

## Сказка о настройке волшебного королевства сетей

В далеком цифровом королевстве жили-были разные устройства - гордые серверы, трудолюбивые маршрутизаторы и скромные рабочие станции. И вот однажды мудрый системный администратор решил объединить их всех в единую сеть, чтобы они могли общаться между собой и служить на благо королевства.

### Глава 1: Наречение имен

Прежде всего, администратор решил дать каждому устройству свое особенное имя, чтобы можно было легко отличать их друг от друга. Он подошел к каждому устройству и произнес волшебные слова:\

```
hostnamectl set-hostname «имя_машины»; exec bash
```

Так каждое устройство получило свое полное доменное имя, став полноправным гражданином сетевого королевства. Серверы гордо носили имена HQ-SRV и BR-SRV, маршрутизаторы - HQ-RTR и BR-RTR, а рабочие станции скромно назывались HQ-CLI и BR-DC.

### Глава 2: Раздача волшебных адресов

Теперь нужно было обеспечить каждому устройству свой уникальный адрес в королевстве. Для устройств с графическим интерфейсом это было просто - достаточно было щелкнуть правой кнопкой по значку сети, выбрать настройки IPv4 и аккуратно вписать нужные цифры, не забыв сохранить изменения.

А вот для устройств без графического интерфейса пришлось использовать волшебный инструмент nmtui:

Администратор тщательно распределил адреса между всеми жителями королевства, записав их в священный свиток:

```
isp
ens34 (bf)    ens35 (c9)
172.16.4.1/28 172.16.5.1/28
172.16.4.2/28 172.16.5.2/28
ens34 (ba)    ens34 (54)
hq-rtr       br-rtr
ens35 (c4)    ens35 (5e)
172.16.0.1/26 172.16.6.1/27
ens33 (d6)    ens34 (af)    ens33 (b2)    (Он винда)
172.16.0.2/26 172.16.0.3/28 172.16.6.2/27 172.16.6.3/27
hq-srv       hq-cli       br-srv       br-DC
```

### Глава 3: Создание верных слуг

Чтобы управлять королевством, администратору нужны были верные помощники. Он создал специального пользователя `sshuser`, который мог бы выполнять любые команды без лишних вопросов:

```
useradd -m -u 1010 sshuser
```

```
passwd sshuser
```

Затем он открыл священный свиток `sudoers` и добавил туда магическую строку, дающую `sshuser` неограниченные права:

```
nano /etc/sudoers
```

Добавил:

```
sshuser ALL=(ALL:ALL)NOPASSWD:ALL
```

Сохранил изменения священной комбинацией клавиш: `Ctrl+X`, `Y`, `Enter`. Теперь у него был верный слуга, готовый выполнять любые поручения.

### Глава 4: Защитные заклинания

Королевству нужна была защита от злых духов и хакеров. Администратор создал специальное предупреждение для всех, кто попытается войти без разрешения:

```
nano /etc/mybanner
```

Написал строгое предупреждение:

```
Authorized access only
```

Затем настроил защитные механизмы SSH, изменив конфигурационный файл:

```
nano /etc/openssh/sshd_config
```

Установил:

```
#port 22, раскоментируем и пишем port 2024
```

```
Banner /etc/mybanner
```

```
MaxAuthTries 2
```

ДОБАВИТЬ строчку - `AllowUsers sshuser`

После этого перезапустил службу SSH, чтобы изменения вступили в силу:

```
systemctl restart sshd.service
```

Теперь королевство было под надежной защитой.

### Глава 5: Автоматическая раздача адресов

Чтобы жителям королевства не приходилось вручную запоминать свои адреса, администратор настроил DHCP-сервер на HQ-RTR. Сначала он указал, какой интерфейс будет раздавать адреса:

```
nano /etc/sysconfig/dhcpd
```

Добавил строку:

```
DHCPARGS=ens35
```

Затем создал конфигурационный файл, взяв за основу пример:

```
cp /etc/dhcp/dhcpd.conf{.example,}
```

```
nano /etc/dhcp/dhcpd.conf
```

Прописал основные параметры:

Доменное имя королевства "au-team.irpo"

Адреса DNS-серверов

```
option domain-name-servers 172.16.0.2;
```

Время аренды адресов

```
default-lease-time 6000;
```

```
max-lease-time 72000;
```

Диапазон раздаваемых адресов

```
authoritative;
```

```
    subnet 172.16.0.0 netmask 255.255.255.192 {
```

```
        range 172.16.0.3 172.16.0.8;
```

```
        option routers 172.16.0.1;
```

```
    }
```

После этого включил и запустил службу DHCP:

```
systemctl enable --now dhcpd
```

Теперь все новые жители королевства автоматически получали свои адреса.

