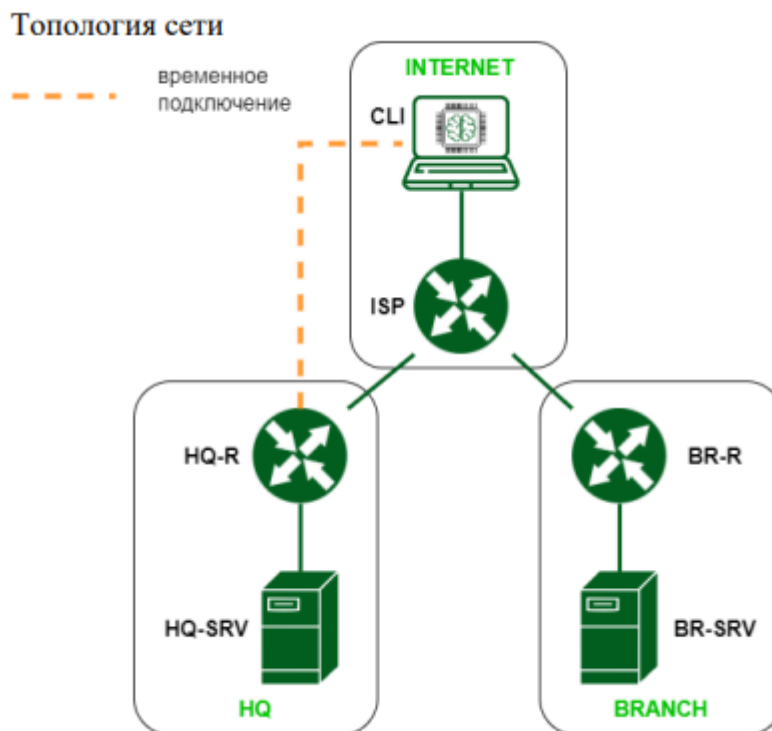


## Обновлено 09.04.2024 V1.3



### Преднастройка

Если в задании не будут использоваться встроенные репозитории, а будет возможность скачивать все пакеты из интернета, необходимо отключить проверку пакетов через cdrom зайдя по пути

**Nano /etc/apt/sources.list**

и закомментировать находящуюся там строку.

### Задание 1 модуля 1

**1. Выполните базовую настройку всех устройств:**

**А. Присвоить имена в соответствии с топологией**

**Примечание:** для выполнения данного задания необходимо постоянное изменение имени каждого устройства, указанного на топологии (временное изменение, действует только до перезагрузки системы и не является верным выполнением задания)

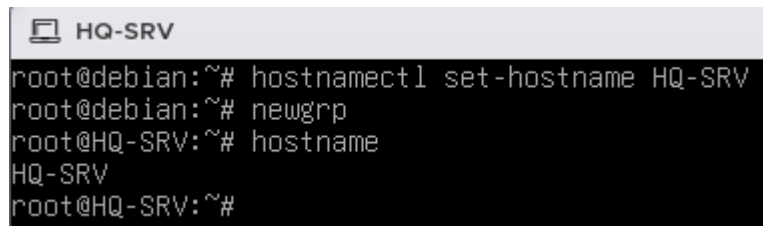
**Решение:**

Для фиксированного изменения имени компьютера, необходимо использовать команду:

## **hostnamectl set-hostname Имя устройства**

Для изменения имени компьютера в текущем сеансе без перезагрузки можно воспользоваться командой:

### **newgrp**



```
HQ-SRV
root@debian:~# hostnamectl set-hostname HQ-SRV
root@debian:~# newgrp
root@HQ-SRV:~# hostname
HQ-SRV
root@HQ-SRV:~#
```

Рисунок 1 — Пример изменения имени устройства

В. Рассчитайте IP-адресацию IPv4 и IPv6. Необходимо заполнить таблицу №1, чтобы эксперты могли проверить ваше рабочее место.

С. Пул адресов для сети офиса BRANCH - не более 16

Д. Пул адресов для сети офиса HQ - не более 64

**Примечание:** Для сетей офисов HQ (входят устройства HQ-R и HQ-SRV) и офисов BRANCH (входят устройства BR-R и BR-SRV), необходимо рассчитать IPv4 и IPv6 адреса согласно пунктам С и Д, для устройств CLI и ISP, можно выбирать адреса из пула серых адресов с стандартной маской /24 255.255.255.0 (при условии, что IP адреса этих устройств не будут заданы заранее или не будут указаны другие условия в задании)

### **Решение:**

**Для расчёта IPv4 адресов можно воспользоваться стандартной таблицей масок от 24 до 32 (при условии, если количество адресов в сети 256 или меньше и изменяется только последний октет в адресе)**

255.255.255.0 /24 маска — 1 сеть в которой 256 адресов от 0 до 255

255.255.255.128 /25 маска — 2 сети в каждой из которых по 128 адресов от 0 до 127 и от 128 до 256

255.255.255.192 /26 маска — 4 сети в каждой из которых по 64 адреса

255.255.255.224 /27 маска — 8 сетей по 32 адреса

255.255.255.240 /28 маска — 16 сетей по 16 адресов

255.255.255.248 /29 маска — 32 сети по 8 адресов

255.255.255.252 /30 маска — 64 сети по 4 адреса

*255.255.255.254 /31 маска — 128 сетей по 2 адреса*

*255.255.255.255 /32 маска — 256 сетей по 1 адресу*

*Исходя из таблицы мы понимаем, что в офисе HQ используется 26 маска, а в офисе Branch используется 28 маска*

*При изменении 3-го октета в адресе используются маски от 16 до 23,*

*при изменении 2-го октета в адресе используются маски от 8 до 15*

*при изменении 1-го октета в адресе используются маски от 1 до 7*

**В сетях IPV6 размер маски составляет 128 бит**

**Однако последние 32 бита маски (от 96 до 128) полностью идентичны маскам в IPv4, однако структура адреса отличается, так как IPv6 работает в шестнадцатеричной системе счисления размер одного октета равен 16 битам, и полный адрес имеет вид xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx , где X значения от 0 до F. Следовательно аналогом диапазона от 24 до 32 в ipv4 , будут маски от 120 до 128 в IPv6.**

**так**

***/120 маска — 1 сеть в которой 256 адресов находящиеся в диапазоне от xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx00 до xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxff***

***/121 маска — 2 сети в каждой из которых по 128 адресов первая сеть в диапазоне***

***от***

***xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx00***

***до***

***xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx7f***

***вторая сеть в диапазоне***

***от***

***xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx80***

***до***

***xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxff***

**Объяснение:**

**Так как размер одного октета в IPv6 равняется 16 битам , в то время как в IPv4 оно равно 8 битам , изменяются лишь два последних числа в**

октете

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****xx** , а размер будет равен 256

битам

, в случае если у нас будут 2 сети по 128 адресов, то есть в каждую сеть вместить по 128 значений (заполнение начинается всегда справа)

при заполнении правого значения на максимум

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****0F** ,

левое значение увеличивается на один

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****10**

и происходит снова заполнение правого значения до

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****1F** и т.д.

пока не будут вмещены все 128 битов (от 0 до 127), 128(число битов) делим на 16 (от 0 до F) и отнимаем 1 (Так как нужно учитывать 0)

=  $128/16-1=80$ (восемь и ноль) -1 = 7F (семь ЭФ) и получаем последний адрес для первой **xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****7F**

следовательно, следующая сеть начинается с

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****80**

и заканчивается на

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****FF**

*/122 маска — 4 сети в каждой из которых по 64 адресов первая сеть в диапазоне*

*от*

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****00**

*до*

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****3f**

*Вторая сеть в диапазоне*

*от*

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****40**

*до*

**xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx****7f**

*и т.д.*

*/123 маска — 8 сетей по 32 адреса*

*/124 маска — 16 сетей по 16 адресов*

*/125 маска — 32 сети по 8 адресов*

*/126 маска — 64 сети по 4 адреса*

*/127 маска — 128 сетей по 2 адреса*

*/128 маска — 256 сетей по 1 адресу*

**Для сетевого взаимодействия можно использовать первый октет 2001**

***Так же необходимо подобрать IP адреса (IPv4 и IPv6) которые будут устанавливаться на интерфейсах между маршрутизаторами (если иного не указано в задании, или если они не выданы заранее)***

Имя устройства	IP
CLI	192.168.0.2 255.255.255.0 — к ISP 2001::3:2/120 — к ISP
ISP	192.168.0.1 255.255.255.0 — к CLI 2001::3:1/120 — к CLI 10.10.10.2 255.255.255.252 — к HQ-R 10.10.10.6 255.255.255.252 — к BR-R 2001::7:2/126 — к HQ-R 2001::7:6/126 — к BR-R
HQ-R	192.168.1.1 255.255.255.192 — к HQ-SRV 2001::1:1/122 — к HQ-SRV 10.10.10.1 255.255.255.252 — к ISP 2001::7:1/126 к — ISP
HQ-SRV	192.168.1.2 255.255.255.192 — к HQ-R 2001::1:2/122 — к HQ-R
BR-R	192.168.2.1 255.255.255.240 — к BR-SRV 2001::2:1/124 — к BR-SRV 10.10.10.5 255.255.255.252 — к ISP 2001::7:5/126 — к ISP
BR-SRV	192.168.2.2 255.255.255.240 — к BR-R 2001::2:2/124 — к BR-R

Следующим шагом необходимо установить выбранные IP адреса на соответствующие машины, для этого существуют 2 способа.

### **Первый способ: через network-manager**

Если network manager не установлен, его можно установить командой

```
root@HQ-R:~# apt install network-manager
```

Рисунок 2 — Установка NMTUI

Для того что бы зайти в Network-manager можно воспользоваться командой:

**nmtui**

В nmtui пройдя по пути **Edit a connection** — имя интерфейса

Необходимо настроить ip адреса в соответствии с таблицей адресации

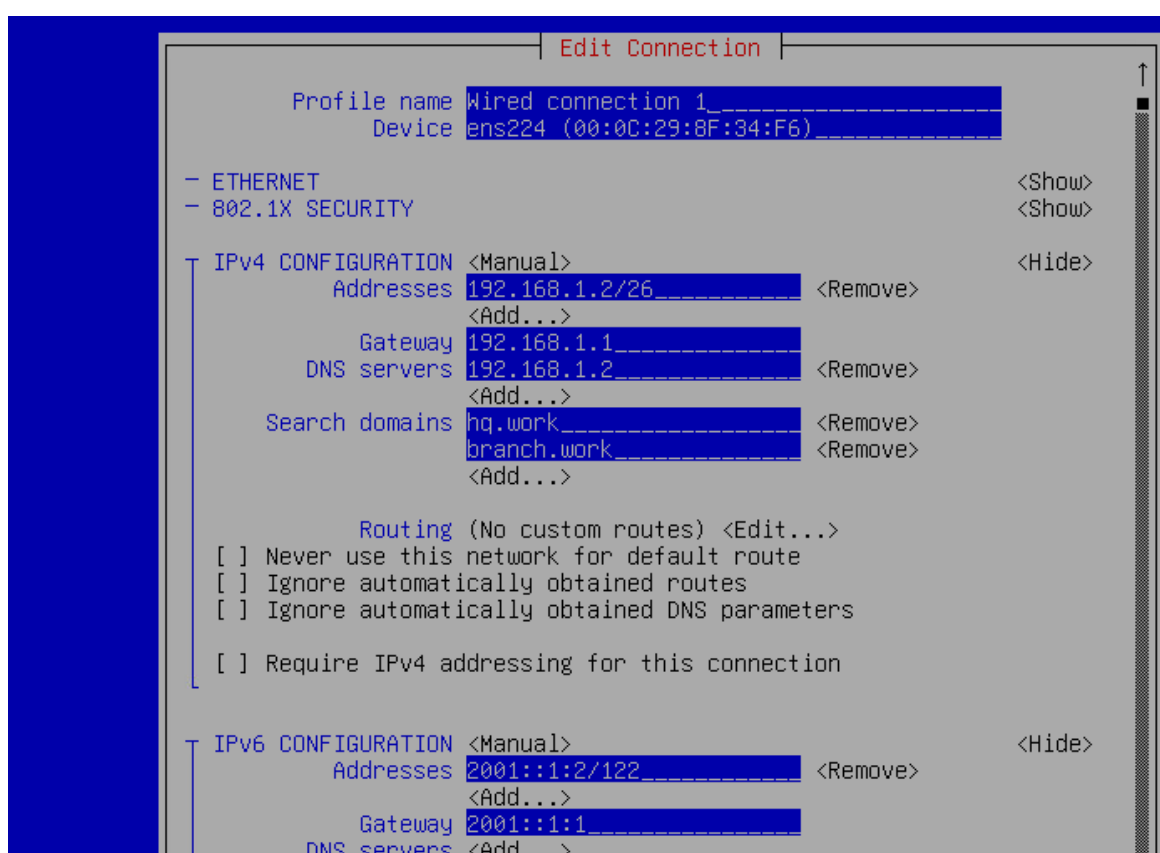


Рисунок 3 — Пример настройки IPv4 и IPv6 на HQ-SRV

После настройки необходимо зайти в activate a connection и перезагрузить все интерфейсы (нажать deactivate и activate на каждом интерфейсе)

