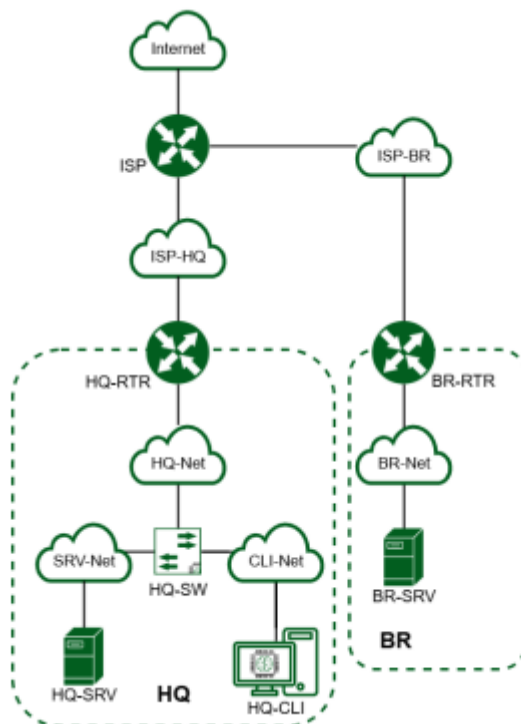


Обновлено 09.04.2024 V1.3



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.

Для корректной работы сети используйте NMTUI только на машине ISP. На остальных машинах настройку IP-адресации производите через файл конфигурации /etc/network/interfaces

Задание 1 модуля 1

1. Произведите базовую настройку устройств

- Настройте имена устройств согласно топологии. Используйте полное доменное имя

Примечание: для выполнения данного задания необходимо постоянное изменение имени каждого устройства, указанного на топологии (временно

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

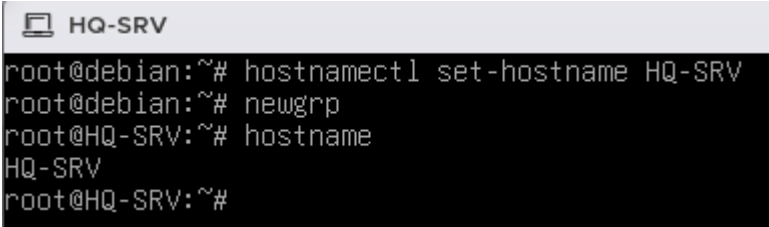


Рисунок 1 — Пример изменения имени устройства

- 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

Имя устройства	IP
HQ-CLI	192.168.200.2 255.255.255.240 — к HQ-RTR
ISP	172.16.4.1 255.255.255.240 — к HQ-R 172.16.5.1 255.255.255.240 — к BR-R
HQ-RTR	192.168.100.1 255.255.255.192 — к HQ-SRV 172.16.4.2 255.255.255.240 — к ISP 192.168.200.1 255.255.255.240 — к HQ-CLI
HQ-SRV	192.168.100.2 255.255.255.192 — к HQ-RTR

BR-RTR	192.168.0.1 255.255.255.224 — к BR-SRV 172.16.5.2 255.255.255.240 — к ISP
BR-SRV	192.168.0.2 255.255.255.224 — к BR-RTR
Сеть управления (VLAN 999)	192.168.999.0 255.255.255.248

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

Первый способ: через network-manager

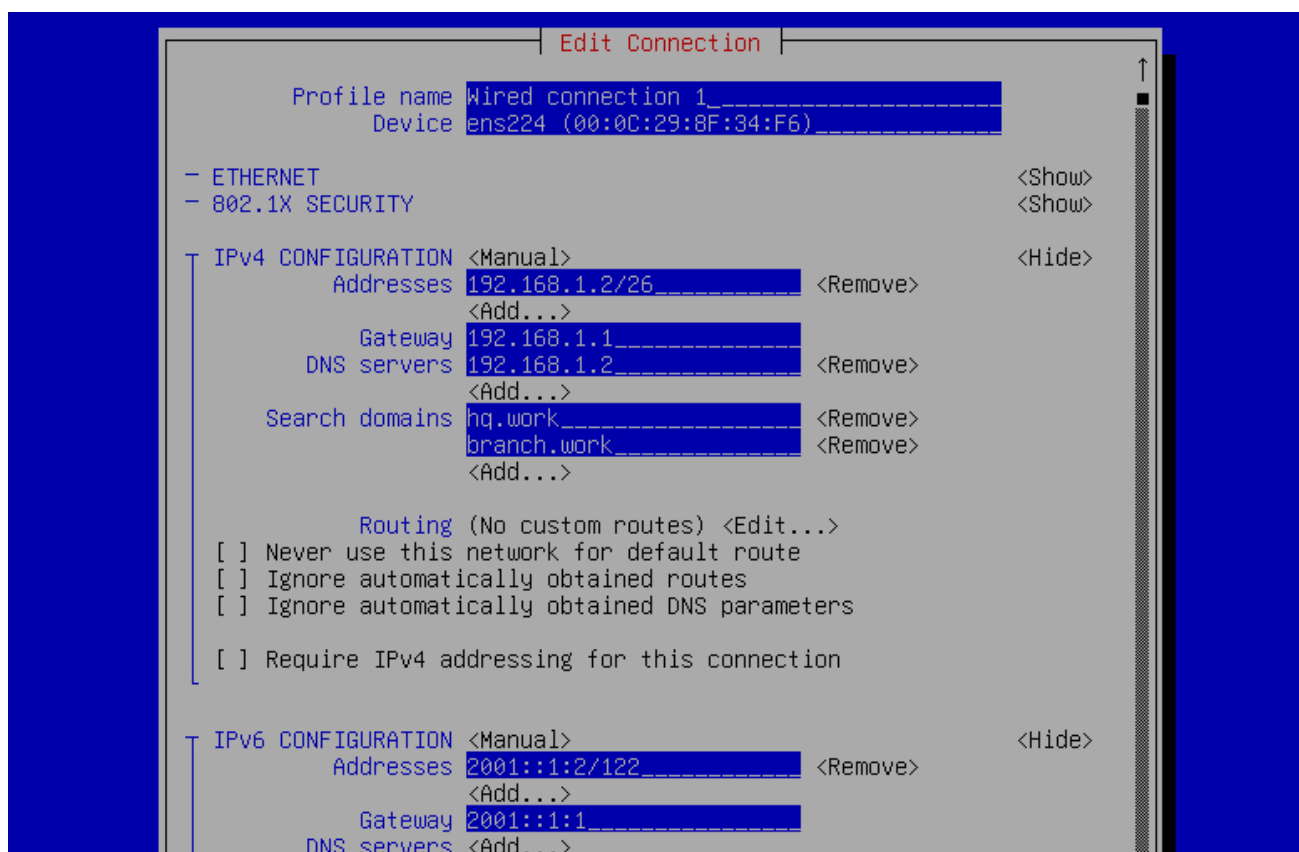


Рисунок 3 — Пример настройки IPv4 и IPv6 на HQ-SRV

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

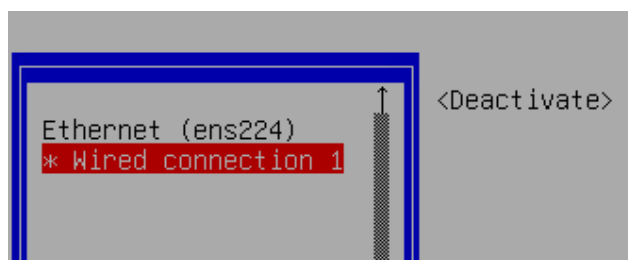


Рисунок 4 — перезагрузка интерфейсов

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

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5)

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

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

auto ens224
iface ens224 inet static
address 172.16.4.1
netmask 255.255.255.240

auto ens256
iface ens256 inet static
address 172.16.5.1
netmask 255.255.255.240
```

Рисунок 6 — Пример настройки интерфейсов ISP

Для настройки VLAN на роутере HQ-RTR нужно скачать утилиту VLAN:

`apt install vlan`. Также нужно установить модуль 8021q:

`modprobe 8021q`

и добавить его в автозапуск `echo 8021q >> /etc/modules`

Теперь можно приступить к настройке файла конфигурации `/etc/network/interfaces`.

Он должен выглядеть следующим образом:

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug ens192
iface ens192 inet static
address 172.16.4.2
netmask 255.255.255.240
gateway 172.16.4.1

auto ens224
iface ens224 inet static
address 102.168.100.1
netmask 255.255.255.192

auto ens224:1
iface ens224:1 inet static
address 102.168.200.1
netmask 255.255.255.240

auto ens224.100
iface ens224 inet static
address 102.168.100.3
netmask 255.255.255.192
vlan-raw-device ens224

auto ens224.200
iface ens224 inet static
address 102.168.200.3
netmask 255.255.255.240
vlan-raw-device ens224:1
```

Рисунок 7 — Настройка интерфейсов HQ-RTR

2) Настройка ISP

iptables:

apt-get install iptables iptables-persistent

Затем нужно создать правила iptables

```
iptables -t nat -A POSTROUTING -s 172.16.4.0/28 -o ens192 -j MASQUERADE
```

```
iptables -t nat -A POSTROUTING -s 172.16.5.0/28 -o ens192 -j MASQUERADE
```