Сказка о настройке волшебного королевства сетей (Продолжение)

## Глава 6: Тайный тоннель между замками

Когда основные дороги королевства были проложены, администратор задумался о создании секретного прохода между главным замком HQ и удалённой крепостью BR. Но для этого сначала нужно было получить разрешение от Хранителя Врат — сервера ISP.

Администратор подошёл к ISP и произнёс священные слова:

```
nano /etc/net/sysctl.conf
```

Найдя строку `net.ipv4.ip_forward`, он изменил её значение на 1, словно поворачивая ключ в скрипучем замке:

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

Теперь пакеты могли свободно проходить через ISP. Вдохновлённый, администратор взял волшебный инструмент `nmtui` и начал настраивать GRE-тоннель между HQ-RTR и BR-

RTR. Это было подобно прокладыванию подземного хода — невидимого для посторонних глаз, но надёжно соединяющего два удалённых замка.

## Глава 7: Живые дороги OSPF

Обычные дороги королевства были статичны — если где-то случался обвал, посланники могли заблудиться. Администратор решил оживить дороги с помощью магии динамической маршрутизации OSPF.

На HQ-RTR он открыл древний свиток:

```
nano /etc/frr/daemons
```

И сменил строку `ospfd=no` на `ospfd=yes`, пробуждая древний дух маршрутизации. Затем произнёс заклинание активации:

```
systemctl enable --now frr
```

Войдя в священный интерфейс `vysh`, администратор начал настраивать маршруты:

```
conf t
```

```
router ospf
```

```
passive-interface default
```

```
network 192.168.0.0/24 area 0
```

```
network 172.16.0.0/26 area 0
```

```
exit
```

```
interface tun1
```

```
no ip ospf network broadcast
```

```
no ip ospf passive
```

```
exit
```

```
do write memory
```

```
exit
```

Не забыл он и про настройку TTL для тоннеля:

```
bash
```

```
nmcli connection edit tun1
```

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

```
save
```

```
quit
```

После перезапуска FRR дороги ожили и стали сами находить обходные пути в случае преград. То же самое он проделал и на BR-RTR, и с тех пор посланники между замками никогда не терялись.

## Глава 8: Великая книга имён

В королевстве было много жителей, и запомнить все имена становилось трудно. Администратор решил создать Великую Книгу Имян (DNS) на HQ-SRV.

Он начал с изменения основных настроек:

```
nano /etc/bind/options.conf
```

Затем создал зоны, скопировав священные образцы:

```
cd /etc/bind/zone
```

```
cp localdomain au.db
```

```
cp 127.in-addr.arpa 0.db
```

Изменив владельцев файлов, чтобы только избранные могли вносить изменения:

```
chown root:named {au,0}.db
```

После настройки зонных файлов он перезапустил службу:

```
systemctl restart bind
```

Теперь, произнеся заклинание:

```
host hq-rtr.au-team.irpo
```

можно было мгновенно узнать адрес любого жителя королевства. "Благослови тебя Омниссия!" — прошептал администратор, любуясь своей работой.

## Глава 9: Создание центрального управления

Пришло время объединить всех жителей под единым управлением. Администратор начал настройку Samba AD-DC на HQ-SRV, но сначала временно отключил все интерфейсы через nmtui, чтобы никто не помешал священному ритуалу.

