



Подготовка к Демонстрационному
экзамену по 09.02.06

«Сетевое и системное администрирование»

ПРАКТИКУМ

Команда управления компетенции
«Сетевое и системное администрирование»

**ПОДГОТОВКА
К ДЕМОНСТРАЦИОННОМУ
ЭКЗАМЕНУ ПО 09.02.06
«СЕТЕВОЕ И СИСТЕМНОЕ
АДМИНИСТРИРОВАНИЕ»**

ПРАКТИКУМ

Рекомендовано Федеральным учебно-методическим объединением
в системе среднего профессионального образования
по укрупненной группе профессий и специальностей
09.00.00 «Информатика и вычислительная техника» в качестве
учебно-методического пособия для преподавателей при подготовке
обучающихся к демонстрационному экзамену по специальности
09.02.06 «Сетевое и системное администрирование»

МОСКВА
Базальт СПО
МАКС Пресс
2025

УДК 004.7(075.8)
ББК 32.81я73
П44



<https://elibrary.ru/brfwfq>

Рецензенты:

Щербаков С.М. — д.э.н., доцент, заведующий кафедрой Информационных систем и прикладной информатики ФГБОУ ВО «РГЭУ (РИНХ)»;
Сидоров В.В. — к.т.н., доцент, заведующий кафедрой Информатики РГУ нефти и газа (НИУ) имени И.М. Губкина

Золотарёв, Андрей Петрович.

Подготовка к Демонстрационному экзамену по 09.02.06 «Сетевое и системное администрирование» : практикум / А.П. Золотарёв, Д.И. Носенко, А.Г. Уймин [и др.]. – Москва : Базальт СПО; МАКС Пресс, 2025. – 180 с.
ISBN 978-5-317-07434-0

Практикум предназначен для преподавателей и студентов, осваивающих образовательные программы среднего профессионального образования по укрупненным группам «Информационная безопасность», «Информатика и вычислительная техника», «Электроника, радиотехника и системы связи» в целях повышения уровня знаний и умений в области профессиональной деятельности по направлению «Сетевое и системное администрирование» с применением ИТ-инфраструктуры на базе отечественных ИТ технологий.

Материалы, составляющие данную книгу, распространяются на условиях лицензии GNU FDL.

УДК 004.7(075.8)
ББК 32.81я73

ISBN 978-5-317-07434-0

© ООО «Базальт СПО», 2025
© Оформление. ООО «МАКС Пресс», 2025

Оглавление

ПРЕДИСЛОВИЕ	5
Благодарности.....	8
ВВЕДЕНИЕ	9
КОД 09.02.06-1-2025	
Сетевой и системный администратор.....	12
Модуль 1. Настройка сетевой инфраструктуры.....	12
Базовая настройка устройств	16
Настройка ISP.....	25
Создание локальных учетных записей	33
Коммутация, если HQ-SW — виртуальная машина	38
Коммутация, если HQ-SW не является виртуальной машиной ..	41
Настройка безопасного удаленного доступа.....	45
Настройка IP-туннеля между офисами	47
Настройка динамической маршрутизации	50
Настройка динамической трансляции адресов	53
Настройка протокола динамической конфигурации хостов.....	55
Настройка DNS.....	59
Настройка часовых поясов	65
Модуль 2. Организация сетевого администрирования операционных систем	67
Настройка файлового хранилища	71
Настройка служб сетевого времени на базе сервиса chrony.....	76
Настройка ansible	78
Развертывание приложений в Docker	80
Настройка трансляции портов	83
Настройка сервиса Moodle	84
Настройка веб-сервера nginx как обратного прокси-сервера.....	88
Установка Яндекс Браузера.....	91
Начало работы с Кибер Инфраструктурой	92
Установка системы.....	92
О Кибер Инфраструктуре	92
Требования к системе	93
Как получить дистрибутив	94
Свойства стенда.....	96
Установка системы	97