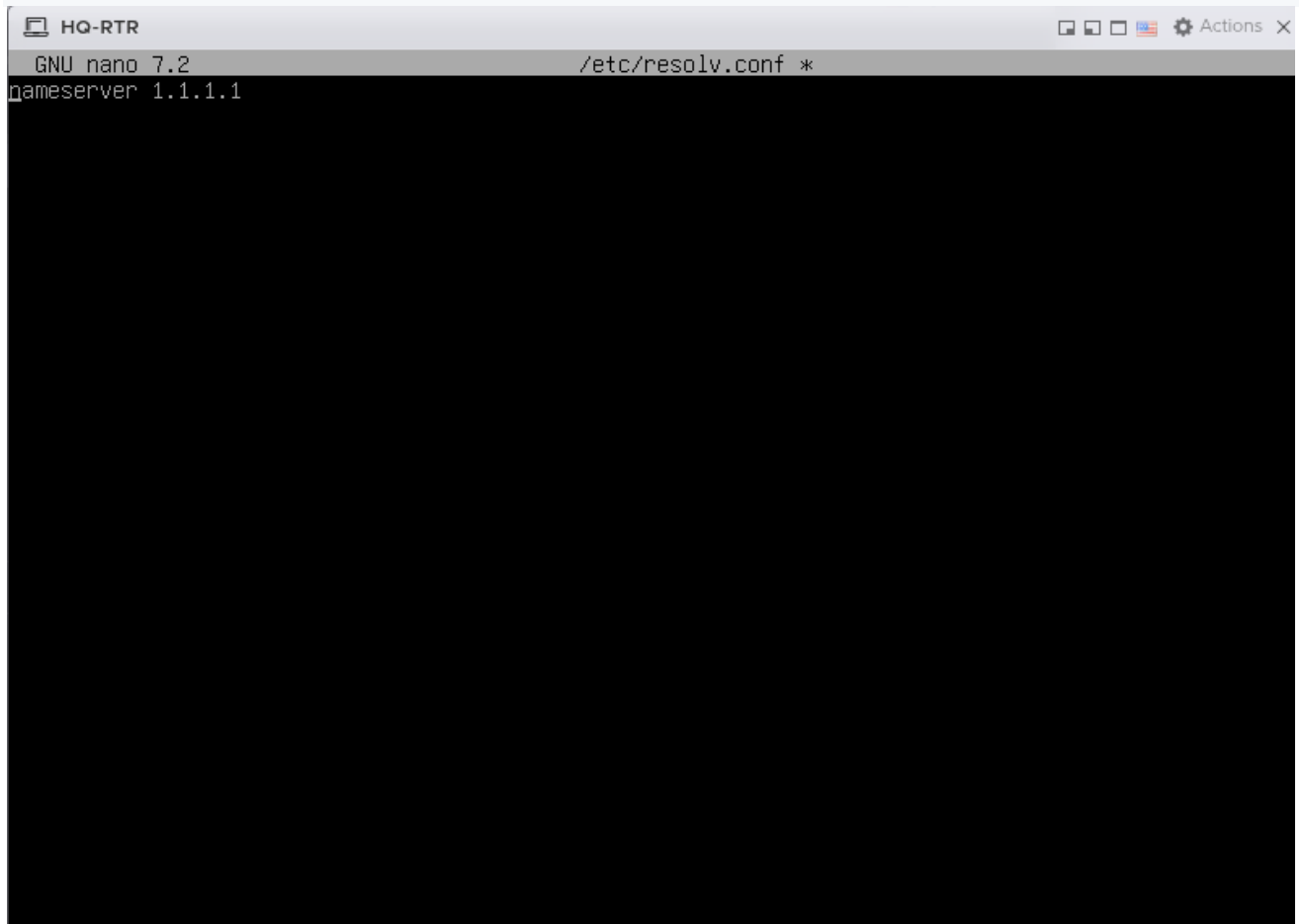
```
iptables-save > /etc/iptables/rules.v4
```

Перезапускаем iptables: `systemctl restart iptables`

Для проверки вводим команду **iptables -L -t nat**

После настройки на интерфейсах ISP может слететь Ip. Также на роутерах и ISP нужно зайти в файл `/etc/sysctl.conf` и раскомментировать строку «**net.ipv4.ip_forward=0**» и привести её к виду «**net.ipv4.ip_forward=1**». Также для работы nat и доступа в интернет на роутерах в качестве gateway указать адрес ISP.

На HQ-RTR и BR-RTR Нужно зайти в файл `/etc/resolv.conf` и оставить там только одну строку: `nameserver 1.1.1.1`



The screenshot shows a terminal window titled 'HQ-RTR'. The top bar indicates 'GNU nano 7.2' and the file being edited is '/etc/resolv.conf *'. The content of the file is 'nameserver 1.1.1.1'. The terminal background is black, and the text is white.

Рисунок 8 — Пример настройки интерфейсов ISP

Правила для HQ-RTR

```
iptables -t nat -A POSTROUTING -s 192.168.100.0/26 -o ens192 -j MASQUERADE
```

3. Создание локальных учётных записей

adduser sshuser

Затем появится поле ввода пароля как показано на рисунке 9

```
root@hq-cli:/home/locadm# useradd sshuser -u 1010 -U
root@hq-cli:/home/locadm# passwd sshuser
New password:
Retype new password:
passwd: password updated successfully
root@hq-cli:/home/locadm#
```

Рисунок 9 — окно ввода пароля при создании пользователя

Для смены id используется команда **usermode -u 1010 sshuser**

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 10

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

Рисунок 10 — выдача Root прав пользователям

5. Настройка безопасного удалённого доступа

Первым делом необходимо перейти по пути **nano /etc/ssh/sshd_config** где в окне конфигурации нам необходимо на HQ-SRV найти строки и изменить значения как указанно на рисунке 10

```
Port 2024
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
AllowUsers sshuser
#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
```

Рисунок 10— смена порта доступа по ssh и права подключения только определённого пользователя

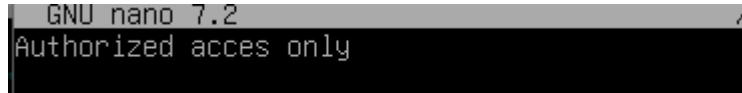
```
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
MaxAuthTries 2
#MaxSessions 10
```

Рисунок 11— Ограничение попыток авторизации

```
# no default banner path
Banner /etc/ssh-banner
```


Рисунок 12— Указание файла баннера

Для настройки баннера нужно зайти в файл `/etc/ssh-banner` и написать следующее: `Authorized acces only`



```
GNU nano 7.2 /etc/ssh-banner
Authorized acces only
```

Рисунок 13— баннера

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

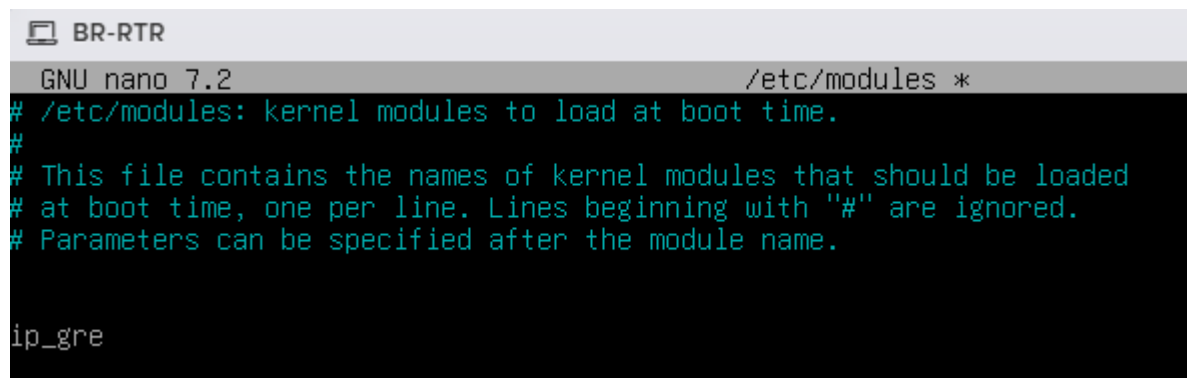
Для проверки доступа нужно написать команду: `ssh sshuser@192.168.100.2 -p 2024`

Где `-p` — указание порта. Без указания порта подключиться не получится

Также, при неправильном вводе пароля должно вывестись сообщение баннера

6) Реализация GRE-туннеля между офисами

Нужно зайти в файл `/etc/modules` и добавить там строку `ip_gre`:



```
BR-RTR
GNU nano 7.2 /etc/modules
# /etc/modules: kernel modules to load at boot time.
#
# This file contains the names of kernel modules that should be loaded
# at boot time, one per line. Lines beginning with "#" are ignored.
# Parameters can be specified after the module name.
ip_gre
```

Вся последующая настройка проводится в файле `/etc/network/interfaces`

```
auto tun1
iface tun1 inet tunnel
address 10.10.0.1
netmask 255.255.255.252
mode gre
local 172.16.4.2
endpoint 172.16.5.2
ttl 64
```

Рисунок 14 - Настройка GRE на HQ-RTR

```
auto tun1
iface tun1 inet tunnel
address 10.10.0.2
netmask 255.255.255.252
mode gre
local 172.16.5.2
endpoint 172.16.4.2
ttl 64_
```

Рисунок 15 - Настройка GRE на BR-RTR

Ping 10.10.0.1 и ping 10.10.0.2 для проверки работоспособности туннеля с обеих сторон:

```
root@br-rtr:~# ping 10.10.0.1
PING 10.10.0.1 (10.10.0.1) 56(84) bytes of data.
64 bytes from 10.10.0.1: icmp_seq=1 ttl=64 time=0.972 ms
64 bytes from 10.10.0.1: icmp_seq=2 ttl=64 time=0.690 ms
64 bytes from 10.10.0.1: icmp_seq=3 ttl=64 time=1.10 ms
64 bytes from 10.10.0.1: icmp_seq=4 ttl=64 time=0.509 ms
^C
--- 10.10.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.509/0.817/1.098/0.231 ms
root@br-rtr:~#
```

Рисунок 16 – Проверка работоспособности

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

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

```
bgpd=no  
ospfd=yes  
ospf6d=no
```

Рисунок 17 — 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 18 — 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

```
rtr.au-team.irpo# conf t  
rtr.au-team.irpo(config)# router ospf  
rtr.au-team.irpo(config-router)# passive-interface default  
rtr.au-team.irpo(config-router)# network 192.168.100.0/26 area 0  
rtr.au-team.irpo(config-router)# network 192.168.200.0/28 area 0  
rtr.au-team.irpo(config-router)# network 10.10.0.0/30 area 0  
rtr.au-team.irpo(config-router)#
```

Рисунок 19 — пример настройки OSPF на HQ-RTR

```
br-rtr.au-team.irpo(config-router)# area 0 authentication
br-rtr.au-team.irpo(config-router)# c
br-rtr.au-team.irpo(config-router)# exit
br-rtr.au-team.irpo(config)# interface tun1
br-rtr.au-team.irpo(config-if)# no ip ospf passive
br-rtr.au-team.irpo(config-if)# ip ospf authentication
br-rtr.au-team.irpo(config-if)# ip ospf authentication-key password
br-rtr.au-team.irpo(config-if)#
```