Он очистил старые конфигурации:

```
rm -f /etc/samba/smb.conf
```

```
rm -rf /var/lib/samba
```

```
rm -rf /var/cache/samba
```

Создал новые каталоги и начал процесс провижининга:

```
mkdir -p /var/lib/samba/sysvol
```

```
samba-tool domain provision
```

После настройки включил службы:

```
systemctl enable --now samba
```

```
systemctl enable --now bind
```

Когда bind отказался запускаться, администратор не растерялся. Он заглянул в конфигурационный файл, внёс необходимые изменения и перезапустил службу:

```
nano /etc/bind/named.conf
```

```
systemctl restart bind
```

Проверив статус службы, он убедился, что всё работает как надо. Затем настроил аутентификацию Kerberos:

```
nano /etc/krb5.conf
```

```
samba-tool domain info 127.0.0.1
```

```
kinit administrator@au-team.irpo
```

## **Глава 10: Первые подданные королевства**

Пришло время создать первых пользователей. Администратор открыл волшебный инструмент `adms` и создал пять верных подданных:

```
user1.hq
```

```
user2.hq
```

```
user3.hq
```

```
user4.hq
```

```
user5.hq
```

Каждый получил свой уникальный пароль и права в королевстве.































ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ  
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ Г. МОСКВЫ  
«КОЛЛЕДЖ ПРЕДПРИНИМАТЕЛЬСТВА №11»  
ЦЕНТР ИНФОРМАЦИОННО–КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

Отчёт по выполнению задания демонстрационного экзамена  
специальности 09.02.06 «Сетевое и системное администрирование»  
КОД 09.02.06-3-2025

Выполнил студент гр. С-41  
Печенкин Тимофей Владимирович

Москва 2025

## **Задания:**

- 1. Расчет IP-адресации**
- 2. Выбор и создание туннеля**
- 3. Выбор технологии динамической маршрутизации и её настройка**
- 4. Настройка динамической адресации**
- 5. Создание и настройка файлового хранилища**
- 6. Настройка moodle**
- 7. Установка браузера**
- 8. Настройка туннеля до уровня обеспечивающего шифрование трафика**
- 9. Выбор системы мониторинга и настройка этой системы**

### **1. Расчет IP-адресации**

В таблице показано, какие адреса закреплены за конкретными устройствами.

Имя устройства	IP-адрес	Шлюз по умолчанию
ISP	172.16.4.1/28 172.16.5.1/28	
HQ-RTR	172.16.4.2/28 172.16.0.1/26	172.16.4.1
BR-RTR	172.16.5.2/28 172.16.6.1/27	172.16.5.1
HQ-SRV	172.16.0.2/26	172.16.0.1

HQ-CLI	172.16.0.3/28	
BR-SRV	172.16.6.2/27	
BR-DC	172.16.6.3/27	

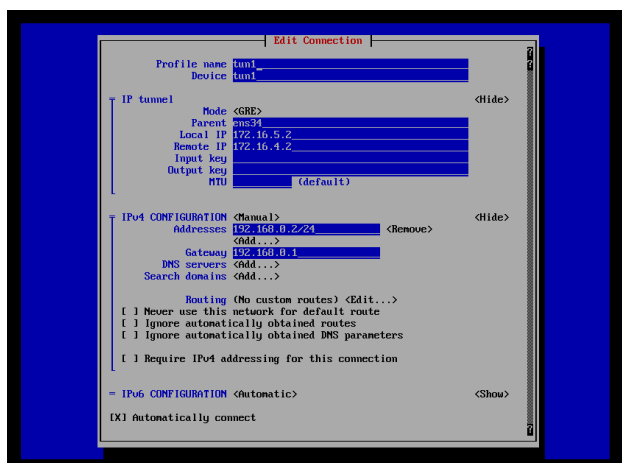
## 2. Выбор и создание туннеля

Для организации соединения между BR-RTR и HQ-RTR предпочтение отдали протоколу GRE вместо IP-in-IP благодаря его расширенному функционалу. Основные причины выбора GRE включают:

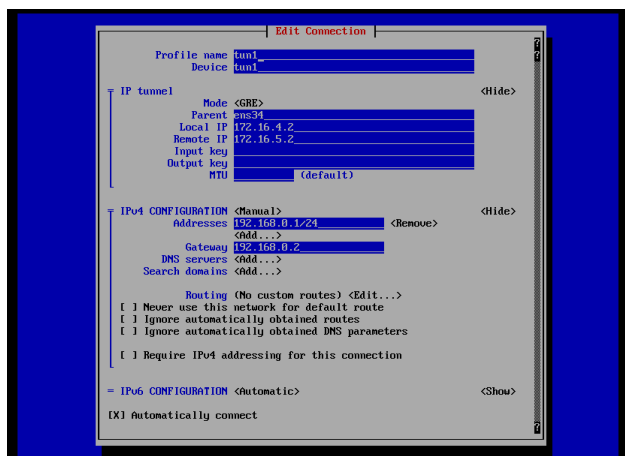
1. Поддержка широковещательного трафика — GRE позволяет инкапсулировать широковещательные и multicast-пакеты, что критично для работы некоторых сетевых протоколов.
2. Универсальная совместимость — оборудование и ОС, не поддерживающие обработку IP-in-IP, как правило, корректно взаимодействуют с GRE-туннелями.
3. Механизмы защиты — GRE предоставляет опцию аутентификации заголовков туннеля, снижая риски несанкционированного доступа.

Эти особенности делают GRE более гибким и безопасным решением для построения защищенных туннелей в гетерогенных сетевых средах.

### Настройка GRE на BR-RTR



## Настройка GRE на HQ-RTR



### 3. Выбор технологии динамической маршрутизации и её настройка

Протокол OSPF выбран в качестве основного решения, исходя из ключевых требований:

Высокая скорость конвергенции — быстрое формирование маршрутных таблиц при старте или изменении топологии.

Нативная совместимость с Alt Linux — полная поддержка на уровне ОС, включая инструменты управления и мониторинга.

Адаптивность к изменениям — автоматическая корректировка маршрутов при расширении сети или обновлении оборудования.

## Настройка протокола OSPF на BR-RTR



## Настройка протокола OSPF на HQ-RTR

```
GNU nano 7.2 /etc/frr/frr.conf
frr version 8.5.1
frr defaults traditional
hostname HQ-R
log file /var/log/frr/frr.log
no ip forwarding
no ip6 forwarding
!
interface tun1
 no ip ospf passive
exit
!
router ospf
 passive-interface default
 network 172.16.0.0/24 area 0
 network 172.16.0.0/26 area 0
 network 172.16.0.0/28 area 0
 network 192.168.0.0/24 area 0
exit
!
```

## 4. Настройка динамической адресации

### Настройка протокола DHCP на HQ-RTR

```
GNU nano 7.2 /etc/dhcp/dhcpd.conf
# Sample configuration file for ISC dhcpd
#

# option definitions common to all supported networks...
option domain-name "au-team.ipro";
option domain-name-servers 172.16.0.2 ;

default-lease-time 6000;
max-lease-time 72000;
authoritative;
# Use this to enable / disable dynamic dns updates globally.
#ddns-update-style none;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
#authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;

# No service will be given on this subnet, but declaring it helps the
# DHCP server to understand the network topology.

subnet 10.152.187.0 netmask 255.255.255.0 {
}

# This is a very basic subnet declaration.

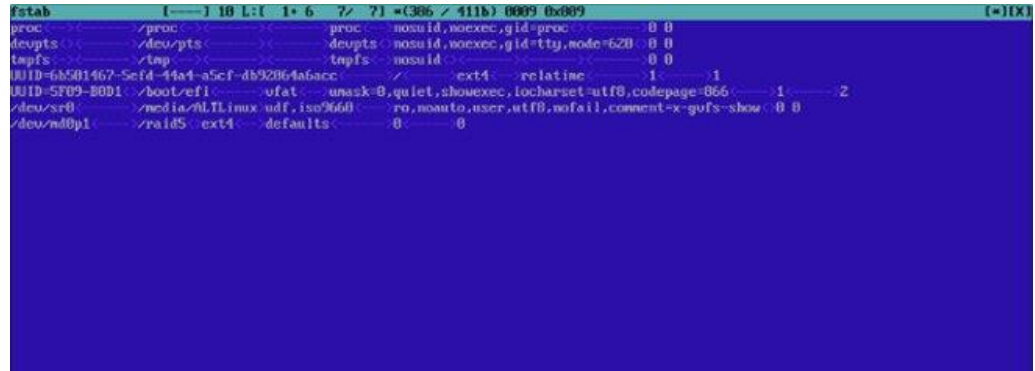
subnet 172.16.0.0 netmask 255.255.255.224 {
 range 172.16.0.3 172.16.0.8;
 option routers 172.16.0.1;
}
```