Настройка системы	101
Начало настройки	101
Настройка сети	102
Настройка вычислительного кластера	103
Подключение сервера	105
Настройка сети ВМ	106
Домен. Проект. Пользователи	107
Создание домена и проекта	107
Загрузка образов	109
Вход в портал самообслуживания	110
Портал самообслуживания	111
Создание виртуальной машины	113
Автоматизация	117
Автоматизация (IaC)	117
Установка и подключение OpenStack CLI	118
Создание профиля Putty	120
Работа в CLI	124
Начало работы	124
Openstack CLI	128
Подключение и проверка работы	128
Создание сетей	129
Создание хостов	134
Удаление ресурсов	136
Разворачивание инфраструктуры единым скриптом	138
ПРИЛОЖЕНИЯ	140
Приложение 1	140
Инструкция по застройке стенда для демонстрационного экзамена по КОД 09.02.06-1-2025 «Сетевое и системное администрирование» 2025	140
Приложение 2	144
Установка EcoRouter в GNS3	144
Установка EcoRouter в Альт Виртуализация PVE	148
Базовая настройка EcoRouter	153
Приложение 3	158
Знакомство с Ideco NGFW	158
Установка Ideco NGFW в VirtualBox	161
Установка Ideco NGFW в Альт Виртуализация PVE	166
Базовая настройка Ideco NGFW	172
Приложение 4	177
Развертывание инфраструктуры при помощи автоматизированного скрипта	177

Предисловие

«Технологическая независимость в области ИТ критически важна в современном мире. Это стало очевидным после введения секторальных санкций в 2014 году, а затем после ухода с российского рынка зарубежных ИТ-фирм после 2022 года.

Сегодня отечественное ПО внедряют не только государственные структуры, но и предприятия различных отраслей — как крупные корпорации, так и малый бизнес.

При этом и российские разработчики, получая мощную государственную поддержку и обратную связь от реальных пользователей, постоянно совершенствуют свои программные продукты.

В этих условиях актуальным становится вопрос подготовки кадров, умеющих работать с современным отечественным софтом и оборудованием.

Сегодняшние выпускники завтра придут на производство: в госсектор, бизнес, образование и здравоохранение, поэтому крайне важно готовить студентов к реальным практическим задачам. ИТ-сфера меняется быстро: появляются новые технологии, а требования рынка растут. Для построения реальной технологической независимости страны необходимо постоянно повышать уровень технического образования, совершенствовать учебные программы, чтобы знания, полученные студентами, соответствовали актуальным потребностям рынка.

Ключевую роль в этом процессе играет совместная работа образовательных организаций и ИТ-компаний. Разработчики знают состояние ИТ-рынка, обладают экспертизой, могут сформировать актуальные требования к навыкам и знаниям сотрудников. Они готовы активно участвовать в разработке образовательных программ, учебных пособий, в то время как преподаватели могут методически грамотно и понятно реализовывать учебный процесс.

Важное преимущество дает и использование в обучении свободного программного обеспечения. Оно дает студентам доступ к исходному коду, позволяя не просто пользоваться программами, но и разбираться в их устройстве, изучать код и вносить в него изменения. В будущем такие студенты смогут не только администрировать системы, но и разрабатывать собственные программные продукты, тем самым укрепляя технологическую независимость страны».

Смирнов А.В., председатель совета директоров, ООО «Базальт СПО»

Учебное пособие предназначено для практической подготовки студентов, осваивающих основные профессиональные образовательные программы среднего профессионального образования (далее – СПО) укрупненных групп специальностей «Информатика и вычислительная техника» и «Информационная безопасность», в целях содействия формированию профессиональных компетенций, необходимых в трудовой деятельности сетевого и системно-