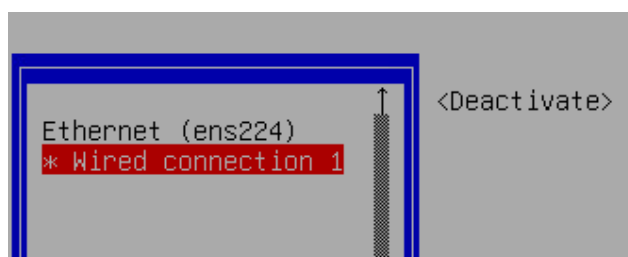


Рисунок 4 — перезагрузка интерфейсов

**Примечание:** на интерфейсах, находящихся между маршрутизаторами, не

нужно указывать dns, достаточно это сделать на внутренних локальных интерфейсах маршрутизаторов.

### Второй способ: через редактирования конфига интерфейсов

Вариант ручной настройки без использования любых программ (в случае если не будет возможности установки nmtui или она будет запрещена). Перед установкой интерфейсов необходимо воспользоваться командой IP A для определения имён 7интерфейсов, находим незаполненный интерфейс, в примере ниже незаполненным интерфейсом является ens256

```
root@HQ-R:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens192: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether 00:0c:29:24:32:0d brd ff:ff:ff:ff:ff:ff
    altname enp11s0
3: ens224: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:24:32:17 brd ff:ff:ff:ff:ff:ff
    altname enp19s0
    inet 192.168.1.1/26 brd 192.168.1.63 scope global noprefixroute ens224
        valid_lft forever preferred_lft forever
    inet6 2001::1:1/122 scope global noprefixroute
        valid_lft forever preferred_lft forever
    inet6 fe80::f275:379c:1db3:ec04/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
4: ens256: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:24:32:21 brd ff:ff:ff:ff:ff:ff
    altname enp27s0
    inet6 fe80::d0fb:69f7:64ae:73b6/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Рисунок 5— Поиск имён интерфейсов для настройки

Определив интерфейс, необходимо воспользоваться командой для просмотра и изменения конфигураций интерфейсов

**nano /etc/network/interfaces**

*или*

**vi /etc/network/interfaces**

И затем сконфигурировать настройки интерфейсов в соответствии с таблицей адресации по примеру, представленному на скриншоте ниже

```
# The primary network interface
allow-hotplug ens192
iface ens192 inet dhcp

auto ens256
iface ens256 inet static
address 10.10.10.1
netmask 255.255.255.252
gateway 10.10.10.2

auto ens256
iface ens256 inet6 static
address 2001::7:1
netmask 126
gateway 2001::7:2
```

Рисунок 6 — Пример настройки интерфейса HQ-R по IPv4 и IPv6 между ISP и HQ-R

Где:

**auto [имя интерфейса]** – команда для подключения к заданной сетевой карте при запуске операционной системы.

**iface [имя интерфейса] inet static** – указание будет ли статичным или динамичным IPv4 адрес адаптера.

**iface [имя интерфейса] inet6 static** – указание будет ли статичным или динамичным IPv6 адрес адаптера.

**address [адрес]** – порт ethernet.

**netmask [адрес]** – маска подсети.

**gateway [адрес]** – шлюз по умолчанию

Так же есть дополнительные настройки:

**dns-nameservers [адрес]** — указание dns адреса

**dns-search [имя]** — указание имени dns (например hq.work)

**Похожие настройки необходимо проделать на всех машинах сети (Если иного не указано в задании)**

**2. Настройте внутреннюю динамическую маршрутизацию по средствам FRR. Выберите и обоснуйте выбор протокола динамической маршрутизации из расчёта, что в дальнейшем сеть будет**



**масштабироваться.**

а. Составьте топологию сети L3.

**Примечание:** Для данного задания необходимо самостоятельно выбрать протокол динамической маршрутизации, исходя из всех поддерживаемых протоколом FRR (OSPF , EIGRP , IS-IS , BGP и т.д.), OSPF подходит для построения средних по размеру сетей, и при этом является открытым стандартом протоколов динамической маршрутизации , в то время как EIGRP проприетарный протокол CISCO IOS, IS-IS и BGP используются для глобальной маршрутизации на уровне провайдеров.

**Решение:** Первым делом необходимо установить пакеты FRR, для этого необходимо воспользоваться командой:

**apt install frr**

Следующим шагом необходимо произвести изменения конфигурационных файлов

**nano /etc/frr/daemons**

и изменить параметры на YES для протокола OSPF

```
GNU nano 7.2 /etc/frr/daemons
# This file tells the frr package which daemons to start.
#
# Sample configurations for these daemons can be found in
# /usr/share/doc/frr/examples/.
#
# ATTENTION:
#
# When activating a daemon for the first time, a config file, even if it is
# empty, has to be present *and* be owned by the user and group "frr", else
# the daemon will not be started by /etc/init.d/frr. The permissions should
# be u=rw,g=r,o=.
# When using "vtysh" such a config file is also needed. It should be owned by
# group "frrvty" and set to ug=rw,o= though. Check /etc/pam.d/frr, too.
#
# The watchfrr, zebra and staticd daemons are always started.
#
bgpd=no
ospfd=yes
ospf6d=yes
ripd=no
ripngd=no
isisd=no
pimd=no
pim6d=no
ldpd=no
nhrpd=no
eigrpd=no
babeld=no
sharpd=no
pbrd=no
bfd=no
fabricd=no
vrpd=no
```

Рисунок 7 — настройка конфигурации FRR

После сохранения конфига, следующим шагом необходимо, перезапустить frr.service командой

**systemctl restart frr**

Далее, после перезагрузки, посредством команды **vtysh** перейти в режим конфигурирования (Настройки идентичны Cisco IOS).

```
Hello, this is FRRouting (version 8.4.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

BR-R# _
```

Рисунок 8 — пример конфигурационного окна

Посредством команд:

**Conf t**

**router ospf**

перейти к конфигурированию протокола ospf

Настройка производится посредством объявления

ospf router-id x.x.x.x

и прилегающих к маршрутизатору сетей

network x.x.x.x/x area x

как показано на рисунке 9

```
router ospf
ospf router-id 3.3.3.3
network 10.10.10.4/30 area 0
network 192.168.2.0/28 area 3
```

Рисунок 9 — пример настройки OSPF на BR-R

Где network 10.10.10.4/30 area 0 — относится к зоне между ISP и BR-R

а network 192.168.2.0/28 area 3 — относится к зоне между BR-R и BR-SRV

**Примечание:** area 0 — является транзитной зоной между маршрутизаторами, а area 1,2,3,4,5 персональными зонами для локальных сетей, для каждой локальной сети отдельная зона

*Похожие настройки, выполняется на всех остальных маршрутизаторах.*

**После завершения конфигурации в frr, необходимо записать конфигурацию в память устройства, командой write, иначе при перезагрузке frr или устройства, все настройки вернутся к дефолтным**

Параллельно на маршрутизаторах участвующих в передаче межсетевого трафика, необходимо настроить маршрутизацию по протоколу IPv6

Для этого необходимо

Для завершения настройки сети необходимо сконфигурировать настройку для передачи пакетов между сетями в файле **nano /etc/sysctl.conf**

переменную **net.ipv4.ip\_forward=1** необходимо раскомментировать и сохранить изменения в файле, и применить изменения командой **sysctl -p**

```
# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
```

Рисунок 10 — настройка пересылки пакетов в режиме маршрутизатора

**Примечание:** при каждой перезагрузке устройства, данная настройка будет изменяться обратно, что связано с загрузкой операционной системы на виртуальной машине для того, чтобы снова включить пересылку пакетов необходимо прописать **sysctl -p**

Так же необходимо настроить похожую конфигурацию, для настройки ospf для протокола IPv6, первым делом настроим пересылку пакетов IPv6

```
# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
net.ipv6.conf.all.forwarding=1
```

Рисунок 11 — настройка пересылки пакетов

в режиме маршрутизатора IPv6

в vtysh посредством команд:

**Conf t**

**router ospf6**

Обозначить роутер ID, и зоны вокруг маршрутизатора

```
router ospf6
ospf6 router-id 0.0.0.1
area 0.0.0.0 range 2001::1:0/122
area 0.0.0.0 range 2001::7:0/126
exit
```

Рисунок 12 — настройка ospf6

Последним шагом в настройке OSPF6 необходимо, привязать зоны к интерфейсам маршрутизатора, т.к. разделение по зонам не обозначено, все

интерфейсы и маршруты можно обозначить в одной зоне.