## 5. Создание и настройка файлового хранилища

На сервере HQ-SRV реализован массив RAID 5 уровня, объединяющий три накопителя емкостью по 1 ГБ каждый. Выбор данной конфигурации обусловлен оптимальным сочетанием производительности и



отказоустойчивости — технология RAID 5 обеспечивает защиту данных за счет распределенной чётности, сохраняя высокую скорость операций чтения. Для упрощения работы с массивом выполнена настройка автоматического подключения в системную директорию /raid5, что гарантирует бесперебойный доступ к хранилищу при перезагрузках.



```
fstab
proc /proc proc nosuid,noexec,gid=proc 0 0
devpts /dev/pts devpts nosuid,noexec,gid=ttty,mode=620 0 0
tmpfs /tmp tmpfs nosuid 0 0
UUID=6A201467-5cf4-44a4-a5cf-4b92064a6acc / ext4 relative 1 1
UUID=5F09-B0D1 /boot/efi vfat unask=0,quiet,showexec,iocharset=utf8,codepage=866 1 2
/dev/xr0 /media/ALTlinux udf,iso%60 ro,noauto,user=utf8,nofail,comment=x-gvfs-show 0 0
/dev/md0p1 /raid5 ext4 defaults 0 0
```

## 6. Настройка moodle

На сервере HQ-SRV развернута платформа Moodle для управления образовательным процессом, интегрированная с СУБД MariaDB. Конфигурация включает:

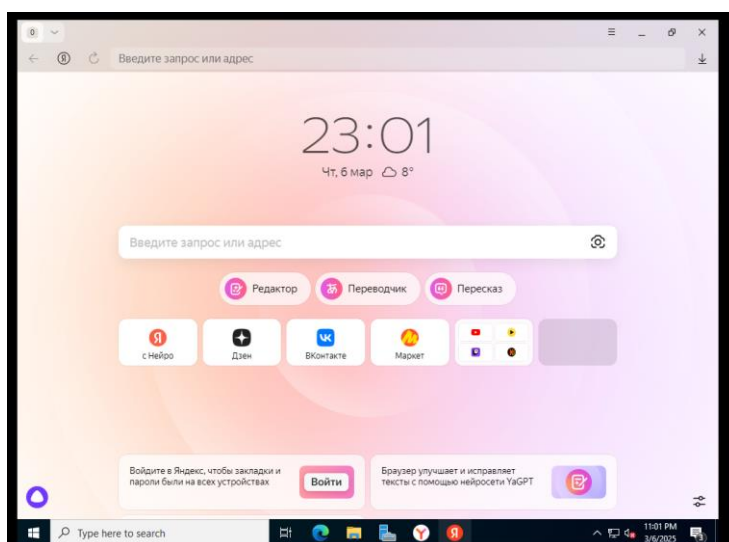
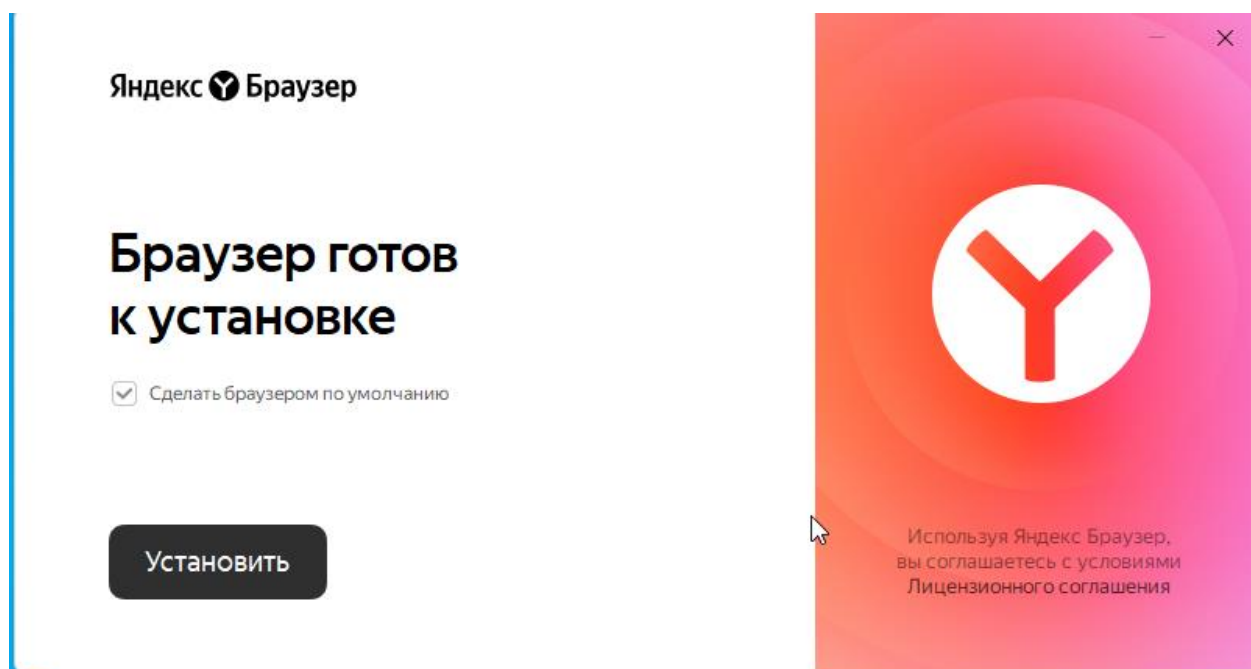
- База данных: moodledb
- Учетные записи: Пользователи: moodle (для работы системы), admin (административный доступ)