го администратора. В системе СПО основным инструментом объективной оценки уровня подготовки студентов является демонстрационный экзамен, который проводится независимыми экспертами по итогам обучения либо при промежуточной аттестации. Данное пособие включает рекомендации по выполнению заданий демонстрационного экзамена, организуемого в рамках государственной итоговой аттестации по завершении освоения образовательной программы СПО по специальности 09.02.06 «Сетевое и системное администрирование». Содержание пособия соответствует требованиям Федерального государственного образовательного стандарта среднего профессионального образования (Федеральный государственный образовательный стандарт (ФГОС) по специальности 09.02.06 «Сетевое и системное администрирование», утвержден приказом Министерства образования и науки РФ от 09.12.2016 №1548 (ред. от 17.12.2020), Федеральный государственный образовательный стандарт (ФГОС) по специальности 09.02.06 «Сетевое и системное администрирование», утвержден приказом Министерства просвещения Российской Федерации от 10.07.2023 №519) и профессиональным квалификационным требованиям, описанным в профстандарте: 06.026 «Системный администратор информационно-коммуникационных систем», утвержденным приказом Министерства труда и социальной защиты Российской Федерации от 29 сентября 2020 года N 680н Системный администратор информационно-коммуникационных систем, поддержано специалистами ФГБОУ ДПО «Институт развития профессионального образования».

Пособие составлено с учетом следующих нормативных документов, регламентирующих процедуру проведения демонстрационного экзамена:

- приказ Министерства просвещения Российской Федерации от 08 ноября 2021 г. № 800 «Об утверждении порядка проведения государственной итоговой аттестации по образовательным программам среднего профессионального образования» (в ред. Приказов Минпросвещения РФ от 05.05.2022 № 311, от 19.01.2023 № 37, от 24.04.2024 № 272, от 22.11.2024 № 812);
- приказ ФГБОУ ДПО ИРПО от 25 апреля 2024 г. № 01-09-139/2024 «Об утверждении Методических указаний по разработке оценочных материалов для проведения демонстрационного экзамена;
- приказ ФГБОУ ДПО ИРПО от 22 июня 2023 г. № П-291 «О введении в действие Методики организации и проведения демонстрационного экзамена».

Авторский коллектив:

ФИО	Должность, место работы
Дегтярев Сергей Сергеевич	г. Ростов-на-Дону, преподаватель, ГБПОУ РО «РКСИ», ведущий эксперт компетенции «Сетевое и системное администрирование», разработчик КОД 09.02.06-5- 2025 Специалист по администрированию сети, асси- стент кафедры ИСиПИ ФГБОУ ВО «РГЭУ (РИНХ)»

ФИО	Должность, место работы
Ефименко Татьяна Ивановна	г. Санкт-Петербург, Колледж туризма и прикладных технологий Санкт-Петербурга, преподаватель, председатель ПЦК цифровых технологий, ведущий эксперт компетенции «Сетевое и системное администрирование», разработчик КОД 09.02.06-2-2025 Системный администратор (Эксплуатация облачных сервисов) и КОД 09.02.06-3-2025 Системный администратор (Эксплуатация объектов сетевой инфраструктуры)
Золотарёв Андрей Петрович	г. Кировск, Ленинградская обл., преподаватель, ГБОУ СПО ЛО «Кировский политехнический техникум», ведущий эксперт компетенции «Сетевое и системное администрирование»
Морозов Илья Михайлович	г. Москва, мастер производственного обучения, РГУ нефти и газа (НИУ) имени И.М. Губкина, ведущий эксперт компетенции «Сетевое и системное администрирование», эксперт НОВОТЕХ, менеджер компетенции «Облачные технологии»
Носенко Дмитрий Игоревич	г. Боровичи, Новгородская обл., преподаватель ОГА ПОУ «Боровичский Педагогический Колледж», ведущий эксперт компетенции «Сетевое и системное администрирование», тренер чемпиона России 2024 года по сетевому и системному администрированию
Уймин Антон Григорьевич	г. Москва, зав. лаб., РГУ нефти и газа (НИУ) имени И.М. Губкина, эксперт НОВОТЕХ, менеджер компетенции «Сетевое и системное администрирование», руководитель команды #au_team
Шальнев Владимир Валентинович	г. Ногинск, Московская обл., преподаватель высшей квалификационной категории по специальности 09.02.06 «Сетевое и системное администрирование», ГБПОУ МО «Ногинский колледж», ведущий эксперт компетенции «Сетевое и системное администрирование»

Благодарности

Коллективу компании «Базальт СПО» за предоставление возможности преподавателям и студентам изучать системное администрирование GNU/Linux-систем на примере ОС семейства «Альт», помощь и содействие в решении технических вопросов и выборе технологий при написании пособия и отдельно Губиной Татьяне Николаевне, к.п.н., руководителю направления по работе с образовательными организациями «Базальт СПО» за помощь в экспертной оценке материалов.