```
interface ens224
  ipv6 ospf6 area 0.0.0.0
exit
!
interface ens256
  ipv6 ospf6 area 0.0.0.0
exit
```

Рисунок 13 — обозначение зон ospf6 на интерфейсах

**Не стоит забывать о команде write !**

**Последним шагом можно составить топологию L3**

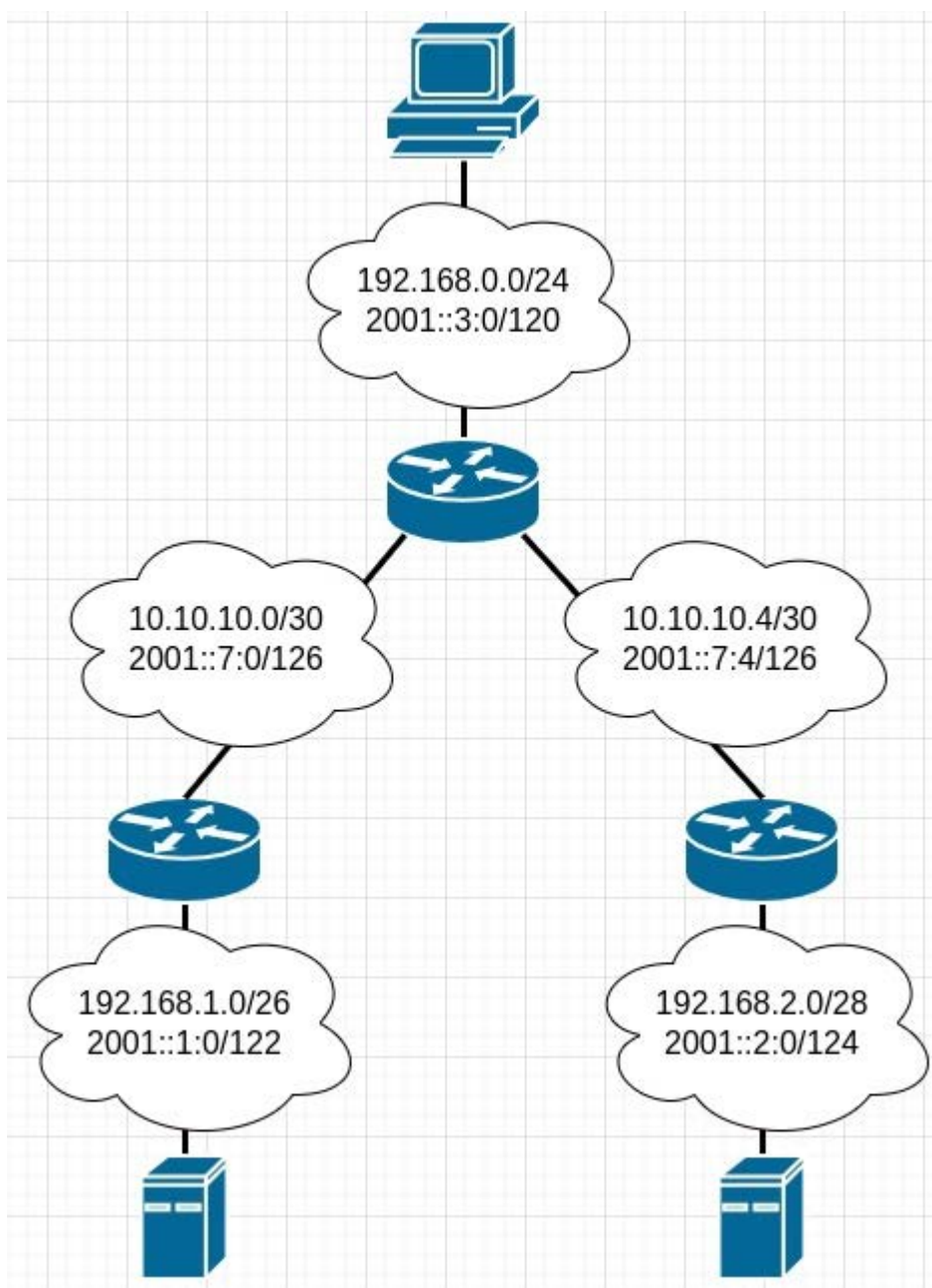


Рисунок 14 — топология L3

### 3. Настройте автоматическое распределение IP-адресов на роутере HQ-R.

#### а. Учтите, что у сервера должен быть зарезервирован адрес.

Первым шагом необходимо на машине HQ-R установить dhcp server командой

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

После установки пакета следующим шагом необходимо сконфигурировать файл для указания интерфейсов прослушивания DHCP сервера зайти можно с помощью команды

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

и настроить интерфейс, направленный в сторону клиента, если в сети подразумевается DHCP-relay, то 2 интерфейса в сторону клиента, и в сторону сети откуда исходит запрос.



```
# Additional options to start dhcpd with.  
# Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead  
#OPTIONS=""  
  
# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?  
# Separate multiple interfaces with spaces, e.g. "eth0 eth1".  
INTERFACESv4="ens224 ens256"  
INTERFACESv6="ens224 ens256"
```

Рисунок 15 — Пример указания интерфейсов прослушивания

Далее необходимо настроить 2 конфигурационных файла для IPv4 для IPv6

Которые можно найти по путям **nano /etc/dhcp/dhcpd.conf** и **nano /etc/dhcp/dhcpd6.conf** соответственно

```

default-lease-time 600;
max-lease-time 7200;
ddns-updates on;
ddns-update-style interim;
authoritative;

subnet 192.168.1.0 netmask 255.255.255.192 {
    range 192.168.1.8 192.168.1.62;
    option routers 192.168.1.1;
    option domain-name "hq.work";
    option domain-name-servers 192.168.1.2;
}

```

Рисунок 16 — Пример настройки DHCP для ipv4 без Relay

**ddns-update-style interim** — способ автообновления базы dns

**authoritative** — делает сервер доверенным

**subnet** — указание сети

**range** — пул адресов

**option routers** — шлюз по умолчанию

**Примечание:** после каждого изменения конфигурации необходимо перезагружать DHCP сервер для применения конфигурации

***systemctl stop isc-dhcp-server***

***systemctl start isc-dhcp-server***

А для того, чтобы после перезагрузки DHCP-сервер автоматически включался можно воспользоваться командой ***systemctl enable isc-dhcp-server***

Настройка DHCP по ipv6 имеет похожие настройки как показано на рисунке 17

```

default-lease-time 2592000;
preferred-lifetime 604800;
option dhcp-renewal-time 3600;
option dhcp-rebinding-time 7200;
allow leasequery;

subnet6 2001::1:0/122 {
    range6 2001::1:3 2001::1:3e;
    option dhcp6.name-servers 2001::1:2;
    option dhcp6.domain-search "hq.work";
}

option dhcp6.info-refresh-time 21600;
authoritative;

```

Рисунок 17 Пример настройки DHCP для IPv6

Однако dhcp6 не способен выдавать шлюз по умолчанию, эту функцию должен выполнять маршрутизатор

Поэтому для настройки маршрутизации для клиентов можно воспользоваться утилитой `radvd`

которую можно установить посредством команды

**`apt install radvd`**

После установки нужно сконфигурировать файл по пути `/etc/radvd.conf` следующего содержания



```

interface ens224
{
MinRtrAdvInterval 3;
MaxRtrAdvInterval 60;
AdvSendAdvert on;
};

```

Рисунок 18 — Пример конфигурации Radvd

где **interface** — это имя интерфейса направленного в локальную сеть

**Min** и **MAX** интервалы — это интервалы рассылки объявлений

**AdvSendAdvert** — это разрешение на выдачу объявлений от маршрутизатор клиентам

После окончания конфигурирования так же необходимо перезагрузить службу Radvd и отправить в Enable

**systemctl stop radvd**

**systemctl start radvd**

**systemctl enable radvd**

**4.Настройте локальные учётные записи на всех устройствах в соответствии с таблицей 2.**

Учётная запись	Пароль	Примечание
Admin	P@ssw0rd	CLI HQ-SRV HQ-R
Branch admin	P@ssw0rd	BR-SRV BR-R
Network admin	P@ssw0rd	HQ-R BR-R BRSRV

Для создания пользователей необходимо ввести команду

**adduser имя\_пользователя**

Затем появится поле ввода пароля как показано на рисунке 19

```

root@HQ-R:~# adduser admin
Adding user `admin' ...
Adding new group `admin' (1001) ...
Adding new user `admin' (1001) with group `admin (1001)'
adduser: The home directory `/home/admin' already exists
New password: _

```

Рисунок 19 — окно ввода пароля при создании пользователя

Из необязательных параметров можно указать имя как показано на

рисунке 20

```
Full Name []: Admin
Room Number []:
Work Phone []:
Home Phone []:
Other []:
Is the information correct? [Y/n] _
```

Рисунок 20 — параметры учётной записи

Так же возможно понадобится выдать Root права для данных клиентов это можно выполнить посредством команды **visudo**

в открывшемся окне необходимо вписать изменения для каждой новой созданной учётной записи как показано на рисунке 21

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
admin   ALL=(ALL:ALL) ALL
```

Рисунок 21 — выдача Root прав пользователям

**5.Измерьте пропускную способность сети между двумя узлами HQ-R-ISP по средствам утилиты iperf 3. Предоставьте описание пропускной способности канала со скриншотами.**

Для начала необходимо установит утилиту iperf3 (не путать с iperf) на машины HQ-R и ISP посредством команды

**apt install iperf3**

при установке будет показано окно автоматического включения демона, нужно выбрать пункт **yes** как показано на рисунке 22

```
| Configuring Iperf3 |
Choose this option if Iperf3 should start automatically as a daemon, now and at boot time.
Start Iperf3 as a daemon automatically?
<Yes>                                     <No>
```

Рисунок 22 — включения демона для iperf3

После установки на обеих машинах, достаточно воспользоваться командой

**iperf3 -c (ip адрес проверяемой машины) -i1 -t20**

```

root@HQ-R:~# iperf3 -c 10.10.10.2 -i1 -t20
Connecting to host 10.10.10.2, port 5201
[ 5] local 10.10.10.1 port 38922 connected to 10.10.10.2 port 5201

```

[ ID]	Interval		Transfer	Bitrate	Retr	Cwnd	
[ 5]	0.00-1.00	sec	1.21 GBytes	10.4 Gbits/sec	400	2.21 MBytes	
[ 5]	1.00-2.00	sec	1.20 GBytes	10.3 Gbits/sec	0	2.42 MBytes	
[ 5]	2.00-3.00	sec	1.16 GBytes	9.99 Gbits/sec	306	1.35 MBytes	
[ 5]	3.00-4.00	sec	1.14 GBytes	9.78 Gbits/sec	81	1.15 MBytes	
[ 5]	4.00-5.00	sec	1.04 GBytes	8.97 Gbits/sec	13	1.15 MBytes	
[ 5]	5.00-6.00	sec	1.05 GBytes	9.05 Gbits/sec	9	1.08 MBytes	
[ 5]	6.00-7.00	sec	1.13 GBytes	9.67 Gbits/sec	2	1.21 MBytes	
[ 5]	7.00-8.00	sec	1.00 GBytes	8.62 Gbits/sec	1	1.19 MBytes	
[ 5]	8.00-9.00	sec	1.03 GBytes	8.83 Gbits/sec	11	1.17 MBytes	
[ 5]	9.00-10.00	sec	1.06 GBytes	9.10 Gbits/sec	64	1.32 MBytes	
[ 5]	10.00-11.00	sec	1.06 GBytes	9.10 Gbits/sec	30	1.23 MBytes	
[ 5]	11.00-12.00	sec	1.26 GBytes	10.8 Gbits/sec	0	1.68 MBytes	
[ 5]	12.00-13.00	sec	1.28 GBytes	11.0 Gbits/sec	392	1.51 MBytes	
[ 5]	13.00-14.00	sec	1.02 GBytes	8.80 Gbits/sec	71	1.37 MBytes	
[ 5]	14.00-15.00	sec	1.56 GBytes	13.4 Gbits/sec	74	1.32 MBytes	
[ 5]	15.00-16.00	sec	1.02 GBytes	8.80 Gbits/sec	2	1.40 MBytes	
[ 5]	16.00-17.00	sec	1.61 GBytes	13.9 Gbits/sec	132	1.32 MBytes	
[ 5]	17.00-18.00	sec	932 MBytes	7.82 Gbits/sec	43	1.09 MBytes	
[ 5]	18.00-19.00	sec	1.14 GBytes	9.80 Gbits/sec	9	1.08 MBytes	
[ 5]	19.00-20.00	sec	1.27 GBytes	10.9 Gbits/sec	106	1.18 MBytes	

---

[ ID]	Interval		Transfer	Bitrate	Retr	
[ 5]	0.00-20.00	sec	23.2 GBytes	9.95 Gbits/sec	1746	sender
[ 5]	0.00-20.00	sec	23.2 GBytes	9.95 Gbits/sec		receiver

Рисунок 23 — скриншот описания пропускной способности

**6.Составьте backup скрипты для сохранения конфигурации сетевых устройств, а именно HQ-R BR-R. Продемонстрируйте их работу.**

Для начала на машинах HQ-R, BR-R создадим каталог, где будет хранится файл созданного скриптом бекапа.