Рисунок 20— Включение авторизации и последующая настройка интерфейса

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

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 раскоментить 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 21 — 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**

Идентичная настройка проводится на BR-RTR, только указывается

другая подсеть(192.168.0.0/27). Указание сети туннеля и настройки авторизации абсолютно идентичны

8) Настройка динамической трансляции адресов

Точно также как и для ISP устанавливаем пакеты iptables:

`apt install iptables iptables-persistent`

```
iptables -t nat -A POSTROUTING -s 192.168.100.0/26 -o ens192 -j  
MASQUERADE
```

```
iptables -t nat -A POSTROUTING -s 192.168.200.0/28 -o ens192 -j  
MASQUERADE
```

Правило для BR-RTR

```
iptables -t nat -A POSTROUTING -s 192.168.0.0/27 -o ens192 -j  
MASQUERADE
```

Затем эти правила нужно сохранить: `iptables-save > /etc/iptables/rules.v4`

После этого перезапускаем службу

9.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. Строка v6 закоментирована, чтобы DHCP даже не думал пробовать его раздавать

```
INTERFACESv4="ens224"  
#INTERFACESv6=""
```

Рисунок 22 — Указание интерфейса для передачи адреса

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

```
subnet 192.168.200.0 netmask 255.255.255.240 {  
    range 192.168.200.4 192.168.200.14;  
    option domain-name-servers 192.168.100.2;  
    option domain-name "au-team.irpo";  
    option routers 192.168.200.1;  
    default-lease-time 600;  
    max-lease-time 7200;
```

Рисунок 23— 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

Перезапускаем службу DHCP: *systemctl restart isc-dhcp-server*

Для проверки на HQ-CLI нужно указать получение адреса по DHCP

```
|allow-hotplug ens192  
|iface ens192 inet dhcp  
|address 192.168.200.2  
|netmask 255.255.255.240  
|gateway 192.168.200.1
```

Рисунок 24 — Настройка интерфейса

Прописываем **systemctl restart networking** для применения и проверяем выданный

ip-адрес командой `ip -c a`

```
root@hq-cli:/home/locadm# ip -c 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: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:13:a6:91 brd ff:ff:ff:ff:ff:ff
    altname enp11s0
    inet 192.168.200.5/28 brd 192.168.200.15 scope global dynamic ens192
        valid_lft 597sec preferred_lft 597sec
    inet6 fe80::20c:29ff:fe13:a691/64 scope link
        valid_lft forever preferred_lft forever
```

Рисунок 25 — Проверка выдачи ip-адреса

10. Настройте DNS-сервер на сервере HQ-SRV:

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`)

Следующим шагом необходимо создать зоны для прямого и обратного просмотра dns

Для этого переходим по пути **nano /etc/bind/named.conf.default-zones** и создаём зоны как показано на скриншотах ниже

```

zone "au-team.irpo" {
    type master;
    file "/etc/bind/au-team.irpo";
};

zone "100.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/au-team.irpo_obr";
};

zone "200.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/au-team.irpo_hqobr";
};

```

Рисунок 26 — зоны для hq.work(На скриншоте указаны 2 обратные зоны, т. к. у HQ-CLI и HQ-SRV IP-адреса заканчиваются на одинаковые октеты и из-за этого DNS может не работать)

где:

zone — создаваемая зона

type — выбор между первичным и вторичным dns. (Master и Slave)

file — расположение конфигурационного файла зоны

allow-update — разрешение динамических обновлений

где zone:

hq.work — зона прямого просмотра

in-addr.arpa — зона обратного просмотра ipv4

Следующим шагом необходимо создать конфигурационные файлы для наших зон. Это можно сделать, скопировав стандартные шаблоны командой **cp**

Пример:

cp /etc/bind/db.local /etc/bind/au-team.irpo — создание файла для прямой зоны

cp /etc/bind/db.127 /etc/bind/ au-team.irpo_obr — создание обратной зоны ipv4

Первым шагом сконфигурируем зону прямого просмотра, переходим по пути

nano /etc/bind/au-team.irpo и конфигурируем файл как показано на

скриншоте ниже

```
GNU nano 7.2 /etc/bind/au-team.irpo
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      au-team.irpo. root.au-team.irpo. (
                        2      ; Serial
                        604800 ; Refresh
                        86400  ; Retry
                        2419200; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS       au-team.irpo.
@         IN      A        192.168.100.2
hq-rtr    IN      A        192.168.100.1
hq-srv    IN      A        192.168.100.2
hq-cli    IN      A        192.168.200.2
br-rtr    IN      A        192.168.0.1
br-srv    IN      A        192.168.0.2
moodle    CNAME    hq-rtr.au-team.irpo
wiki      CNAME    hq-rtr.au-team.irpo
```

Рисунок 27 — зона прямого просмотра

Где:

NS запись — обозначение сервера ответственного за разрешение запросов к dns

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

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/au-team.irpo_obr

```
GNU nano 7.2 /etc/bind/au-team.irpo_obr
;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      au-team.irpo. root.au-team.irpo. (
                        1      ; Serial
                        604800  ; Refresh
                        86400   ; Retry
                        2419200 ; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS       au-team.irpo.
1         IN      PTR      hq-rtr.au-team.irpo.
2         IN      PTR      hq-srv.au-team.irpo.
```

Рисунок 28— настройка зоны обратного просмотра hq.work для ipv4

```
HQ-SRV
GNU nano 7.2 /etc/bind/au-team.irpo_hqobr
;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      au-team.irpo. root.au-team.irpo. (
                        1      ; Serial
                        604800  ; Refresh
                        86400   ; Retry
                        2419200 ; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS       au-team.irpo.
2         IN      PTR      hq-cli.au-team.irpo.
```

Рисунок 29— настройка второй зоны обратного просмотра hq.work для ipv4

Где:

PTR запись — основная запись для зоны обратного просмотра

Проверка выполняется посредством команд

host IP-адрес

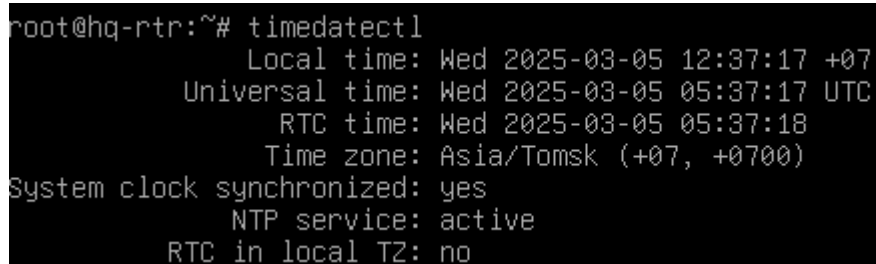
host имя машины

Задание 11

Настройка даты и времени согласно месту проведения экзамена

timedatectl set-timezone Asia/Tomsk

Команда для проверки: timedatectl



```
root@hq-rtr:~# timedatectl
          Local time: Wed 2025-03-05 12:37:17 +07
          Universal time: Wed 2025-03-05 05:37:17 UTC
             RTC time: Wed 2025-03-05 05:37:18
            Time zone: Asia/Tomsk (+07, +0700)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

Рисунок 30 — Проверка Даты и времени

Для выполнения задания будет использоваться утилита MDADM. Её нужно установить: `apt install mdadm`.

Далее прописываем команду `lsblk` для отображения дисков(Рисунок 31)

```
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
sdc       8:32   0    1G  0 disk
sdd       8:48   0    1G  0 disk
sr0      11:0    1 1024M  0 rom
root@hq-srv:~#
```

Рисунок 31 — Отображение дисков

Для создания рейда будут использоваться диски `sdb`, `sdc` и `sdd`. Для начала на них нужно создать разделы командой `fdisk /dev/sdb`. Сначала вводим `g`, чтобы создать раздел, затем вводим `n` и прокликаем `enter`. Для выхода и сохранения вводим `w`. (Рисунок 32)

```
root@hq-srv:~# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.38.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS (MBR) disklabel with disk identifier 0x1eb6119c.

Command (m for help): g
Created a new GPT disklabel (GUID: FB594CE0-3614-054F-A03B-4F160E70E73A).

Command (m for help): n
Partition number (1-128, default 1):
First sector (2048-2097118, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097118, default 2095103):

Created a new partition 1 of type 'Linux filesystem' and of size 1022 MiB.

Command (m for help):
```

Рисунок 32 — создание разделов

Всё тоже самое проделываем с дисками `sdc` и `sdd`.

Теперь можно приступить к созданию рейд-массива(Рисунок 33):

```
mdadm --create --verbose /dev/md0 -l 5 -n 3 /dev/sdb1 /dev/sdc/1 /dev/sdd1
```

```

root@hq-srv:~# mdadm --create --verbose /dev/md0 -l 5 -n 3 /dev/sdb1 /dev/sdc1 /dev/sdd1
mdadm: layout defaults to left-symmetric
mdadm: layout defaults to left-symmetric
mdadm: chunk size defaults to 512K
mdadm: size set to 1044480K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
root@hq-srv:~# _

```

Рисунок 33 — создание массива

Теперь можно приступить к созданию файловой системы. Для начала отформатируем разделы в ext4: `mkfs.ext4 /dev/md0` (Рисунок 34)

```

root@hq-srv:~# mkfs.ext4 /dev/md0
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 522240 4k blocks and 130560 inodes
Filesystem UUID: e4c26943-d6ce-4fc9-9a6e-ba7d75aef704
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

```

Рисунок 34 — форматирование

Следующим шагом нужно создать директорию `/raid5` в корне системы:

`mkdir /raid5` и примонтировать её к рейду: `mount /dev/md0 /raid5`.

Для автоматического монтажа необходимо зайти в `/etc/fstab` и привести его к виду, показанному на рисунке 35:

```

# / was on /dev/sda1 during installation
UUID=248b28d7-738b-49db-af61-5a84148f1756 / ext4 errors=remount-ro 0 1
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0
/dev/md0 /raid5 ext4 defaults 0 0

```

Рисунок 35 — настройка автомонтирования

Теперь устанавливаем серверную часть nfs: `apt install nfs-kernel-server`.

После установки нужно создать общую папку: `mkdir /raid/nfs` и задать права: `chmod -R 777 /raid/nfs`.

Переходим в файл `/etc/exports` и добавляем строчку, как показано на рисунке 36.