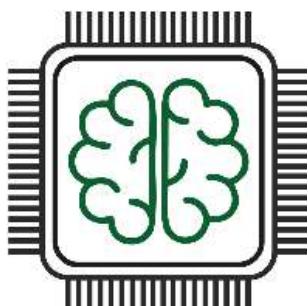
ООО «РДП Инновации» (бренд EcoRouter) за возможность изучать сетевые технологии на примере высокотехнологичного российского оборудования, которое формирует облик современной сетевой инфраструктуры и решает вопросы импортозамещения. Благодаря образовательным инициативам ООО «РДП Инновации» (бренд EcoRouter) у системы образования появляются сетевые инженеры, востребованные в промышленности, телеком-секторе, банках и государственных организациях по всей стране.

Отдельно хотелось бы отметить вклад EcoRouter и «Базальт СПО» в поддержку чемпионатного движения по компетенции «Сетевое и системное администрирование», участники которого демонстрируют высокий уровень профессионального мастерства, наглядно демонстрирующий развитие российской отрасли ИТ.

ООО «Киберпротект» за активную поддержку компетенции «Сетевое и системное администрирование» в области резервного копирования и систем виртуализации.

ООО «Айдеко» за активную поддержку компетенции «Сетевое и системное администрирование» в области сетевой безопасности.

Барышниковой Алене Дмитриевне за вклад в оформление и вычитку текста.



Команда #au_team

Введение

Проведение ГИА в 2025 году в форме демонстрационного экзамена регламентируется локальными актами образовательных организаций, нормативными актами Минпросвещения России и федеральными государственными образовательными стандартами среднего профессионального образования (далее — ФГОС СПО), в соответствии с которыми обучающиеся завершают обучение. Оценочные материалы для проведения ГИА в форме демонстрационного экзамена разработаны прошедшими конкурсным отбор экспертами и открыто размещены на следующих информационных ресурсах:

2025 год — <https://bom.firpo.ru/>;

до 2023 года — <https://om.firpo.ru/archive>.

О ДЕМОНСТРАЦИОННОМ ЭКЗАМЕНЕ > МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ > БРИФИНГИ > АРХИВ ОМ >

БАНК ОЦЕНОЧНЫХ МАТЕРИАЛОВ
информационная система оператора демонстрационного экзамена базового и профильного уровней по образовательным программам среднего профессионального образования, предназначенная для размещения в общем доступе разработанных комплектов оценочной документации для проведения демонстрационного экзамена

Комплекты оценочной документации для проведения демонстрационного экзамена в 2025 году

Все	05.00.00	07.00.00	08.00.00	09.00.00	10.00.00	11.00.00	12.00.00	13.00.00	14.00.00	15.00.00	18.00.00	
	19.00.00	20.00.00	21.00.00	22.00.00	23.00.00	24.00.00	25.00.00	26.00.00	27.00.00	29.00.00	31.00.00	33.00.00
	35.00.00	36.00.00	38.00.00	39.00.00	40.00.00	42.00.00	43.00.00	44.00.00	46.00.00	49.00.00	54.00.00	

Информационная система оператора демонстрационного экзамена базового и профильного уровней по образовательным программам среднего профессионального образования, предназначенная для размещения в общем доступе разработанных комплектов оценочной документации для проведения демонстрационного экзамена:

ФГОС 09.02.06 Сетевое и системное администрирование (приказ №1548 от 09 декабря 2016)
(<https://spolab.firpo.ru/storage/NPD//0JLPhkk6Wi1rTWUlJWUDhifmY1EX5JLxyZAczvbA.docx>)