Можно создать его в директории mnt

для этого пропишем **mkdir /mnt/backup**

Далее нам нужно создать сам файл для создания бэкап скрипта, для этого пропишем команду

**touch /etc/backup.sh**

зайдя в файл, необходимо прописать следующие параметры как показано на рисунке 24 или рисунке 25 (по заданию достаточно упрощённого скрипта)

```
#!/bin/bash
backup_files="/home /etc /root /boot /opt"

dest="/mnt/backup"

archive_file="backup.tgz"
echo "Backing up $backup_files to $dest/$archive_file"

tar czf $dest/$archive_file $backup_files

echo "Backup finished"

ls -lh $dest
```

Рисунок 24 — упрощённый backup скрипт

```
#!/bin/bash
backup_files="/home /etc /root /boot /opt"

dest="/mnt/backup"

day=$(date +%A)
hostname=$(hostname -s)
archive_file="$hostname-$day.tgz"
echo "Backing up $backup_files to $dest/$archive_file"
date
echo

tar czf $dest/$archive_file $backup_files

echo
echo "Backup finished"
date

ls -lh $dest
```

Рисунок 25 — расширенный backup скрипт

где **backup\_files** — копируемые директории

**dest** — место куда копируем директории

**day** — параметр который указывает день бэкапа

**hostname** — имя от кого он выполнялся

**archive\_file** — конечное имя файла

**tar czf** — в месте указанное в dest помещает файл с именем указанным в archive\_file с содержимым указанным в backup\_files

**echo** — необязательные строки вывода

Для запуска скрипта достаточно написать **bash (имя\_файла)**

После создания скрипта для того, чтобы распаковать наш backup архив можно воспользоваться командой, указанной на рисунке 26 или 27

```
tar -xvpzf /mnt/backup/backup.tgz -C / --numeric-owner
```

Рисунок 26 – распаковка простого backup архива

```
root@HQ-R:~# tar -xvpzf /mnt/backup/HQ-R-Thursday.tgz -C / --numeric-owner _
```

Рисунок 27 — распаковка сложного backup архива

Для того что бы не писать скрипт дважды, можно с помощью ssh перекинуть его на вторую машину посредством команды scp

для начала подключаемся по ssh командой ssh имя@адрес

**Пример:** ssh network\_admin@192.168.1.1

затем посредством команды

scp /расположение/имя\_файла имя@адрес :/расположение/имя\_файла

**Пример:**

scp /etc/backup.sh network\_admin@192.168.2.1:/home/network\_admin

После успешного копирования возвращаемся в нашу машину и можем перенести скрипт в любое более удобное место

**7. Настройте подключение по SSH для удалённого конфигурирования устройства HQ-SRV по порту 2222. Учтите, что вам необходимо перенаправить трафик на этот порт по средствам контролирования трафика.**

Первым делом необходимо перейти по пути nano /etc/ssh/sshd\_config где в окне конфигурации нам необходимо на HQ-SRV найти строку и изменить значения как указано на рисунке 28

```
Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
```

Рисунок 28 — смена порта доступа по ssh

Для применения конфигурации необходимо перезагрузить службу командой **systemctl restart ssh**

Для перенаправления трафика воспользуемся утилитой `iptables-persistent` которая устанавливается командой **`apt install iptables-persistent`**

После установки создадим правило на подмену порта командой, указанной на рисунке 29

```
root@HQ-SRV:~# iptables -t nat -A PREROUTING -d 192.168.1.0/26 -p tcp -m tcp --dport 22 -j DNAT --to-destination 192.168.1.2:2222
```

Рисунок 29 — правило iptables для подмены порта ssh

Для того что бы не прописывать команду при каждой перезагрузке сохраним нашу текущую конфигурацию командой

**`iptables-save > /etc/iptables/rules.v4`**

Которая будет подгружаться при каждой перезагрузке системы

**8.Настройте контроль доступа до HQ-SRV по SSH со всех устройств, кроме CLI.**

В зависимости от учётной записи, которая должна иметь доступ до сервера возможны следующие развития события, **если нам необходим доступ только от локальных учётных записей, то шаг 1 после всех настроек необходимо вернуть в исходный вид**

### Шаг 1

Заходим в настройки ssh по пути использованному ранее

**`nano /etc/ssh/sshd_config`**

находим и меняем строку как показано на рисунке 30

```
#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

Рисунок 30 — разрешение доступа через root по ssh

***после сохранения изменений перезагружаем службу ssh***

### Шаг 2

Следующим шагом необходимо создать ключ аутентификации ssh с помощью команды `ssh-keygen -C «имя_устройства_с_которого_создан_ключ»` везде необходимо нажать ENTER пока не создастся ключ

Теперь необходимо перенести публичный ключ, на сервер к которому мы будем получать доступ с помощью команды `ssh-copy-id имя@адрес`

### Пример:

ssh-copy-id root@192.168.1.2

ssh-copy-id admin@192.168.1.2

Последним шагом запретим любой доступ клиенту до нашего сервера

На HQ-SRV переходим по пути

**nano /etc/hosts.deny**

и вносим следующую строку в файл

**sshd: 192.168.0.2** (адрес машины CLI)

перезагружаем ssh

В конце не забудьте отключить доступ по root, если иного не указано в задании !

## Задание модуля 2

### 1. Настройте DNS-сервер на сервере HQ-SRV:

а. На DNS сервере необходимо настроить 2 зоны

Зона hq.work, также не забудьте настроить обратную зону.

<b>HQ-R.hq.work</b>	<b>A,PTR</b>	<b>IP - адрес</b>
<b>HQ-SRV.hq.work</b>	<b>A,PTR</b>	<b>IP - адрес</b>

Зона branch.work

<b>BR-R.branch.work</b>	<b>A,PTR</b>	<b>IP - адрес</b>
<b>BR-SRV.branch.work</b>	<b>A</b>	<b>IP - адрес</b>

Вся настройка будет происходить на сервере HQ-SRV

Первым делом необходимо установить пакеты для dns командой

**apt install bind9 dnsutils**

где:

**bind9** — пакеты для создания dns сервера

**dnsutils** — дополнительные пакеты, которые помогут проверить работоспособность (команда host)







Где:

**NS запись** — обозначение сервера ответственного за разрешение запросов к dns

**A запись** — основная запись для зоны прямого просмотра по протоколу ipv4

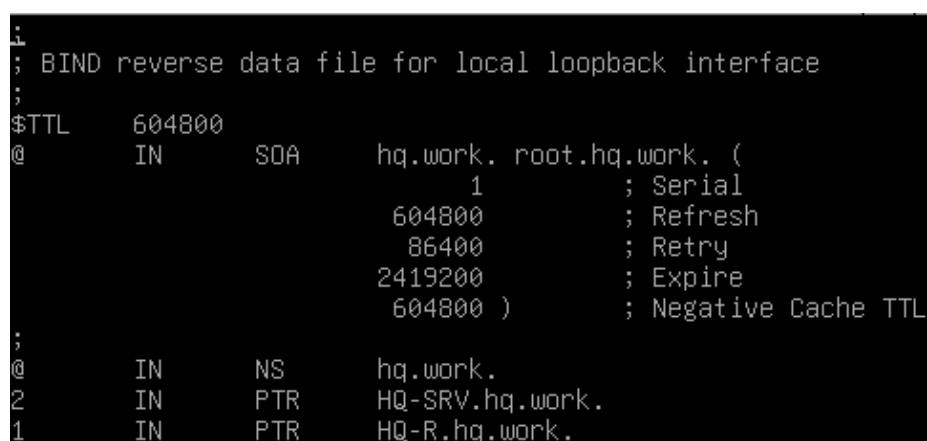
**AAAA запись** - запись для зоны прямого просмотра по протоколу ipv6

**CNAME** — необязательный параметр, для указания альтернативного имени записи

Вторым шагом настроим зону обратного просмотра как указано на скриншоте ниже

Зона находится по пути

**nano /etc/bind/hq\_arpa**



```
1;
2; BIND reverse data file for local loopback interface
3;
4$TTL      604800
5@         IN      SOA      hq.work. root.hq.work. (
6                             1          ; Serial
7                             604800     ; Refresh
8                             86400      ; Retry
9                             2419200    ; Expire
10                            604800 )    ; Negative Cache TTL
11;
12@         IN      NS       hq.work.
132         IN      PTR      HQ-SRV.hq.work.
141         IN      PTR      HQ-R.hq.work.
```

Рисунок 34 — настройка зоны обратного просмотра hq.work для ipv4

Где:

**PTR запись** — основная запись для зоны обратного просмотра

Третьим шагом настроим запись для зоны обратного просмотра для ipv6, для этого достаточно скопировать зону **hq\_arpa**, то есть

**cp /etc/bind/hq\_arpa /etc/bind/hq6\_arpa**

После создания всех конфигов необходимо перезагрузить службу bind9

**systemctl restart bind9** (лучше stop и start)

Похожая настройка выполняется для зоны **branch.work**

**Проверка выполняется посредством команд**

**host IP-адрес**

**host имя машины**

**Примечание:**

Не забывайте, что для br-srv по заданию нет PTR записи, её создание может считаться ошибкой

**2. Настройте синхронизацию времени между сетевыми устройствами по протоколу NTP.**

- a. В качестве сервера должен выступать роутер HQ-R со стратумом 5
- b. Используйте Loopback интерфейс на HQ-R, как источник сервера времени
- c. Все остальные устройства и сервера должны синхронизировать свое время с роутером HQ-R
- d. Все устройства и сервера настроены на московский часовой пояс (UTC +3)

Настройка производится на всех машинах, указанных в топологии, при этом настройка на машине, выступающей в роли NTP сервера уникальна, а на NTP клиентах идентична

Для начала на всех машинах необходимо установить московский часовой пояс, для этого следует воспользоваться командой

**timedatectl set-timezone Europe/Moscow**

Следующим шагом установим альтернативную службу NTP, под названием CHRONY, так как для задания 3, где происходит развёртывание домена, будет использоваться именно этот сервис. Устанавливаем с помощью команды:

**apt install chrony**

Произведём установку NTP сервиса Chrony

Далее следует осуществить настройку машины, выступающей в роли NTP сервера HQ-R, посредством команды

**nano /etc/chrony/chrony.conf**

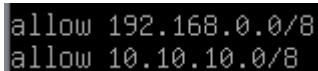
осуществим вход в конфигурацию chrony, где следует установить значения

как указано на рисунках 35 и 36



```
local stratum 5
```

Рисунок 35 — указание адреса NTP сервера с определённым стратумом



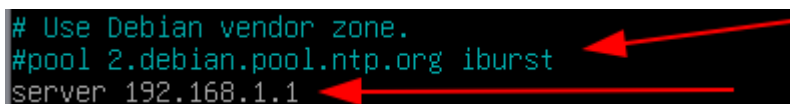
```
allow 192.168.0.0/8  
allow 10.10.10.0/8
```

Рисунок 36 — разрешение передачи NTP рассылок в указанной сети

**Примечание:** Нет необходимости указывать все сети которые присутствует в нашей сети, достаточно указать только одну сеть каждой машины , а так как у нас используется сети 192.168.0.0 , 192.168.1.0 и 192.168.2.0 , есть возможность взять сеть 192.168.0.0 с 22 маской которая будет включать в себя сеть начинающуюся с адреса 192.168.0.0 и заканчивающаяся адресом 192.168.3.255

Для настройки NTP клиентов chrony так же необходимо перейти в конфиг **nano /etc/chrony/chrony.conf**

И необходимо провести внести изменения в конфиг как указано на рисунке 37