```

# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes      hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4       gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
/raid5/nfs       192.168.200.0/28(rw,sync,no_root_squash,subtree_check)

```

Рисунок 36 — создание папки, доступной для сети

Выполняем экспорт данных: `exportfs -a`

Теперь устанавливаем клиентскую часть `nfs` на HQ-CLI:

`apt install nfs-common.`

Создаём директорию: `mkdir /mnt/nfs` и монтируем её:

`mount -t nfs 192.168.100.2:/raid5/nfs /mnt/nfs`

Для автоматического монтирования заходим в `/etc/fstab` и приводим его к виду, показанному на рисунке 37

```

UUID=248b28d7-738b-49db-af61-5a84148f1756 /          ext4      errors=remount-ro 0 1
/dev/sr0      /media/cdrom0  udf,iso9660 user,noauto   0 0
192.168.100.2:/raid5/nfs      /mnt/nfs      nfs         auto         0 0

```

Рисунок 37 - автосмонтирование

Задание 3 — NTP

Для начала на hq-rtr нужно установить chrony: `apt install chrony`

Заходим в файл `/etc/chrony/chrony.conf` и приводим его к виду рисунка 38.

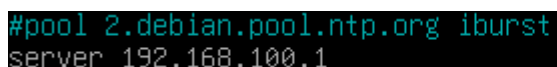


```
confdir /etc/chrony/conf.d
# Use Debian vendor zone.
#pool 2.debian.pool.ntp.org iburst
server 127.0.0.1 iburst prefer
local stratum 5
allow 0/0
```

Рисунок 38 — Конфигурация серверами

Перезапускаем chrony: `systemctl restart chronyd`

Теперь на каждую машину, кроме ISP устанавливаем chrony: `apt install chrony` и в файле конфигурации указать сервер hq-rtr(рисунок 39)



```
#pool 2.debian.pool.ntp.org iburst
server 192.168.100.1
```

Рисунок 39 — указание сервера

перезапускаем chrony: `systemctl restart chrony`

Проверяем работу командой `chronyc sources`

Задание 4 — файл инвентаря

Для начала нужно установить ansible: `apt install ansible`

Теперь нужно создать ключи rsa(Рисунок 38)

```
sshuser@br-srv:/root$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/sshuser/.ssh/id_rsa):
Created directory '/home/sshuser/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/sshuser/.ssh/id_rsa
Your public key has been saved in /home/sshuser/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:4Br1z2qa5DxBIPNbaFnazuBD7PkaT1fRMyoXQP9hIyU sshuser@br-srv.au-team.irpo
The key's randomart image is:
+----[RSA 3072]-----+
|      .o E .          |
| o . .o o .          |
| = Bo = = +          |
| Xo+o * = o          |
| =.O. S +            |
| *O+ =               |
| ooo.. o             |
| Boo..               |
| ..*+.              |
|-----[SHA256]-----+
```

Рисунок 38 — Генерация ключа

Теперь это ключ нужно переместить на остальные машины командой:

ssh-copy-id -p 2024 sshuser@192.168.100.2

```
sshuser@br-srv:/root$ ssh-copy-id -p 2024 sshuser@192.168.100.2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/sshuser/.ssh/id_rsa.pub"
The authenticity of host '[192.168.100.2]:2024 ([192.168.100.2]:2024)' can't be established.
ED25519 key fingerprint is SHA256:tfe9mdhYOG4Rprw5au0pTulg8lVl2I6tQoquqYHhNpg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are alr
eady installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to inst
all the new keys
Authorized access only
sshuser@192.168.100.2's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh -p '2024' 'sshuser@192.168.100.2'"
and check to make sure that only the key(s) you wanted were added.
```

Рисунок 39 — передача ключа на HQ-SRV

Тоже самое нужно сделать для остальных машин

Теперь нужно залогиниться обратно под root и создать директорию /etc/ansible
и зайти в файл /etc/ansible/hosts для конфигурации файла инвентаря(Рисунок 40)


```
GNU nano 7.2 /etc/ansible/hosts *
[hq]
192.168.100.2 ansible_port=2024 ansible_user=sshuser
192.168.200.4 ansible_user=user
172.16.4.2 ansible_user=net_admin

[br]
172.16.5.2 ansible_user=net_admin
```

Рисунок 40 — файл инвентаря

Для проверки файла инвентаря используется команда `ansible all -m ping`

```
root@br-srv:~# ansible all -m ping
172.16.4.2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
192.168.100.2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
172.16.5.2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
192.168.200.4 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
```

Рисунок 41 - проверка

Задание 5 — mediawiki

Для установки DOCKER нужно запустить скрипт

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

и установить docker-compos: apt install docker-compose

Для упрощённого создания файла `wiki.yml` нужно с HQ-CLI по ssh подключиться к BR-SRV: **ssh -p 2024 sshuser@192.168.0.2**

После подключения по ssh на hq-cli открываем браузер и ищем mediawiki docker-compose. Нужно перейти по ссылке показанной на рисунке 42.

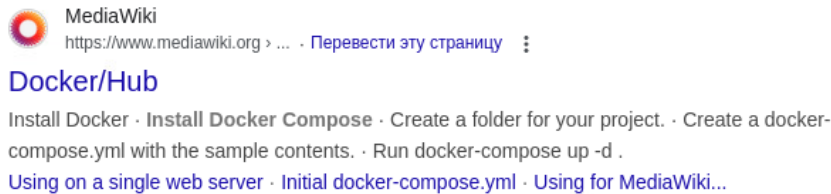


Рисунок 42 — Нужная ссылка

Ищем пункт «Adding a database server» и копируем весь конфиг(рисунок 43)

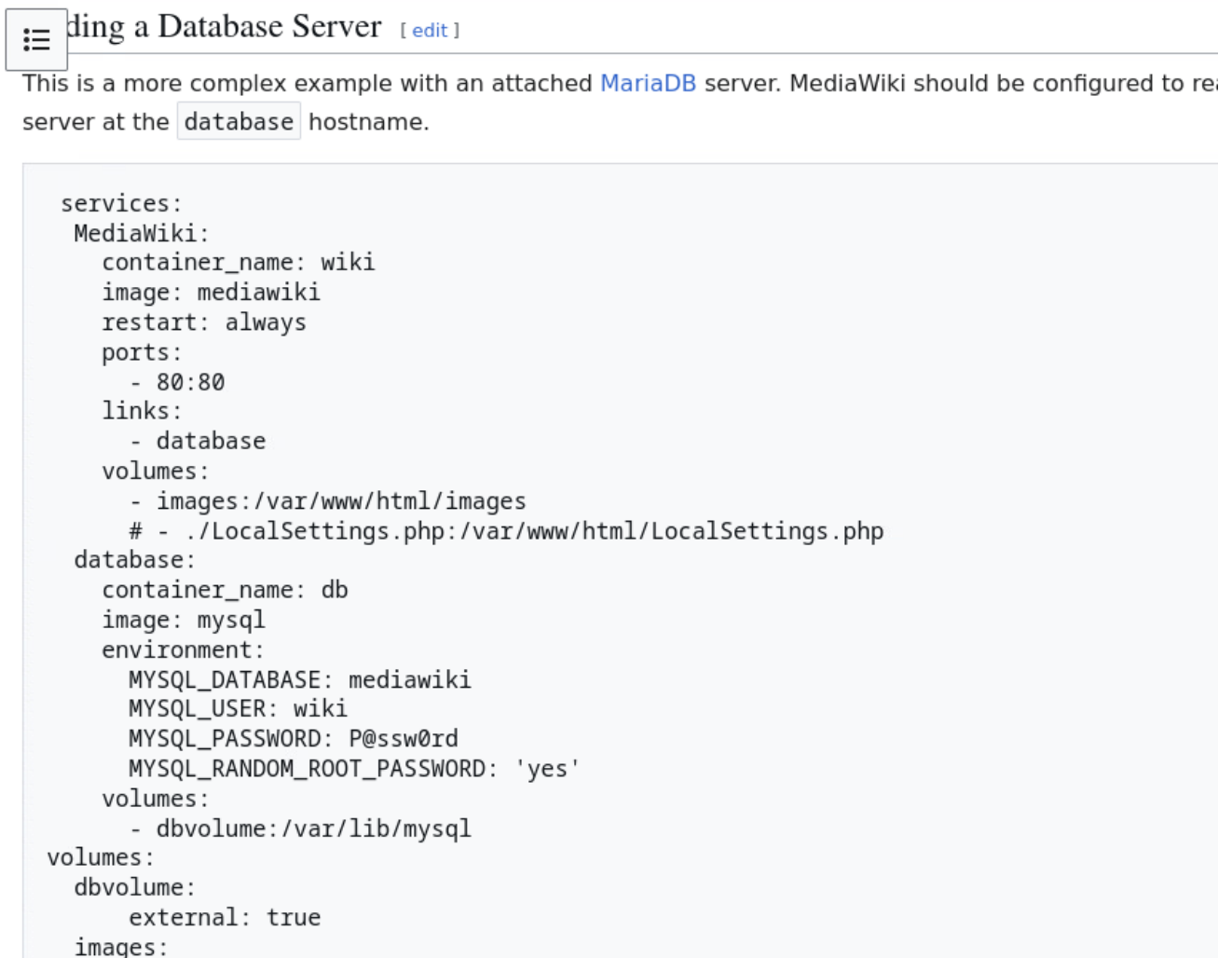


Рисунок 43 — конфиг mediawiki

теперь переходим в директорию /home/sshuser и создаём в ней файл wiki.yml. Заходим в этой файл. ВСЕ СТРОКИ, ОТМЕЧЕННЫЕ СТРЕЛКАМИ

НЕОБХОДИМО ЗАПОЛНИТЬ КАК ПОКАЗАНО НА РИСУНКЕ 44!!!!!!!!!!!!!!!!!!!!