Аутентификация: пароль P@ssw0rd

Такая связка обеспечивает стабильную работу Moodle с поддержкой транзакций, резервного копирования и управления правами доступа через MariaDB.

## 7. Установка браузера

Для установки браузеры был выбран Yandex браузер так как он соответствует требованиям задания



## 8. Настройка туннеля до уровня обеспечивающего шифрование трафика

Для повышения защищенности соединения между серверами HQ-SRV и BR-SRV базовый IP-туннель был усовершенствован. Внедрение протокола IPsec обеспечило сквозное шифрование трафика с использованием алгоритма AES-256, обеспечивающего криптостойкость за счет 256-битных ключей. Аутентификация реализована через Pre-Shared Key (PSK) — метод, упрощающий развертывание, но менее надежный по сравнению с сертификатной аутентификацией.

Ключевые изменения:

Переход от незащищенного туннеля к зашифрованному каналу передачи данных;

Оптимальный баланс между безопасностью (AES-256) и простотой конфигурации (PSK);

Совместимость с существующей инфраструктурой без необходимости внедрения PKI.

Данный подход минимизирует риски перехвата данных, сохраняя при этом умеренные требования к ресурсам настройки.

## **9. Выбор системы мониторинга и настройка этой системы**

В качестве решения для мониторинга выбран Zabbix, что обусловлено следующими факторами:

1. Адаптивность и масштабируемость — гибкая настройка под задачи инфраструктуры и возможность расширения функционала по мере роста сети.
2. Готовые инструменты для мониторинга — предустановленные шаблоны для отслеживания Windows, Linux, сетевого оборудования и IoT-устройств.
3. Многообразие оповещений — поддержка email, SMS, мессенджеров (Telegram, Slack) и интеграция с системами инцидент-менеджмента.
4. Открытая лицензия — доступ к исходному коду позволяет кастомизировать систему, проводить аудит безопасности и снижать зависимость от вендоров.

Данные преимущества делают Zabbix универсальным выбором для комплексного мониторинга гетерогенных сред с требованиями к гибкости и прозрачности решений.