```
# Use Debian vendor zone.  
#pool 2.debian.pool.ntp.org iburst  
server 192.168.1.1
```

Рисунок 37 — Настройка NTP клиентам chrony

Для проверки используйте команды **chronyc tracking** и **chronyc sources**

**3. Настройте сервер домена выбор, его типа обоснуйте, на базе HQ-SRV через web интерфейс, выбор технологий обоснуйте.**

**a. Введите машины BR-SRV и CLI в данный домен**

**b. Организуйте отслеживание подключения к домену**

В качестве домена может быть выбраны один из двух вариантов, или SAMBA DC, или FREEIPA реализованная через DOCKER, т.к. экспериментальные версии freeipa-server больше не поддерживаются для системы DEBIAN, однако DOCKER позволяет реализовать freeipa-server для любой системы. ДЛЯ настройки будет выбрана именно FreeIpa.

Первым делом необходимо установить докер, воспользовавшись

скриптом, который есть в открытом доступе, однако для этого нам необходимо экспортировать переменные окружения относящиеся к Proxy (Если Proxy отсутствует т. е. Пакеты с не стандартных репозиториях устанавливаются сами, то первый шаг можно пропустить)

Первым шагом необходимо посмотреть переменные, которые необходимо экспортировать, перейдя по пути

**nano /etc/apt/apt.conf.d/01proxy**

и посмотреть находящиеся там значения, после чего посредством команд

**export http\_proxy=http(или https):/(адрес:порт)**

**export https\_proxy=http(или https):/(адрес:порт)**

Экспортировать переменные прокси для доступа в интернет

ПРИМЕР:



```
Acquire::http::Proxy "http://10.0.70.52:3128";
Acquire::https::Proxy "http://10.0.70.52:3128";
# END ANSIBLE MANAGED BLOCK
```

Рисунок 38 — пример файла 01proxy

Команды для экспортирования переменных для конфига из рисунка 38

**export http\_proxy=http://10.0.70.52:3128**

**export https\_proxy=http://10.0.70.52:3128**

Вторым шагом посредством скрипта необходимо установить сам DOCKER, для этого необходимо ввести следующую команду

**wget -qO- https://get.docker.com | bash**

Вся установка происходит автоматически, и не должна выдавать ошибок, если были выполнены все предыдущие шаги

Третьим шагом необходимо запустить готовый контейнер с образом freeipa для centos-8-4.8.4 Для этого создаём каталог для автоматического запуска служб докера (Необходимо если вы делали шаги с Proxy ранее), командой

**mkdir -p /etc/systemd/system/docker.service.d**

Далее заходим в файл

**nano /etc/systemd/system/docker.service.d/http-proxy.conf**

и заполняем в соответствии с рисунком 39

```
[Service]
Environment="HTTP_PROXY=http://10.0.70.52:3128"
Environment="HTTPS_PROXY=http://10.0.70.52:3128"
```

Рисунок 39 — настройка прокси для docker.service

После чего перезапускаем демона и сам докер командами **в указанном порядке**

**systemctl daemon-reload**

и

**systemctl restart docker**

После чего запускаем команду

**docker pull freeipa/freeipa-server:centos-8-4.8.4**

После окончания пула контейнера необходимо создать директорию, в которую будет монтироваться контейнер посредством команды

**mkdir -p /var/lib/ipa-data**

Также необходимо внести изменения в загрузчик системы для указания, необходимости использования обеих версий cgroup (механизм по ограничению ресурсов, начиная с 11 Debian по умолчанию включена только 2 версия)

Для этого посредством команды заходим в загрузчик ядра

**nano /etc/default/grub**

После чего вносятся изменения как показаны на рисунке 40

```
GRUB_CMDLINE_LINUX="quiet systemd.unified_cgroup_hierarchy=0"
```

Рисунок 40 — Изменение параметров cgroup

Для применения изменений необходимо использовать команду

**grub-mkconfig -o /boot/grub/grub.cfg**

**После чего необходимо перезагрузить машину**

Следующим шагом уже переходим к запуску контейнера с хранящейся там FreeIPA, в качестве параметров ключей, указывает имя, указываем доменную сеть, а так открываем все необходимые для работы порты, указываем путь и образ, разрешаем конфликт с IPv6. Все параметры показаны на рисунке 41.

```
root@HQ-SRV:~# docker run --name freeipa-server -ti -h hq-srv.hq.work -p 80:80 -p 443:443 -p 389:389
-p 636:636 -p 88:88 -p 464:464 -p 88:88/udp -p 464:464/udp -p 123:123/udp --read-only --sysctl net.
ipv6.conf.all.disable_ipv6=0 -v /sys/fs/cgroup:/sys/fs/cgroup:rw -v /var/lib/ipa-data:/data:2 freei
pa/freeipa-server:centos-8-4.8.4
```

Рисунок 41 – запуск контейнера с указанием всех параметров

**Важное Примечание:** В случае завершения выполняемых функций в контейнере в результате которых оболочка может перейти в состояние freezing, или при успешном завершении, для выхода из оболочки окружения необходимо последовательно нажать сочетание клавиш **ctrl + p**, а затем **ctrl + q**. В случае если вам необходимо остановить контейнер можно воспользоваться командой **docker stop имя контейнера**, для удаления контейнера **docker rm имя контейнера**, для просмотра существующих образов **docker images**

**После успешного запуска необходимо заполнить форму:**

На вопрос о интеграции DNS нажимаем **Enter**

На вопрос о задании имени сервера нажимаем **Enter**

На вопрос о подтверждение имени домена нажимаем **Enter**

На вопрос о подтверждение имени области нажимаем **Enter**

На запрос ввода пароля для менеджера директорий вводим **P@ssw0rd**

На запрос ввода пароля для IPA админа вводим **P@ssw0rd**

На вопрос синхронизации с службой Chrony нажимаем **Enter**

На вопрос о конфигурирование системы с текущими параметрами вводим **yes**

Процесс установки достаточно длительный и может занимать около 5-10 или более минут.

После завершения установки необходимо подготовить машины, которые будут присоединены к домену. Для этого первым делом переходим по пути:

**Nano /etc/hosts**

И конфигурируем файл на клиенте как указано на рисунке 42

```
127.0.0.1      localhost
127.0.1.1      cli.hq.work      cli
192.168.1.2    hq-srv.hq.work
```

Рисунок 42 – конфигурация хостов машины CLI

Для машины BR-SRV настройка будет выглядеть как показано на рисунке

```
127.0.0.1      localhost
127.0.1.1      br-srv.branch.work    br-srv
192.168.1.2    hq-srv.hq.work
```

Рисунок 43 – конфигурация файла хостов машины BR-SRV

Следующим шагом посредством команды:

**apt install freeipa-client**

Производим установку клиентской части FreeIPA для ввода машины в домен.

На все всплывающие окна во время установки **нажимаем Enter**

После установки клиента, для ввода машины в домен необходимо прописать команды:

НА CLI

**ipa-client-install --mkhomedir --domain hq.work --server=hq-srv.hq.work -p admin -W**

НА BR-SRV

**ipa-client-install --mkhomedir --domain branch.work --server=hq-srv.hq.work -p admin -W**

```
root@BR-SRV:~# ipa-client-install --mkhomedir --domain brach.work --server=hq-srv.hq.work -p admin
W
```

Рисунок 43 – пример команды по вводу в домен на BR\_SRV

На сообщение о продолжении с фиксированными значения пишем **yes**

На вопрос о конфигурирование CHRONY нажимаем **ENTER**

На вопрос о конфигурировании с текущими значение пишем **yes**

Для проверки входа в FreeIPA, на клиентской машине необходимо открыть браузер и в адресной строке написать IP адрес машины HQ-SRV (192.168.1.2) логин и пароль для входа в вебку FreeIPA: admin и P@ssw0rd

**Важное Примечание:** если вы перезагрузите машину, то контейнер выключится, для его запуска можно воспользоваться командой **docker start freeipa-server**



4. Реализуйте файловый SMB или NFS (выбор обоснуйте) сервер на базе сервера HQ-SRV.

a. Должны быть опубликованы общие папки по названиям:

i. **Branch\_Files** - только для пользователя **Branch admin**;

ii. **Network** - только для пользователя **Network admin**;

iii. **Admin\_Files** - только для пользователя **Admin**;

b. Каждая папка должна монтироваться на всех серверах в папку **/mnt/<name\_folder>** (например, **/mnt/All\_files**) автоматически при входе доменного пользователя в систему и отключаться при его выходе из сессии. Монтироваться должны только доступные пользователю каталоги.

Исходя из поставленной задачи NFS будет более удачным выбором из-за его большей совместимости с системами Linux, при этом SMB крайне перегружена за счёт того, что создан для совместного использования широкого спектра сетевых ресурсов, включая службы файлов и печати, устройства хранения данных и хранилища виртуальных машин, в время как NFS, для совместного использования файлов и каталогов.

Поскольку файловый сервер будет работать на основе NFS , первым делом необходимо установить NFS сервер , посредством команды:

**apt install nfs-kernel-server**

Далее необходимо создать каталоги которые будут расшариваться.

**mkdir /mnt/all** — создание корневого каталога в котором будут храниться остальные

**mkdir /mnt/all/Branch\_Files** — каталог для пользователя **branch\_admin**

**mkdir /mnt/all/Network** — каталог для пользователя **network\_admin**

**mkdir /mnt/all/Admin\_Files** — каталог для пользователя **admin**

Так же для того что бы монтируемые директории не были пустыми , и был виден результат монтирования посредством команд

**touch /mnt/all/Branch\_Files/123**

**touch /mnt/all/Network/234**

**touch /mnt/all/Admin\_Files/345**

Создадим файлы с разными имена в директориях

Далее посредством команды

**nano /etc/exports**

Заходим в конфигурационный файл , где будут прописываться все общие ресурсы и их параметры и заполняем как показано на рисунке 44