```
MediaWiki:
  container_name: wiki
  image: mediawiki
  restart: always
  ports:
    - 8080:80
  links:
    - database
  volumes:
    - images:/var/www/html/images
    # - ./LocalSettings.php:/var/www/html/LocalSettings.php
database:
  container_name: mariadb
  image: mariadb
  environment:
    MYSQL_DATABASE: mediawiki
    MYSQL_USER: wiki
    MYSQL_PASSWORD: WikiP@ssw0rd
    MYSQL_RANDOM_ROOT_PASSWORD: 'yes'
  volumes:
    - dbvolume:/var/lib/mariadb
volumes:
  dbvolume:
    external: true
  images:
```

Рисунок 44 — изменённый конфиг

Теперь создаём volume для докера: `docker volume create dbvolume`

```
root@br-srv:/home/sshuser# docker-compose -f wiki.yml up -d
Creating volume "sshuser_images" with default driver
Pulling database (mariadb)...
latest: Pulling from library/mariadb
5a7813e071bf: Downloading [=====] 17.84MB/29.7
5MBcd990c29c: Download complete
5db80086e4da: Download complete
901fe9394c00: Download complete
43eb19e1b102: Download complete
597f7afe50fe: Waiting
e1dede558384: Waiting
5c3a22df929b: Waiting
```

После этого выполняем команду `docker-compose -f wiki.yml up -d` и запускаем стек контейнеров(Рисунок 45)

```
root@br-srv:/home/sshuser# docker-compose -f wiki.yml up -d
Creating volume "sshuser_images" with default driver
Pulling database (mariadb)...
latest: Pulling from library/mariadb
5a7813e071bf: Downloading [=====>] 17.84MB/29.7
5MBcd990c29c: Download complete
5db80086e4da: Download complete
901fe9394c00: Download complete
43eb19e1b102: Download complete
597f7afe50fe: Waiting
e1dede558384: Waiting
5c3a22df929b: Waiting
```

Рисунок 45 — Запуск стека

Ждём пока запуститься стек. После запуска открываем браузер и переходим по ip: 192.168.0.2:8080(Адрес br-srv с указанием порта для mediawiki)



MediaWiki 1.43.0

LocalSettings.php not found.

Please [set up the wiki](#) first.

Рисунок 46 — приветственное окно

Нажимаем СИНЮЮ ССЫЛКУ и выбираем язык. Соглашаемся с

Авторскими правами и условиями.

На рисунке 47 выбираем базу данных. ТАКЖЕ СЛЕДУЕМ СТРЕЛОЧКАМ.
ПАРОЛЬ: WikiP@ssw0rd

Тип базы данных:

☒ MariaDB, MySQL или совместимая

☐ SQLite

Настройки MariaDB/MySQL

Хост базы данных:

[справка](#)

mariadb

☐ Подключиться через SSL

Идентификация этой вики

Имя базы данных (без дефисов):

[справка](#)

mediawiki

Префикс таблиц базы данных (без дефисов):

[справка](#)

Учётная запись для установки

Имя пользователя базы данных:

[справка](#)

wiki

Пароль базы данных:


[справка](#)

••••••••••


[← Назад](#) [Далее →](#)

Рисунок 47 — Выбор базы данных

На рисунке 48 идёт задача названия страницы, создание пользователя для авторизации. ТАКЖЕ СЛЕДУЕМ СТРЕЛОЧКАМ. НА РИСУНКЕ 49 ИДЁТ ПРОДОЛЖЕНИЕ РИСУНКА 48. Там нужно просто поставить галочку.

Название вики:
 справка

LUBOY TEXT


Пространство имён проекта:
 справка

☒ То же, что имя вики: LUBOY_TEXT

☐ Проект

☐ Другое (укажите)

Учётная запись администратора

Ваше имя участника:
 справка


wiki

Пароль:


●●●●●●●●●●

Пароль ещё раз:

●●●●●●●●●●

Адрес электронной почты:
 справка

demo@demo.demo

☐ Подписаться на [рассылку новостей о появлении новых версий MediaWiki.](#)
 справка

☒ Поделиться сведениями об этой установке с разработчи- [Политика конфиденци-
альности.](#)


 справка

Рисунок 48 — создание базовой страницы и пользователям

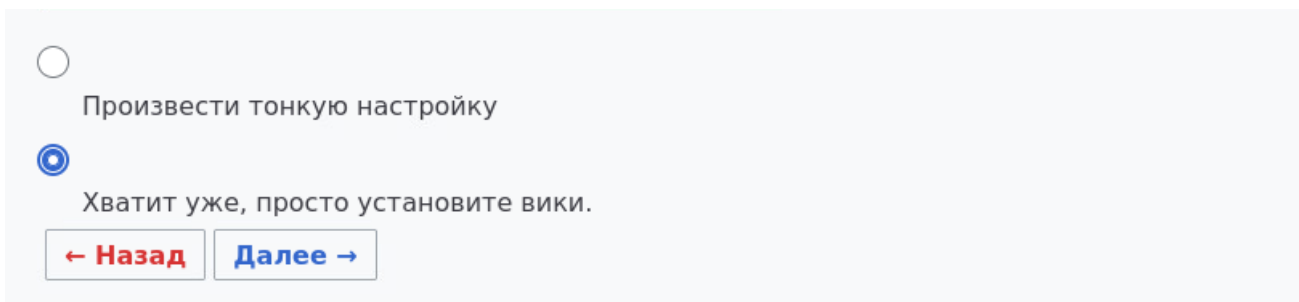


Рисунок 49 — ТА САМАЯ ГАЛОЧКА

На рисунке 50 показан файл.

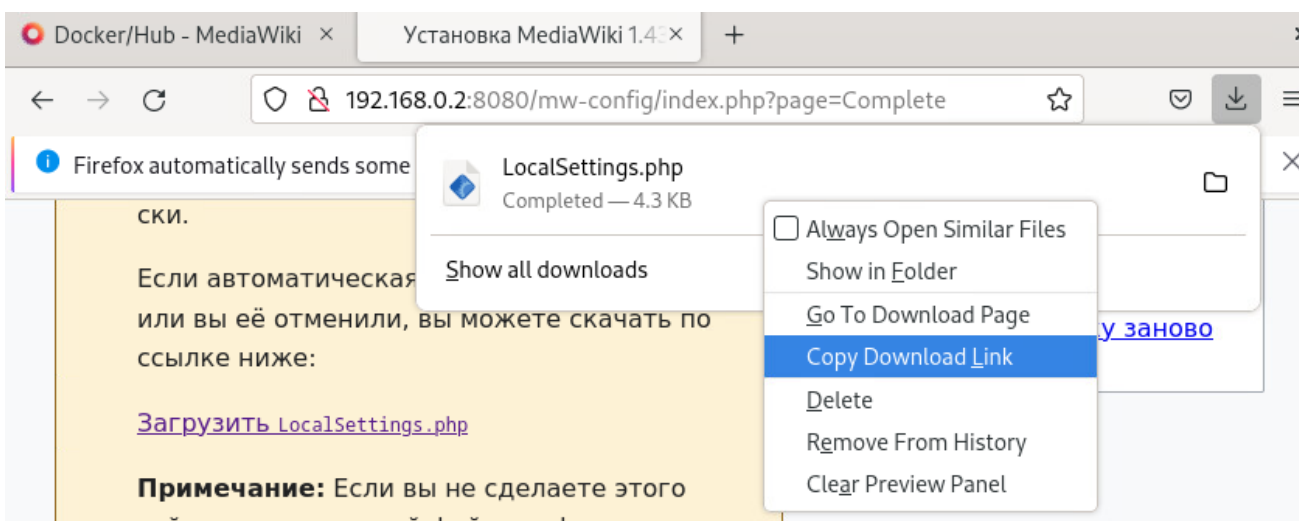


Рисунок 50 — Файл для установки

Перекидываем файл на br-srv:

```
scp -P 2024 /home/locadm/Downloads/LocalSettings.php
sshuser@192.168.0.2:/home/sshuser
```

Переходим в файл `wiki.yml` и раскомментируем единственную закомментированную строчку(рисунок 51). УБЕРИТЕ ЛИШНИЙ ПРОБЕЛ: СТРОКИ `images` и раскоментированная строка должны быть на одно уровне

```
- images: /var/www/html/images
# - ./LocalSettings.php: /var/www/html/LocalSettings.php
```

Рисунок 51 — Строка которую нужно расментировать

перезапускаем сервисы:

```
docker-compose -f wiki.yml stop
docker-compose -f wiki.yml up -d
```

Заходим в браузер на HQ-CLI и вводим 192.168.0.2:8080. Вы должны увидеть
лицевую страницу mediawiki:

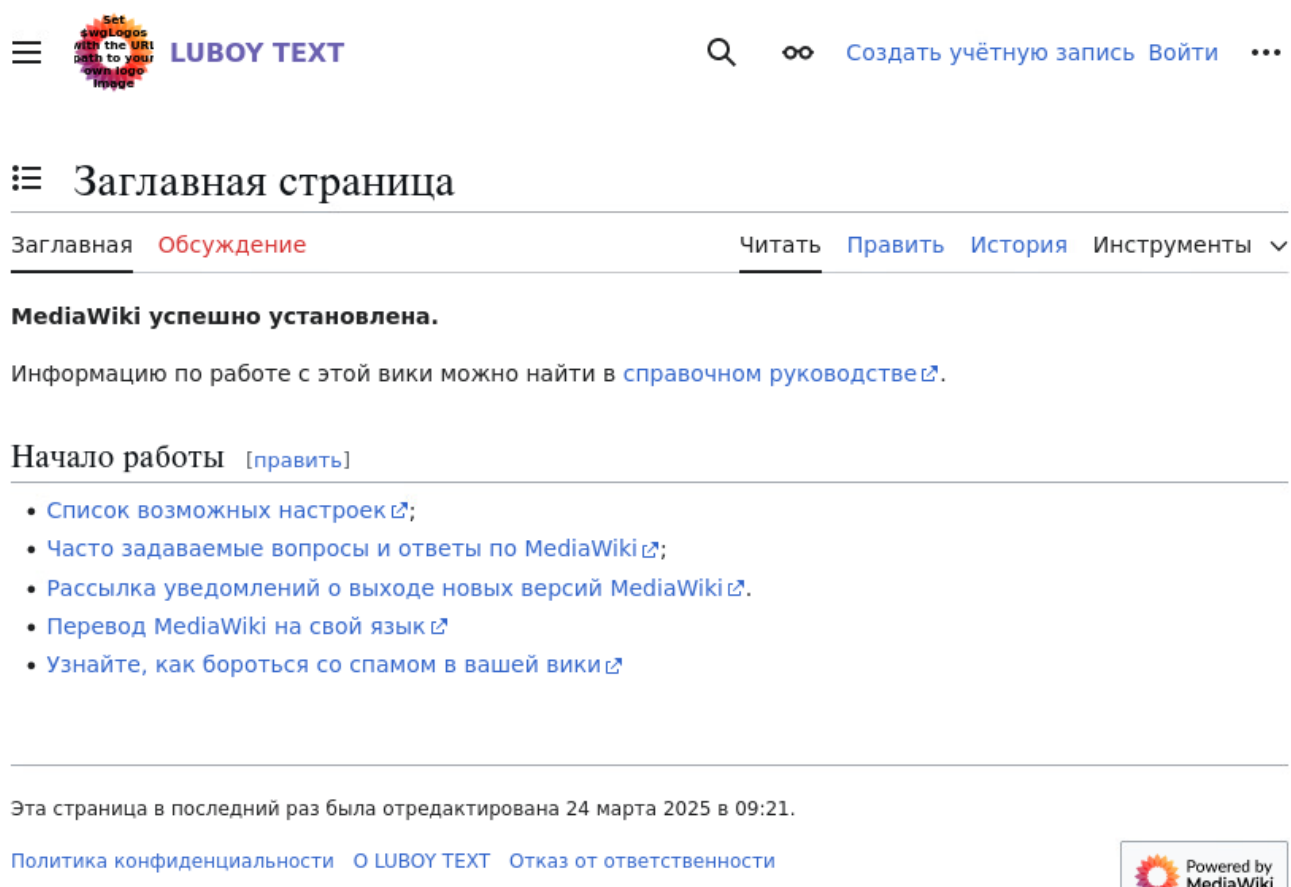


Рисунок 52 — Готовый mediawiki

Задание 6 — Проброс портов

На br-rtr заходи в файл /etc/nftables.conf и приводим его к виду, показанному

на рисунке 53.

```
flush ruleset

table inet filter {
    chain input {
        type filter hook input priority filter;
    }
    chain forward {
        type filter hook forward priority filter;
    }
    chain output {
        type filter hook output priority filter;
    }
}

table inet nat {
    chain PREROUTING {
        type nat hook prerouting priority filter;
        ip daddr 172.16.5.2 tcp dport 80 dnat ip to 192.168.0.2:8080
        ip daddr 172.16.5.2 tcp dport 2024 dnat ip to 192.168.0.2:2024
    }
}
```

Рисунок 53 — проброс порта для mediawiki и ssh на роутере br-rtr

Перезапускаем службу nftables

Пробуем зайти на mediawiki по адресу 172.16.5.2

Также пытаемся подключиться по ssh: **ssh -p 2024 sshuser@172.16.5.2**

```
GNU nano 7.2 /etc/nftables.conf *
#!/usr/sbin/nft -f

flush ruleset

table inet filter {
    chain input {
        type filter hook input priority filter;
    }
    chain forward {
        type filter hook forward priority filter;
    }
    chain output {
        type filter hook output priority filter;
    }
}

table inet nat {
    chain PREROUTING {
        type nat hook prerouting priority filter;
        ip daddr 172.16.4.2 tcp dport 2024 dnat ip to 192.168.100.2:2024
    }
}
```

Рисунок 54 — Проброс порта для ssh на hq-rtr

Для удобства также подключаемся с hq-cli к hq-srv по ssh.

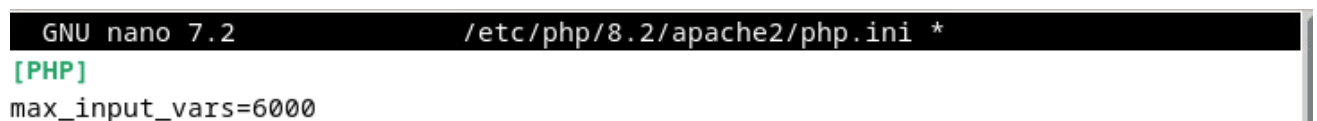
Ssh -p 2024 sshuser@192.168.100.2

Теперь устанавливаем apache2: apt install apache2

Также устанавливаем php и доп. расширения

apt install php php-mysqldb php-pdo php-gd php-mbstring php-zip php-intl php-soap
php-curl

Далее заходи в файл /etc/php/8.2/apache2/php.ini и добавляем там строку
max_input_vars=6000



```
GNU nano 7.2 /etc/php/8.2/apache2/php.ini *
[PHP]
max_input_vars=6000
```

Рисунок 55 — php.ini

перезапускаем apache2: systemctl restart apache2

Устанавливаем mariadb: apt install mariadb-server

Настраиваем mariadb: mysql_secure_installation, прокликаем enter до момента ввода пароля, вводим пароль [P@ssw0rd](#) и прокликаем enter до конца.

Заходим в базу данных: mysql -u root -p и вводим пароль P@ssw0rd

Создадим базу данных и пользователя, а также выдадим ему права для этой базы данных(Рисунок 56)

```
CREATE DATABASE moodledb DEFAULT CHARACTER SET utf8;
```

```
CREATE USER moodle@localhost IDENTIFIED BY 'P@ssw0rd';
```

```
GRANT ALL ON moodledb.* TO 'moodle'@'localhost';
```

flush privileges;

```
MariaDB [(none)]> create database moodledb default character set utf8;  
Query OK, 1 row affected (0.011 sec)  
  
MariaDB [(none)]> create user moodle@localhost identified by 'P@ssw0rd'  
-> ^C  
MariaDB [(none)]> create user moodle@localhost identified by 'P@ssw0rd';  
Query OK, 0 rows affected (0.009 sec)  
  