<p>09.02.06-1-2025: Сетевой и системный администратор (https://bom.firpo.ru/Public/2359)</p>	
<p>09.02.06-5-2025: Специалист по администрированию сети (https://bom.firpo.ru/Public/2369)</p>	
<p>ФГОС 09.02.06 Сетевое и системное администрирование (приказ №519 от 10 июля 2023) (https://spolab.firpo.ru/npdv2/category-doc/get/3774)</p>	
<p>09.02.06-2-2025: Системный администратор (Эксплуатация облачных сервисов) (https://bom.firpo.ru/Public/2361)</p>	
<p>09.02.06-3-2025: Системный администратор (Эксплуатация объектов сетевой инфраструктуры) (https://bom.firpo.ru/Public/2363)</p>	
<p>09.02.06-4-2025: Системный администратор (Эксплуатация операционных систем) (https://bom.firpo.ru/Public/2365)</p>	

Видеообзор комплекта оценочных материалов:

КОД 09.02.06-1-2025

Диплом о профессиональной квалификации

Конкретные оцениваемые действия

Показатель	Действие
Создание подсетей и настройка обмена данными	Настроены подинтерфейсы на HQ-RTR в соответствии с заданием

Учитывается количество подинтерфейсов и соответствствие параметров. При наличии допустимого количества отклонений действие будет частично засчитано. При превышении количества отклонений действие считается невыполненным.

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

- Сервер HQ-SRV должен находиться в ID VLAN 100
- Клиент HQ-CLI в ID VLAN 200
- Создайте подсеть управления с ID VLAN 999
- Основные сведения о настройке коммутатора и выбора реализации подразделения на VLAN занесите в отчет

Важно!
Номера пунктов задания и их содержание и количеству и составу действий.

Но имеют явную связь между собой, что исключает проверку действий которые не отвечают заданным и выполнение заданий не влияющих на оценку.

Вид аттестации	Уровень ДЭ	Кол-во оцениваемых действий
ПЛА	ДЭ	11
ГПА	ДЭ БУ ДЭ ПУ	18 23



https://vkvideo.ru/video-219561594_456239715?list=ln-dyXhMWQqd78jblZ1Ot&ref_domain=bom.firpo.ru

Видеообзор подготовки к ДЭ:

Чат компетенции, структурированный по темам (https://t.me/+Sz-uToWW2zc5OWMy)	
ДЭ 2024 (https://vkvideo.ru/video/playlist/-228030577_1)	
Знакомство с технологиями EcoRouter в СиСА 2025 (https://vkvideo.ru/video/playlist/-228030577_4)	
Знакомство с технологиями IDECO FW в СиСА 2025 (https://vkvideo.ru/video/playlist/-228030577_6)	

КОД 09.02.06-1-2025

Сетевой и системный администратор

Модуль 1. Настройка сетевой инфраструктуры

Модуль 1

Настройка сетевой инфраструктуры

Вид аттестации/уровень ДЭ

ПА, ГИА ДЭ БУ, ГИА ДЭ ПУ (инвариантная часть)

Задание:

Необходимо разработать и настроить инфраструктуру информационно-коммуникационной системы согласно предложенной топологии (см. рисунок 1). Задание включает базовую настройку устройств:

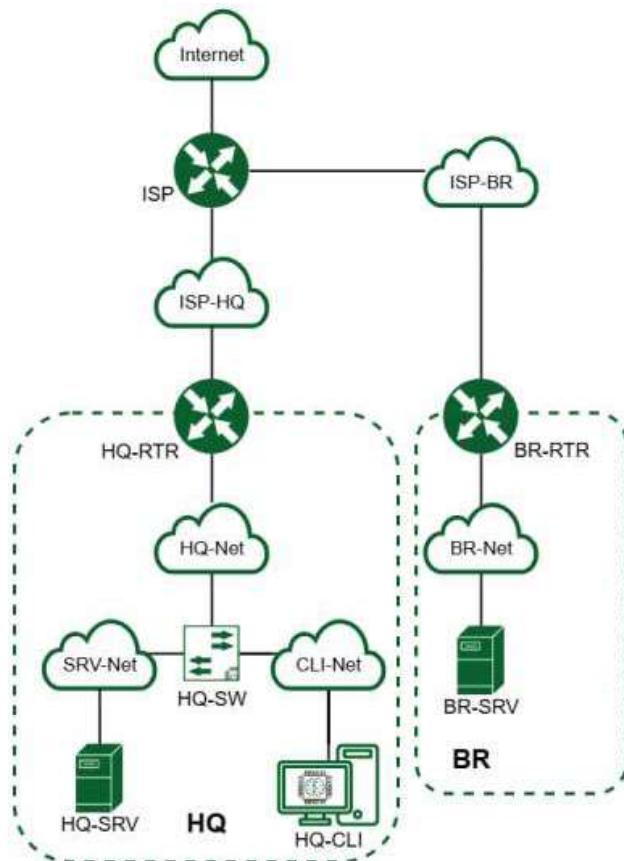


Рисунок 1. Топология сети

Задача:

- Настроить на всех устройствах соответствующие имена
- Настроить на устройства соответствующие IP-адреса

Вариант реализации:

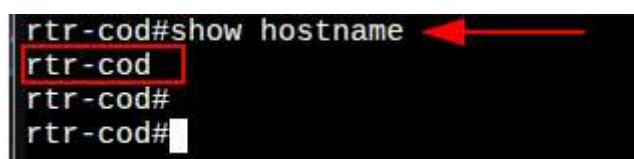
rtr-cod (ecorouter):

Назначение имени на устройство:

- Для назначения имени устройства согласно требованиям задания используем следующие команды:
 1. переходим в режим администрирования (**enable**);
 2. переходим в режим конфигурации (**configure terminal**);
 3. задаём имя устройству (**hostname <NAME>**);
 4. задаём доменное имя (**ip domain-name <DOMAIN_NAME>**);
 5. сохраняем конфигурацию (**write memory**).

```
ecorouter>enable  
ecorouter#configure terminal  
  
Enter configuration commands, one per line. End with CNTL/Z.  
ecorouter(config)#hostname rtr-cod  
rtr-cod(config)#ip domain-name cod.ssa2026.region  
rtr-cod(config)#write memory  
  
Building configuration...  
  
rtr-cod(config)#[
```

- Проверить имя устройства можно командой **show hostname** из режима администрирования (**enable**):



```
rtr-cod#show hostname
```

- Проверить доменное имя устройства можно командой **show running-config | include domain-name** из режима администрирования (**enable**):

```
rtr-cod#show running-config | include domain-name  
ip domain-name cod.ssa2026.region  
rtr-cod#  
rtr-cod#
```

Назначение IP-адресов на устройство:

- Основные понятия касающиеся EcoRouter:
 1. **Порт (port)** – это устройство в составе EcoRouter, которое работает на физическом уровне (L1);
 2. **Интерфейс (interface)** – это логический интерфейс для адресации, работает на сетевом уровне (L3);
 3. **Service instance (Сабинтерфейс, SI, Сервисный интерфейс)** является логическим сабинтерфейсом, работающим на канальном уровне (L2) и связывает L1, L2 и L3 уровни:
 - Данный вид интерфейса необходим для соединения физического порта с интерфейсами L3, интерфейсами bridge, портами;
 - Используется для гибкого управления трафиком на основании наличия меток VLANов в фреймах, или их отсутствия;
 - Сквозь сервисный интерфейс проходит весь трафик, приходящий на порт.
- Таким образом, для того чтобы назначить IPv4-адрес на EcoRouter, необходимо придерживаться следующего алгоритма в общем виде:
 1. Создать интерфейс с произвольным именем и назначить на него IPv4-адрес;
 2. В режиме конфигурирования порта создать service-instance с произвольным именем:
 - указать (инкапсулировать) что будет обрабатываться тегированный или не тегированный трафик;
 - указать в какой интерфейс (ранее созданный) нужно отправить обработанные кадры.
- Посмотреть физические порты можно командой **show port brief** из режима администрирования (**enable**):
 - порт **te0** направлен в сторону BM **isp**;
 - порт **te1** направлен в сторону BM **fw-cod**.

```
rtr-cod#show port brief  
Name          Physical Admin   Lacp    Description  
-----  
te0           UP      UP      *  
te1           UP      UP      *  
rtr-cod#
```