```
/mnt/all/Branch_Files *(rw,async,no_subtree_check)
/mnt/all/Network *(rw,async,no_subtree_check)
/mnt/all/Admin_Files *(rw,async,no_subtree_check)
```

Рисунок 44 — создание общих ресурсов для пользователей

где:

/mnt/all/имя — Указание директории на сервере до которой будет выдан общий доступ

\* - указание IP адресов, которые имеют доступ в эту директорию (звёздочка значит все, так как по заданию не указано делать ограничения)

rw — разрешение на чтение и запись

async — включение обработки запросов клиента , до окончания предыдущего действия

no\_subtree\_check — отключает проверку вложенных директорий

Для экспорта всех общих ресурсов необходимо воспользоваться командой **exportfs -ra**

Также ещё одним необходимым шагом является создание доменных пользователей в Freeipa домене, для этого посредством адреса необходимо зайти в web-интерфейс Freeipa (адрес 192.168.1.2) , и сконфигурировать всех пользователей которые необходимы по заданию

**Важное примечание:** Пользователь **admin** является встроенной учётной записью и его конфигурировать не нужно.

Во вкладке users необходимо нажать кнопку add

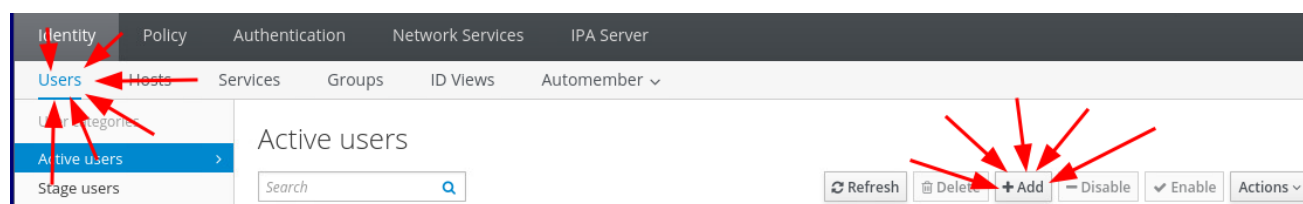


Рисунок 45 — переход к процессу создания пользователя

И в появившемся окне необходимо заполнить следующие данные:

**user login** — **network\_admin** или **branch\_admin**

**first name** — **Network Admin** или **Branch Admin**

**last name** - **Network Admin** или **Branch Admin**

**New Password** - 123

**Verify Password** -123

Пароль задаётся 123 , поскольку после захода в систему, необходимо будет сменить пароль

The screenshot shows a window titled "Add user" with a close button (X) in the top right corner. The window contains the following fields and controls:

- User login:** A text input field containing "branch\_admin". A red arrow points to this field.
- First name \*:** A text input field containing "Branch Admin". A red arrow points to this field.
- Last name \*:** A text input field containing "Branch Admin". A red arrow points to this field.
- Class:** An empty text input field.
- No private group:** A checkbox that is currently unchecked.
- GID:** A dropdown menu with a blue arrow icon on the right.
- New Password:** A text input field with three dots (•••) indicating a password. A red arrow points to this field.
- Verify Password:** A text input field with three dots (•••) indicating a password. A red arrow points to this field.
- \* Required field:** A label with a red arrow pointing to the "Verify Password" field.
- Buttons:** Four buttons are located at the bottom: "Add", "Add and Add Another", "Add and Edit", and "Cancel". Three red arrows point to the "Add and Edit" button.

Рисунок 46 — пример окна создания доменного пользователя branch\_admin

После ввода параметров необходимо нажать Add and Edit

и сконфигурировать параметры Login shell и Home directory. Пример конфигурирования для пользователя branch\_admin указан на рисунке 47

Job Title		User login	branch_admin
First name *	Branch admin	Password	*****
Last name *	Branch admin	Password expiration	2024-06-19 03:58:41Z
Full name *	Branch admin Branch admin	UID	36400015
Display name	Branch admin Branch admin	GID	36400015
Initials	BB	Principal alias	branch_admin@HQ.WORK <span>Delete</span>
GECOS	Branch admin Branch admin		<span>Add</span>
Class		Kerberos principal expiration	YYYY-MM-DD <span>hh</span> : <span>mm</span> UTC
		Login shell	/bin/bash
		Home directory	/home/branch_admin123

Рисунок 47 — Изменения параметров пользователя branch\_admin

где:

**Login shell** — изменения оболочки окружения в которую будем попадать при входе с sh(shell) на bash

**Home directory** — изменение домашней директории , необходимо так как иначе она будет совпадать с директориями локальных пользователей созданных на машинах

После этого можно перейти к настройке клиента , т. к. в задании указано что монтирование должно осуществляться при входе доменного пользователя , настройка будет проводится на машинах которые занесены в домен CLI и BR-SRV

Первым шагом необходимо установить NFS-клиент и Pam модуль для автоматического монтирования разделов при входе пользователя командой:

**apt install nfs-common libpam-mount**

Так же необходимо создать каталог куда будет проводится монтирование

**mkdir /mnt/all**

После чего перейдя по пути

**nano /etc/security/pam\_mount.conf.xml**

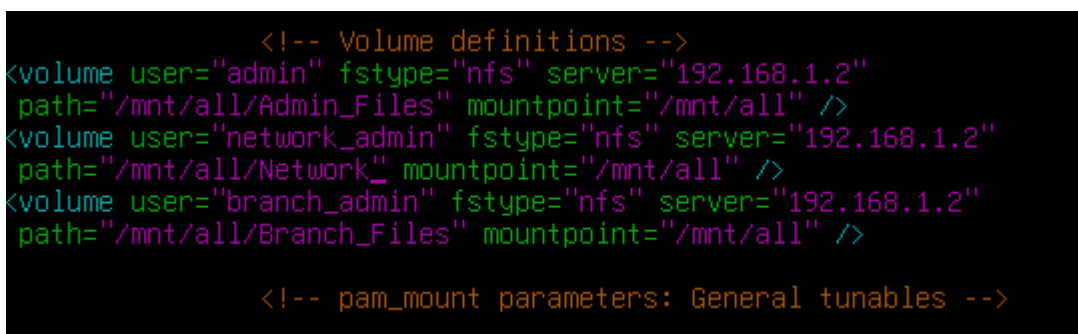
Необходимо добавить строки приведённые ниже в разделе <volume definitions>

```
<volume      user="admin"      fstype="nfs"      server="192.168.1.2"
path="/mnt/all/Admin_Files" mountpoint="/mnt/all" />
```

```

<volume user="branch_admin" fstype="nfs" server="192.168.1.2"
path="/mnt/all/Branch_Files" mountpoint="/mnt/all" />
<volume user="network" fstype="nfs" server="192.168.1.2"
path="/mnt/all/Network" mountpoint="/mnt/all" />

```



```

<!-- Volume definitions -->
<volume user="admin" fstype="nfs" server="192.168.1.2"
path="/mnt/all/Admin_Files" mountpoint="/mnt/all" />
<volume user="network_admin" fstype="nfs" server="192.168.1.2"
path="/mnt/all/Network_" mountpoint="/mnt/all" />
<volume user="branch_admin" fstype="nfs" server="192.168.1.2"
path="/mnt/all/Branch_Files" mountpoint="/mnt/all" />

<!-- pam_mount parameters: General tunables -->

```

Рисунок 48 — пример настройки ramlib для всех пользователей

Для проверки работы общих ресурсов необходимо зайти под доменным пользователем, для этого посредством команды

**sudo login**

переходим в окно для входа в систему

в вкладке Login указывается пользователь по шаблону :

**имя@домен**

Пример:

**branch\_admin@hq.work**

В вкладке Password вводится пароль, для пользователей branch\_admin и network\_admin необходимо будет ввести пароль по схеме:

Password: 123

Current Password: 123

New password: P@ssw0rd

Retype new password: P@ssw0rd

И зайдя в пользователя проверить содержимое папки /mnt/all на наличие созданных файлов

## 5. Сконфигурируйте веб-сервер LMS Apache на сервере BR-SRV:

**a. На главной странице должен отражаться номер места**

**b. Используйте базу данных mySQL**

**с. Создайте пользователей в соответствии с таблицей, пароли у всех пользователей «P@ssw0rd»**

Пользователя	Группа
Admin	Admin
Manager1	Manager
Manager2	Manager
Manager3	Manager
User1	WS
User2	WS
User3	WS
User4	WS
User5	TEAM
User6	TEAM
User7	TEAM

Вся настройка пунктов А и В будет выполняться исключительно на машине BR-SRV, для пункта С, а также проверки пункта А необходимо воспользоваться машиной CLI, так как на ней присутствует графика.

Первым шагом необходимо установить пакеты для веб-сервера АРАСНЕ и пакеты поддержки РНР, так как РНР, быстрее всего позволит создать страницу с номер места сдающего.

Для этого посредством команды

**Apt install apache2 libapache2-mod-php**

Устанавливаются пакеты для apache сервера и поддержки РНР сервером

Далее необходимо сконфигурировать страницу, которой в будущем заменится дефолтная страница АРАСНЕ, командой

**nano /var/www/html/mesto.php**

Создаётся страница, которую необходимо заполнить как указано на рисунке 49

```
GNU nano 7.2 /var/www/html/mesto.php
<?php
$fontSize = "200px";
echo "<div style=\"text-align:center\">";
print '<p style="font-size:' . htmlspecialchars($fontSize) . ';>5</p>';
?>
```

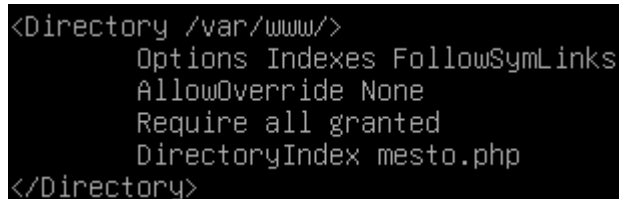
Рисунок 49 – Код РНР для создания страницы

Единственная часть кода, которую необходимо будет менять, это цифра 5, её будет необходимо заменить на номер своего места.

Следующим шагом необходимо заменить дефолтную страницу, для того что бы при обращении к серверу в качестве главной страницы, показывался номер места, для этого перейдя по пути

**nano /etc/apache2/apache2.conf**

Переходим в конфигурационный файл, и ищем и заполняем раздел который указан на рисунке 50



```
<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
    DirectoryIndex mesto.php
</Directory>
```

Рисунок 50 — замена главной страницы

Для проверки достаточно зайти на машину CLI, и в браузере прописать IP-адрес сервера, если ошибок допущено не было, должна быть выведена цифра по центру.

Далее переходим к установке базы данных , т. к. напрямую mysql-server установить не получится, будет использоваться пользовательский пакет, необходимо снова прописать команду для экспорта переменных которая была в задании 3.

Далее установим один из пакетов необходимых для работы mysql командой

**apt install gnupg**


Далее посредством команды указанной на рисунке ниже



```
root@dev:~# wget https://dev.mysql.com/get/mysql-apt-config_0.8.29-1_all.deb _
```

Рисунок 51 — указание места скачивание пакета

Указывается путь откуда будет скачиваться пакет, после чего для установки не user friendly пакетов, используется команда указанная на рисунке ниже