MariaDB [(none)]> grant all on moodle.* to 'moodle'@'localhost';  
Query OK, 0 rows affected (0.003 sec)
```

Рисунок 56 — создание базы данных и пользователя

Перезапускаем mariadb: `systemctl restart mariadb`

Теперь необходимо перейти на официальный сайт moodle.org и перейти во вкладку downloads(Рисунок 57)

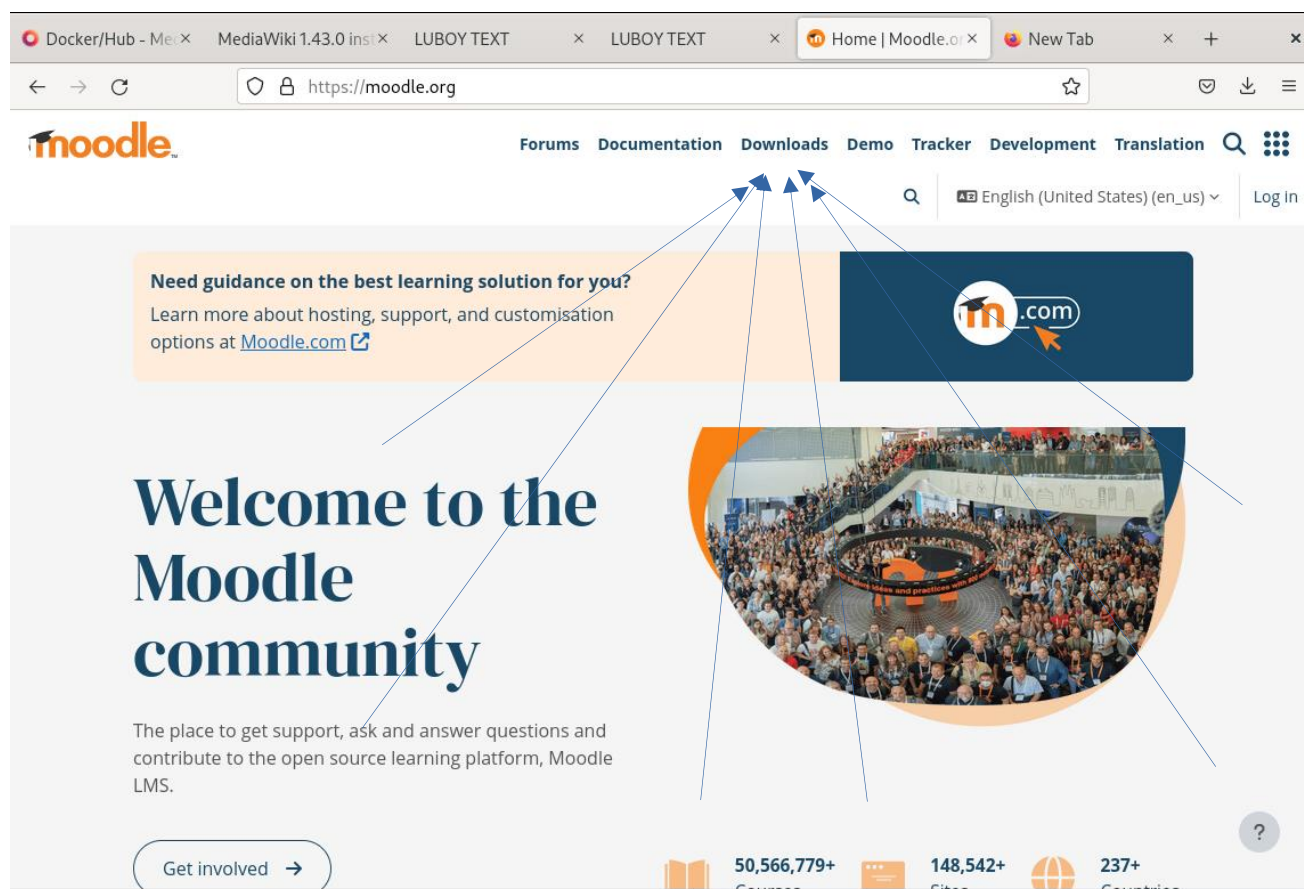


Рисунок 57 — сайт moodle

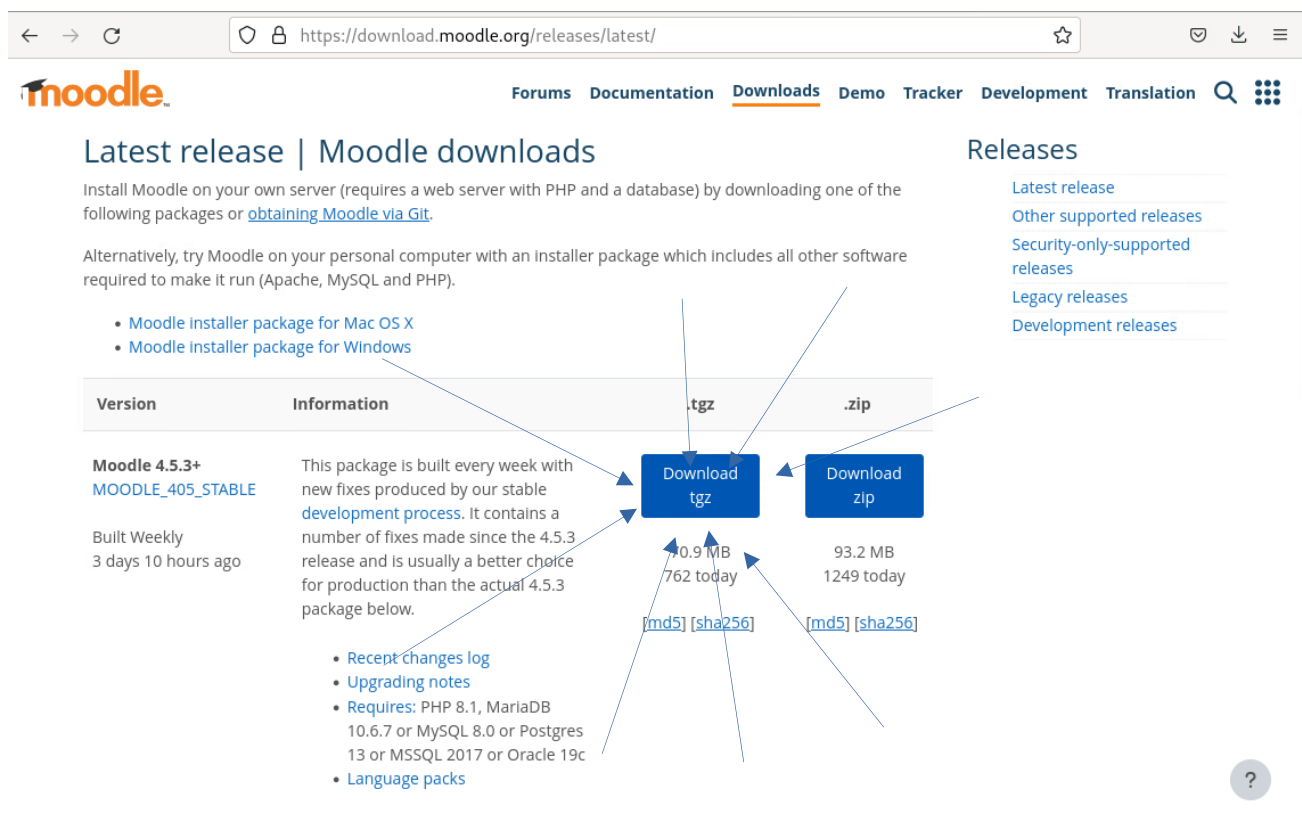


Рисунок 58 — Ссылка для скачивания

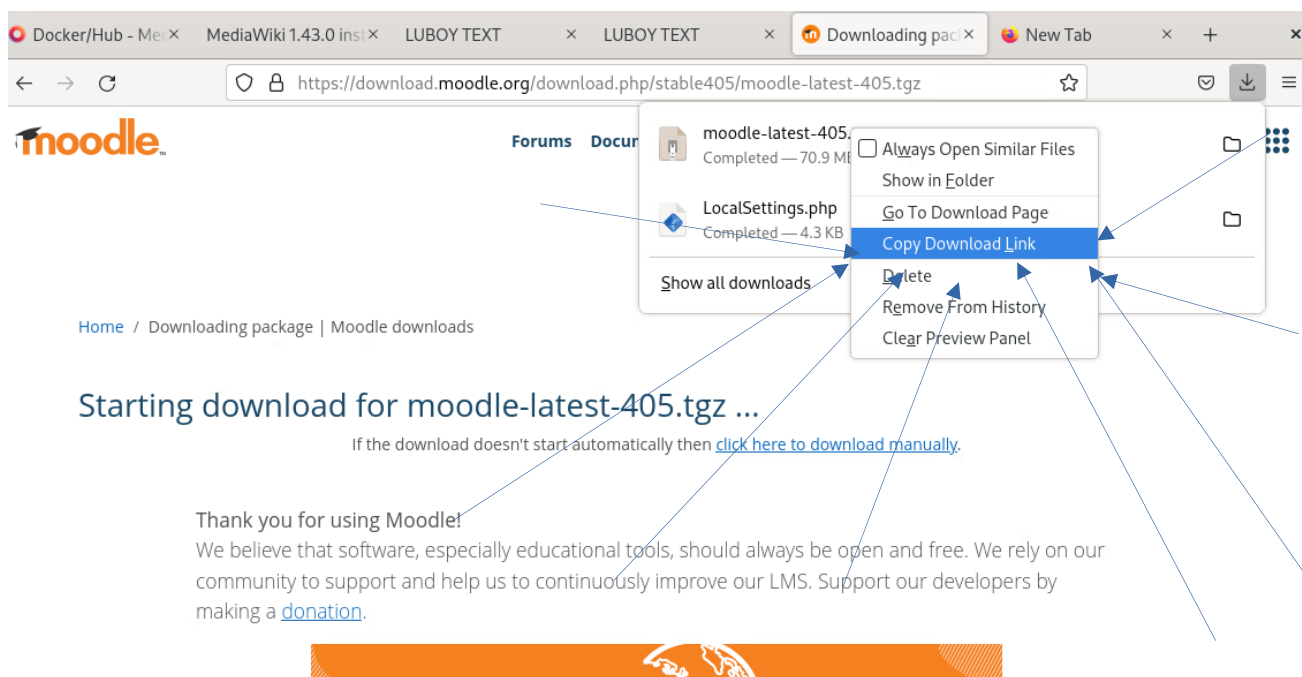


Рисунок 59 — ссылка на файл

wget https://packaging.moodle.org/stable405/moodle-latest-405.tgz -P /tmp

переходим в директорию /tmp: cd /tmp и распаковываем архив
tar -xzf /tmp/moodle-latest-405.tgz

Переместим всё содержимое в директорию `/var/www/html`
`mv -f /tmp/moodle/{.,}* /var/www/html/`

Установка прав на `/var/www/html`

`chmod -R 0755 /var/www/html/`

`chown -R www-data:www-data /var/www/html/`

Теперь нужно создать каталог `/var/moodledata` для модл:

`mkdir /var/moodledata`

И выдать права

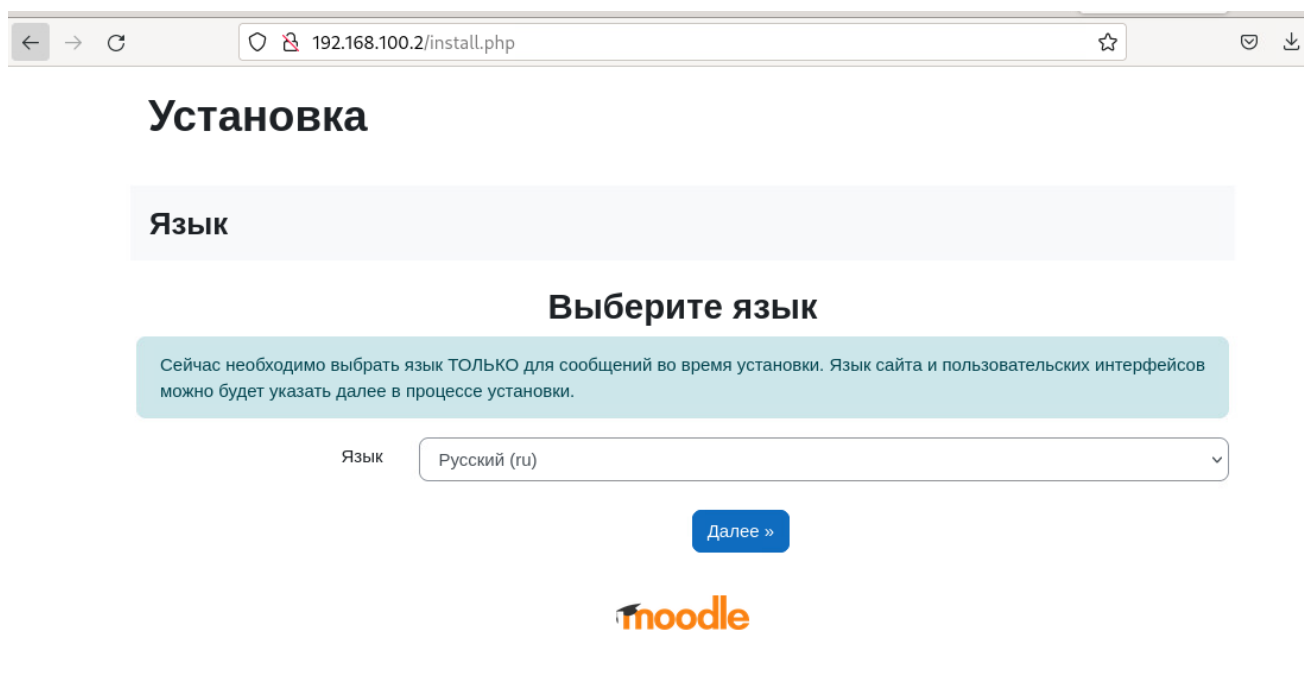
`chmod -R 0755 /var/moodledata`

`chown -R www-data:www-data /var/moodledata`

Удаляем файл `index.html`: `rm /var/www/html/index.html`

Перезапускаем `apache2`: `systemctl restart apache2`

В поисковой строке браузера вбиваем `ip-адрес hq-srv(192.168.100.2)` и выбираем язык модл(рисунок 60)



The screenshot shows a web browser window with the address bar displaying `192.168.100.2/install.php`. The page title is "Установка" (Installation). Below the title, there is a section labeled "Язык" (Language). The main heading is "Выберите язык" (Select language). A light blue informational box states: "Сейчас необходимо выбрать язык ТОЛЬКО для сообщений во время установки. Язык сайта и пользовательских интерфейсов можно будет указать далее в процессе установки." (Now you need to select a language ONLY for messages during installation. The language of the site and user interfaces can be specified later in the installation process). Below this, there is a label "Язык" and a dropdown menu currently showing "Русский (ru)". A blue button labeled "Далее »" (Next ») is positioned below the dropdown. At the bottom of the page is the Moodle logo.

Рисунок 60 — выбор языка

(обычно 'www-data', 'nobody' или 'apache').

Этот каталог не должен быть доступен напрямую через Интернет.

Программа установки попробует создать этот каталог, если он не существует.

Веб-адрес

Каталог Moodle

Каталог данных

« Назад **Далее »**

Рисунок 61 — выбор пути

Название базы данных

Выберите драйвер базы данных

Moodle поддерживает несколько типов серверов баз данных. Свяжитесь с администратором сервера, если не знаете, какой именно тип выбрать.

Тип

« Назад **Далее »**




Рисунок 62 — выбор базы данных

Сервер баз данных

Название базы данных

Пользователь базы данных

Пароль

Префикс имен таблиц

Порт базы данных

Подключение через Unix-сокеты

« Назад **Далее »**

Рисунок 63 — база данных

Возможно возникнет ошибка. Поэтому заходим на сервер и скачиваем php-xml. `Apt install php-xml` и перезапускаем `apache2`.

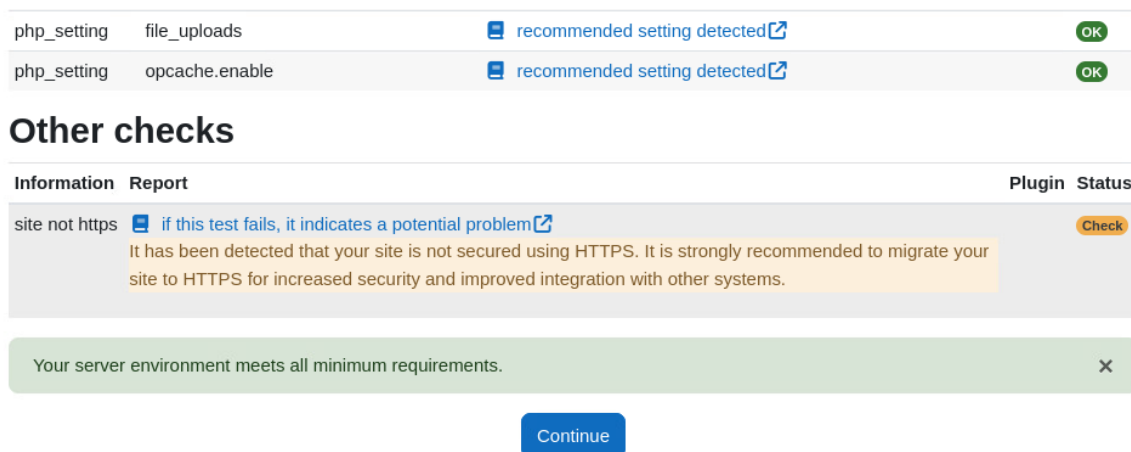


Рисунок 64 — Нажимаем далее

После этого начнётся установка. ПРОСТО ЖДЁМ

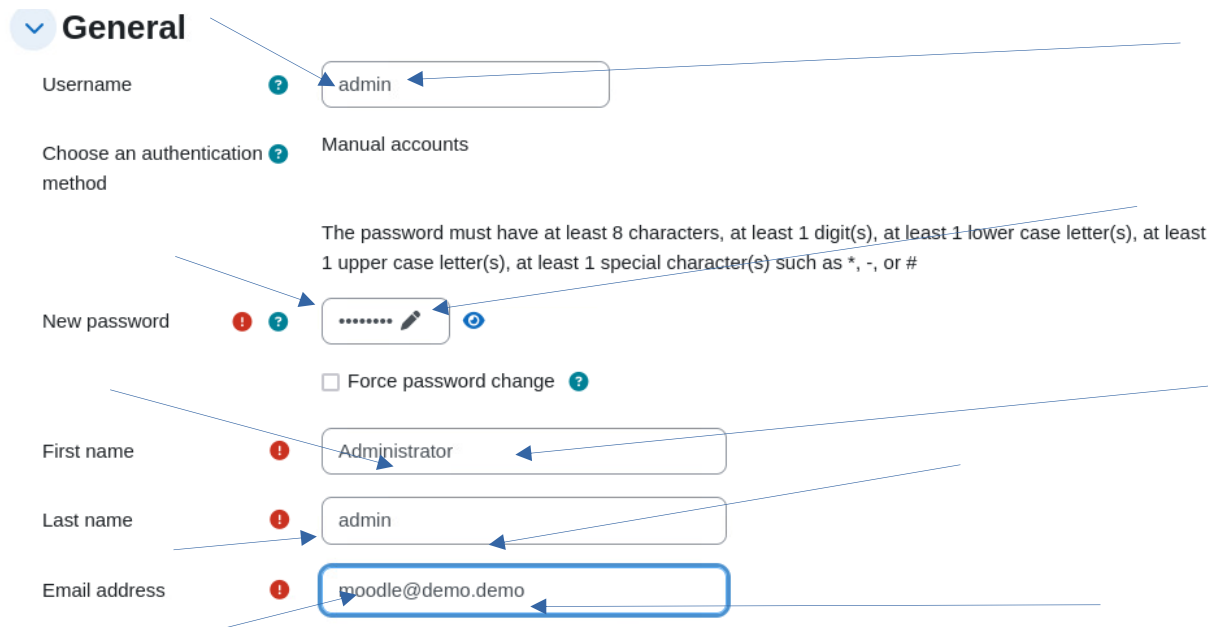


Рисунок 65 — Заполняем базовые данные

Installation

New settings - Site home settings

Full site name
fullname

Short name for site (eg single word)
shortname

Site home summary
summary

mesto200

200

demo 2025 sisa

Рисунок 66 — Вместо 200 указываете номер своего рабочего места

New settings - Support contact

Support email
supportemail

moodle@demo.demo

If SMTP is configured on this site and a support page is not set, this email address will receive messages submitted through the support form. If sending fails, the email address will be displayed to logged-in users.

Рисунок 67 — делается на той же странице, что и рисунок 66

После этого на главной странице выведется номер вашего места(рисунок 68)

200 Home Dashboard My courses Site administration

mesto200

Home Settings Participants Reports Question bank More

Available courses

Add a new course

Рисунок 68 — номер места

Задание 8 — nginx

Для начала нужно установить nginx на hq-rtr: `apt install nginx`

теперь заходи в файл `/etc/nginx/sites-enabled/hq-rtr.conf` и приводим его к виду рисунка 69.