```
dpkg -i ./mysql-apt-config_0.8.29-1_all.deb _
```

Рисунок 52 — установка пользовательского пакета для mysql

После чего при установке в появившемся окне просто выбирается вариант ОК, как указано на рисунке ниже

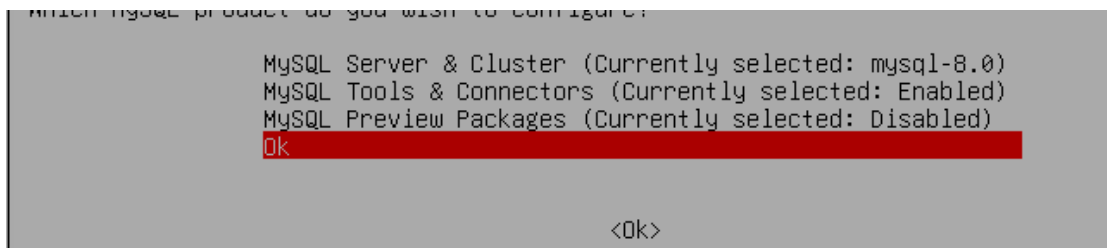


Рисунок 53 — согласие на установку предложенной конфигурации

Далее для обновления репозитория `mysql` необходимо прописать:

**`apt update`**

После успешного обновления, можно переходить к установке пакетов для `mysql`, для этого командой

**`apt install mysql-server php-mysql`**

Устанавливаются пакеты сервера, и его совместимости с `php`, второй из них пригодится чуть позже.

Во время установки будет предложено установить пароль, указывается пароль `R@ssw0rd`, на вопрос о плагине аутентификации выбирается первый вариант

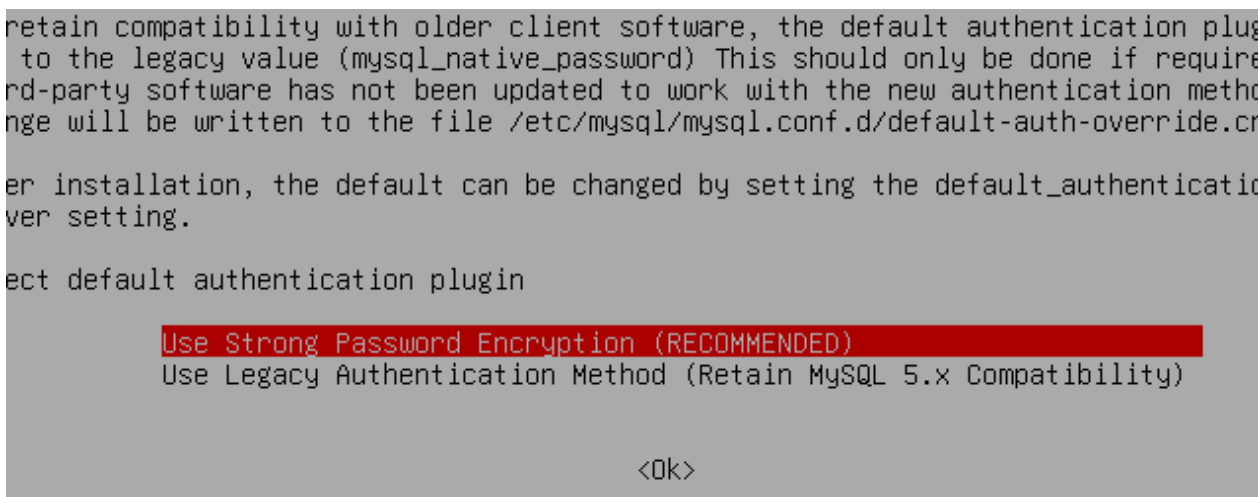


Рисунок 54 — выбор плагина аутентификации

Далее для создания пользователей и другим, можно установить веб-интерфейс для субд `mysql`, под названием `phpmyadmin`, этот шаг не обязателен, если вы самостоятельно можете создать пользователей и группы через консоль управления `mysql-server`.

Для установки `phpmyadmin`, необходимо воспользоваться командой:

**`apt install phpmyadmin`**

Во время установки:



На вопрос о выборе сервера для конфигурации нажимаем **Space (Пробел)** напротив Apache2, что бы появилась звёздочка . После чего **Enter**.



Рисунок 55 — выбор apache2 сервера.

На вопрос о конфигурирование БД для phpmyadmin выбирается вариант **YES**

Во всех вариантах где необходимо ввести пароль вводится пароль **P@ssw0rd**

Далее необходимо перейти по адресу

**IP-адрес сервера/phpmyadmin**

Пример

**192.168.2.2/phpmyadmin**

В окне авторизации:

**В поле Username вводится root (регистр имеет значение)**

**В поле Password вводится P@ssw0rd**

После успешной авторизации необходимо следовать инструкции на рисунках ниже:

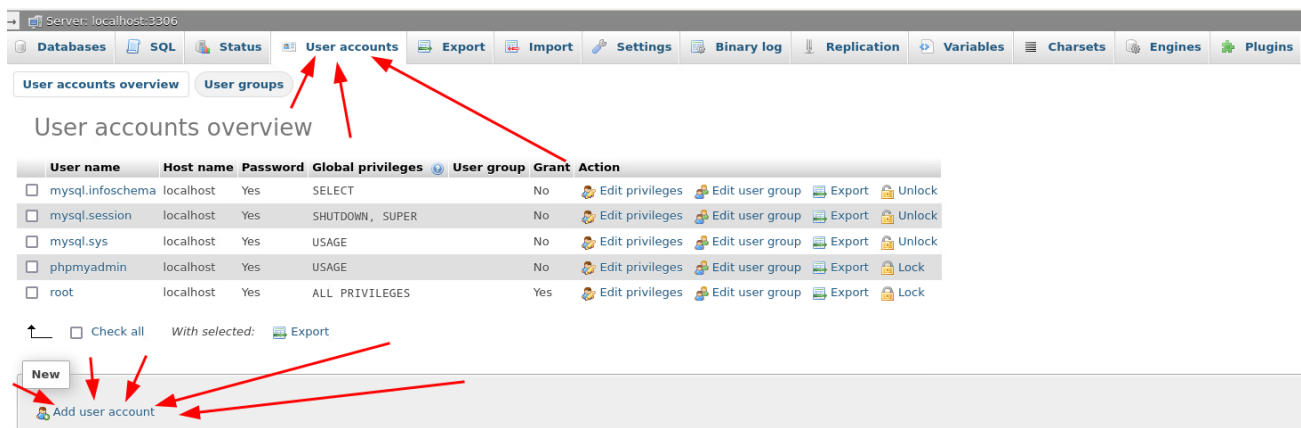


Рисунок 56 — переход к окну создания пользователя

В открывшемся окне, заполняются указанные на рисунке поля

## Add user account

**Login Information**

User name: Use text field

Host name: Any host

Password: Use text field

Re-type:

Authentication plugin: Caching sha2 authentication

Generate password:

**Database for user account**

☐ Create database with same name and grant all privileges.

☐ Grant all privileges on wildcard name (username\\_%).

**Global privileges** ☐ Check all

Note: MySQL privilege names are expressed in English.

☐ **Data**

☐ **Structure**

☐ **Administration**

**Resource limits**

Note: Setting these options to 0 (zero) removes the limit.

MAX QUERIES PER HOUR

MAX UPDATES PER HOUR

MAX CONNECTIONS PER HOUR

MAX USER\_CONNECTIONS

Рисунок 57 — пример создания пользователя Admin

После создания снова необходимо перейти в вкладку User accounts, как указано на рисунке 58

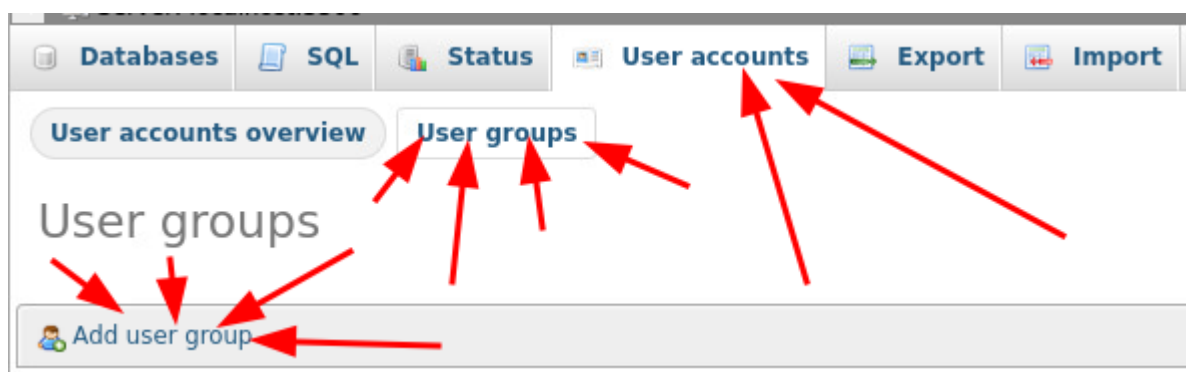


рисунок 58 — переход к вкладке создания пользовательских групп

В открывшемся окне указывается имя группы которая будет создана

User group menu assignments ☐ Check all

Group name: Admin

**Server-level tabs**

- ☐ Databases
- ☐ SQL
- ☐ Status
- ☐ Users
- ☐ Export
- ☐ Import
- ☐ Settings
- ☐ Binary log
- ☐ Replication
- ☐ Variables
- ☐ Charsets
- ☐ Plugins
- ☐ Engines

**Database-level tabs**

- ☐ Structure
- ☐ SQL
- ☐ Search
- ☐ Query
- ☐ Export
- ☐ Import
- ☐ Operations
- ☐ Privileges
- ☐ Routines
- ☐ Events
- ☐ Triggers
- ☐ Tracking
- ☐ Designer
- ☐ Central columns

**Table-level tabs**

- ☐ Browse
- ☐ Structure
- ☐ SQL
- ☐ Search
- ☐ Insert
- ☐ Export
- ☐ Import
- ☐ Privileges
- ☐ Operations
- ☐ Tracking
- ☐ Triggers

Go

Рисунок 59 — создание пользовательской группы Admin

Далее перейдя по пути как указано на рисунке ниже

Databases SQL Status User accounts Export Import Settings Binary log Replication Variables Charsets Engines Plugins

User accounts overview User groups

User accounts overview

User name	Host name	Password	Global privileges	User group	Grant	Action
<input type="checkbox"/> Admin	%	Yes	USAGE	No	<a href="#">Edit privileges</a>	<a href="#">Edit user group</a> <a href="#">Export</a> <a href="#">Lock</a>
<input type="checkbox"/> mysql.infoschema	localhost	Yes	SELECT	No	<a href="#">Edit privileges</a>	<a href="#">Edit user group</a> <a href="#">Export</a> <a href="#">Unlock</a>
<input type="checkbox"/> mysql.session	localhost	Yes	SHUTDOWN, SUPER	No	<a href="#">Edit privileges</a>	<a href="#">Edit user group</a> <a href="#">Export</a> <a href="#">Unlock</a>
<input type="checkbox"/> mysql.sys	localhost	Yes	USAGE	No	<a href="#">Edit privileges</a>	<a href="#">Edit user group</a> <a href="#">Export</a> <a href="#">Unlock</a>
<input type="checkbox"/> phpmyadmin	localhost	Yes	USAGE	No	<a href="#">Edit privileges</a>	<a href="#">Edit user group</a> <a href="#">Export</a> <a href="#">Lock</a>
<input type="checkbox"/> root	localhost	Yes	ALL PRIVILEGES	Yes	<a href="#">Edit privileges</a>	<a href="#">Edit user group</a> <a href="#">Export</a> <a href="#">Lock</a>

Check all With selected: Export

Рисунок 60 — переход к добавлению пользователя в группу

И в открывшемся окне, указывается группа

Edit user group

User group:

Admin

Close Save changes

Рисунок 61 — Внесение пользователя в группу.

**6. Запустите сервис MediaWiki используя docker на сервере HQ-SRV.**

**a. Установите Docker и Docker Compose.**

**b. Создайте в домашней директории пользователя файл wiki.yml для приложения MediaWiki:**

**i. Средствами docker compose должен создаваться стек контейнеров с приложением MediaWiki и базой данных**

**ii. Используйте два сервиса;**

**iii. Основной контейнер MediaWiki должен называться wiki и использовать образ mediawiki;**

**iv. Файл LocalSettings.php с корректными настройками должен находиться в домашней папке пользователя и автоматически монтироваться в образ;**

**v. Контейнер с базой данных должен называться db и использовать образ mysql;**

**vi. Он должен создавать базу с названием mediawiki, доступную по стандартному порту, для пользователя wiki с паролем DEP@ssw0rd;**

**vii. База должна храниться в отдельном volume с названием dbvolume.**

**MediaWiki должна быть доступна извне через порт 8080.**

Первым шагом необходимо установить docker compose, так как сам докер устанавливался в задании №3 второго модуля.

Для этого посредством команды, показанной на рисунке ниже, скачаем необходимый пакет.