```
server {
    listen 80;
    server_name moodle.au-team.irpo;

    location / {
        proxy_pass http://192.168.100.2:80;
    }
}

server {
    listen 80;
    server_name wiki.au-team.irpo;

    location / {
        proxy_pass http://192.168.0.2:8080;
    }
}
```

Рисунок 69 — конфигурация nginx

Теперь нужно проверить конфигурацию на наличие ошибок: `nginx -t`

Если ошибок не найдено, перезапускаем nginx.

Также заходим в `/etc/resolv.conf` (Рисунок 70). Так нужно проделать на всех машинах, кроме ISP

```
# Generated by NetworkManager
search au-team.irpo
domain au-team.irpo
#nameserver 1.1.1.1
nameserver 192.168.100.2
```

Рисунок 70 — resolv.conf

Теперь в браузере пытаемся подключиться к мудл и медиавики по доменному имени (Рисунок 71 и 72)



mesto200

Рисунок 71 — Доступность мудл

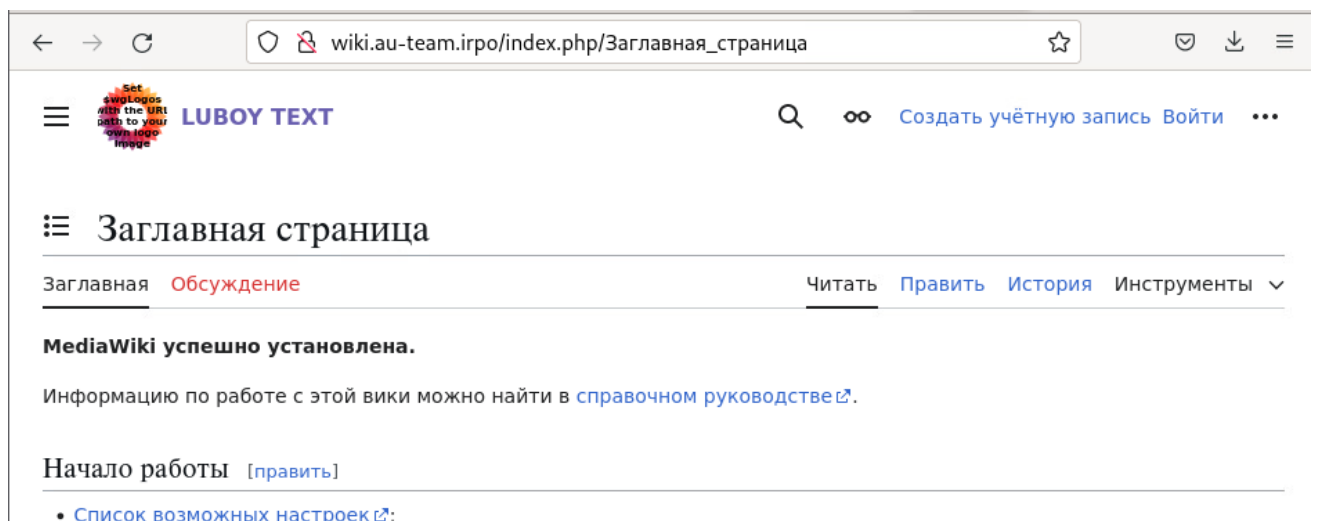


Рисунок 72 — Доступность Медиавики