```
root@HQ-SRV:~# curl -L "https://github.com/docker/compose/releases/download/v2.18.1/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

Рисунок 62 – скачивание пакета для docker-compose

Командой, указанной на рисунке ниже, выдаём необходимые права для скаченной службы

```
chmod +x /usr/local/bin/docker-compose
```

Рисунок 63 – выдача прав для docker-compose

Далее для того, чтобы с нуля не писать yml файл, можно скачать похожий по смыслу файл, приведённый на рисунке ниже (если будет запрещено, будете писать сами)

```
root@HQ-SRV:~# wget -L "https://raw.githubusercontent.com/pirate/wikipedia-mirror/master/docker-compose.mediawiki.yml" -O /home/admin/wiki.yml_
```

Рисунок 64 – скачивание yml файла для последующего изменения

После чего открываем скачанный файл по пути

**Nano /home/admin/wiki.yml**

И приводим к виду, указанному на рисунке ниже, **не удаляя присутствующие на рисунке закоментированные строки ! Соблюдая расстановку пробелов ! Заголовки первого порядка (Нажимаем один TAB или 2 пробела) , Второго порядка (2 TAB или 4 пробела), Третьего порядка (3 TAB или 6 пробелов).**

```
version: '3'
services:
  db:
    image: mysql
    environment:
      MYSQL_DATABASE: mediawiki
      MYSQL_USER: wiki
      MYSQL_PASSWORD: DEP@ssw0rd
      MYSQL_ROOT_PASSWORD: DEP@ssw0rd
    ports:
      - 3306:3306
    volumes:
      - /home/admin/dbvolume

  wiki:
    image: mediawiki
    ports:
      - 8080:80
    # volumes:
    #_ - /home/admin/LocalSettings.php:/var/www/html/LocalSettings.php
```

Рисунок 65 – создание yml файла с двумя контейнерами и указанием параметров

После чего запускаем контейнеры посредством команды

**docker-compose -f /home/admin/wiki.yml up**

После чего начнётся загрузка служб, после загрузки необходимо дождаться запуска контейнеров с сообщением о готовности подключения

```
♦ Container admin-wiki-1   Recreated   0.1s
♦ Container admin-db-1    Created      0.0s
Attaching to admin-db-1, admin-wiki-1
admin-db-1 | 2024-04-08 09:48:18+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.
8.0-1.el8 started.
admin-wiki-1 | AH00558: apache2: Could not reliably determine the server's fully qualified domain n
ame, using 172.18.0.3. Set the 'ServerName' directive globally to suppress this message
admin-wiki-1 | AH00558: apache2: Could not reliably determine the server's fully qualified domain n
ame, using 172.18.0.3. Set the 'ServerName' directive globally to suppress this message
admin-wiki-1 | [Mon Apr 08 09:48:18.468357 2024] [mpm_prefork:notice] [pid 1] AH00163: Apache/2.4.5
7 (Debian) PHP/8.1.27 configured -- resuming normal operations
admin-wiki-1 | [Mon Apr 08 09:48:18.469789 2024] [core:notice] [pid 1] AH00094: Command line: 'apac
he2 -D FOREGROUND'
admin-db-1 | 2024-04-08 09:48:18+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
admin-db-1 | 2024-04-08 09:48:18+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.
8.0-1.el8 started.
admin-db-1 | '/var/lib/mysql/mysql.sock' -> '/var/run/mysqld/mysqld.sock'
admin-db-1 | 2024-04-08T09:48:18.933570Z 0 [System] [MY-015015] [Server] MySQL Server - start.
admin-db-1 | 2024-04-08T09:48:19.325540Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld
8.3.0) starting as process 1
admin-db-1 | 2024-04-08T09:48:19.339953Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization ha
s started.
admin-db-1 | 2024-04-08T09:48:20.224975Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization ha
s ended.
admin-db-1 | 2024-04-08T09:48:20.669031Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem i
s self signed.
admin-db-1 | 2024-04-08T09:48:20.669217Z 0 [System] [MY-013602] [Server] Channel mysql_main confi
gured to support TLS. Encrypted connections are now supported for this channel.
admin-db-1 | 2024-04-08T09:48:20.674586Z 0 [Warning] [MY-011810] [Server] Insecure configuration
for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users. Consider chos
ing a different directory.
admin-db-1 | 2024-04-08T09:48:20.736331Z 0 [System] [MY-011323] [Server] X Plugin ready for connec
tions. Bind-address: '::' port: 33060, socket: /var/run/mysqld/mysqldx.sock
admin-db-1 | 2024-04-08T09:48:20.737922Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready
for connections. Version: '8.3.0' socket: '/var/run/mysqld/mysqld.sock' port: 3306 MySQL Communit
y Server - GPL.
```

Рисунок 66 — Пример окна с запущенными контейнерами служб

Далее необходимо перейти на машину CLI , и в браузере перейти по адре-  
су **192.168.1.2:8080**



Рисунок 67 — Стартовая страница MediaWiki с отсутствующим файлом настроек

Перейди по ссылке необходимо нажать → **Continue**

Затем внизу страницы снова → **Continue**

Далее на следующей странице необходимо указать настройки по заданию ,  
как указано на рисунке ниже. **Пароль DEP@ssw0rd**

The screenshot shows the 'MariaDB/MySQL settings' page of a MediaWiki installation. At the top, there are radio buttons for 'MariaDB, MySQL, or compatible' (selected) and 'SQLite'. Below this is the 'Database host:' section with a text input field containing 'db' and a checkbox for 'Connect over SSL'. The next section is 'Identify this wiki', containing 'Database name (no hyphens):' with a text input field containing 'mediawiki', and 'Database table prefix (no hyphens):' with an empty text input field. The final section is 'User account for installation', containing 'Database username:' with a text input field containing 'wiki', and 'Database password:' with a password input field showing dots. At the bottom right are two buttons: '← Back' and 'Continue →'. Red arrows originate from a single point at the top right and point to the 'db' field, the 'mediawiki' field, the 'wiki' field, the password field, and the 'Continue →' button.

Рисунок 68 — указание настроек БД для MediaWiki

На следующей странице нажимаем → **Continue**

Далее на следующей странице, заполняем поля как указано на рисунке ниже, обязательно не забыв поставить галочку о том что вы очень занятой. **Пароль DEP@ssw0rd**

**Name of wiki:**  
[help](#)

**Project namespace:**  
[help](#)  
☒ Same as the wiki name: Wiki  
☐ Project  
☐ Other (specify)

**Administrator account**

**Your username:**  
[help](#)

**Password:**

**Password again:**

**Email address:**  
[help](#)  
  
☐ [Subscribe to the release announcements mailing list.](#)  
☐ [Share data about this installation with MediaWiki developers. \[Privacy policy.\]\(#\)](#)

**You are almost done!** You can now skip the remaining configuration and install the wiki right now.

☐ Ask me more questions.  
☒ I'm bored already, just install the wiki.

[← Back](#) [Continue →](#)

Рисунок 69 — заполнение данных для работы в MediaWiki

После чего сконфигурированный файл автоматически будет скачен в загрузки

Далее его необходимо перенести на сервер. Если вы выполнили задание с запретом доступа по SSH с машины CLI, файл необходимо будет кидать не напрямую а через промежуточную машину HQ-R

Для этого воспользовавшись командами

**На машине CLI от юзера админ (У вас может быть другой пользователь в зависимости от кого вы авторизовались в систему):**

```
scp /home/admin/Downloads/LocalSettings.php root@192.168.1.1:/home/admin
```

На машине HQ-R:

```
scp /home/admin/LocalSettings.php admin@192.168.1.2:/home/admin/
```

После чего необходимо на машине HQ-SRV перейти по пути

```
nano /home/admin/wiki.yml
```

**И раскоментить и переписать (если они у вас отличаются) строки указанные на рисунке ниже**

```
volumes:
  - /home/admin/LocalSettings.php:/var/www/html/LocalSettings.php
```

Рисунок 70 — Подключение файла настроек к MediaWiki



## После чего снова запустить контейнеры.

И теперь перейдя на машину CLI и зайдя в браузер по тому же адресу. Должна загрузиться главная страница MediaWiki

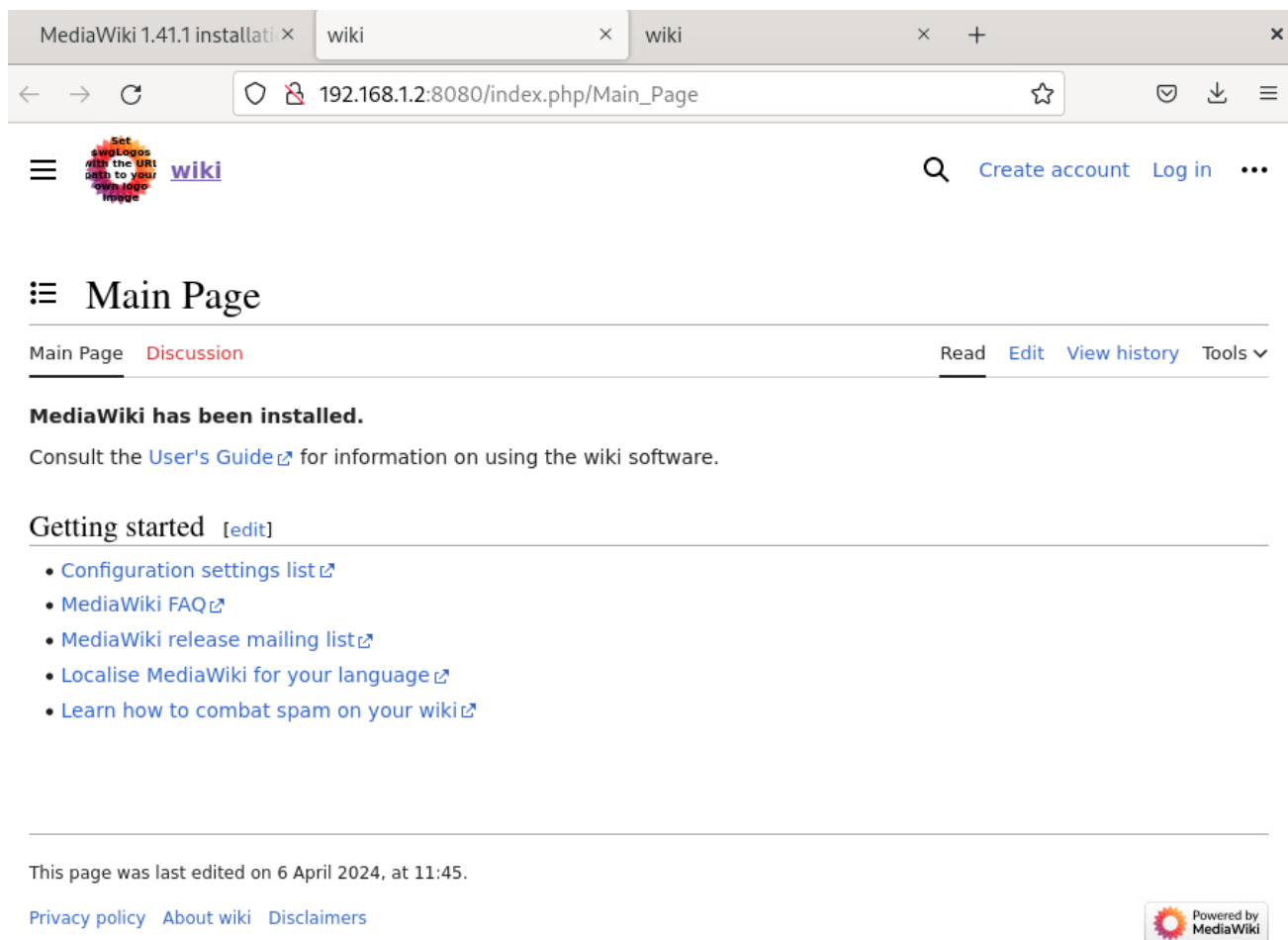


Рисунок 71 — главная страница «вашей» wiki

**Автор:**

**Ложников Н.С.**

**Консультант:**

**Шукуров Э.А.**

**Тестеры:**

**Козин Н.С.**

**Коновалов И.А.**