- Создадим интерфейс с именем **isp** и назначим на него IP-адрес **178.207.179.4/29**, также зададим для данного интерфейса описание (description - опциональный, необязательный параметр):

```
rtr-cod(config)#interface isp
rtr-cod(config-if)#ip address 178.207.179.4/29
rtr-cod(config-if)#description "Connecting to an ISP provider"
rtr-cod(config-if)#exit
rtr-cod(config)#

```

- Создадим интерфейс с именем **fw-cod** и назначим на него IP-адрес **172.16.1.1/30**, также зададим для данного интерфейса описание (description - опциональный, необязательный параметр):

```
rtr-cod(config)#interface fw-cod
rtr-cod(config-if)#ip address 172.16.1.1/30
rtr-cod(config-if)#description "Connecting to fw-cod"
rtr-cod(config-if)#exit
rtr-cod(config)#

```

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (**enable**):
 - созданные интерфейсы пока не добавлены в какие-либо **Service instance**, а значит не привязаны и к порту, отсюда и статус **down**

Interface	IP-Address	Status	VRF
isp	178.207.179.4/29	down	default
fw-cod	172.16.1.1/30	down	default

- В режиме конфигурирования порта **te0** необходимо создать service-instance с произвольным именем, например **te0/isp**:
 - также необходимо указать (инкапсулировать) что будет обрабатываться не тегированный трафик (**untagged**);
 - и указать в какой интерфейс (ранее созданный с именем **isp**) нужно отправлять обработанные кадры.

```
rtr-cod(config)#port te0
rtr-cod(config-port)#service-instance te0/isp
rtr-cod(config-service-instance)#encapsulation untagged
rtr-cod(config-service-instance)#connect ip interface isp
rtr-cod(config-service-instance)#exit
rtr-cod(config-port)#exit

```

```
rtr-cod(config)#
```

- В режиме конфигурирования порта **te1** необходимо создать **service-instance** с произвольным именем, например **te1/fw-cod**:
 - также необходимо указать (инкапсулировать) что будет обрабатываться не тегированный трафик (**untagget**);
 - и указать в какой интерфейс (ранее созданный с именем **fw-cod**) нужно отправлять обработанные кадры.

```
rtr-cod(config-port)#service-instance te1/fw-cod
```

```
rtr-cod(config-service-instance)#encapsulation untagged
```

```
rtr-cod(config-service-instance)#connect ip interface fw-cod
```

```
rtr-cod(config-service-instance)#exit
```

```
rtr-cod(config-port)#exit
```

```
rtr-cod(config)# write memory
```

```
Building configuration...
```

```
rtr-cod(config)#
```

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (**enable**):

Interface	IP-Address	Status	VRF
isp	178.207.179.4/29	up	default
fw-cod	172.16.1.1/30	up	default

- Проверить созданные Service instance можно командой **show service-instance brief** из режима администрирования (**enable**):

Total instances: 2				
> - Active: N - Not connected: D - Down:				
PORT NAME	SERVICE INSTANCE NAME	ENDPOINT TYPE	ENDPOINT NAME	BRIEF DESCRIPTION
> te0	te0/isp	iface	isp	-
> te1	te1/fw-cod	iface	fw-cod	-

- IP-адрес шлюза по умолчанию на данном устройстве на текущий момент не задаётся (будет рассмотрено далее, в соответствующем разделе), т.к. по условиям задания:

- Маршрутизатор ЦОД должен получать маршрут по умолчанию по BGP
 - Ручное создание маршрута по умолчанию ЗАПРЕЩЕНО!
- Но связность с Интернет провайдером ISP проверить стоит:

```
rtr-cod#ping 178.207.179.1 ←
PING 178.207.179.1 (178.207.179.1) 56(84) bytes of data.
64 bytes from 178.207.179.1: icmp_seq=1 ttl=64 time=39.9 ms
64 bytes from 178.207.179.1: icmp_seq=2 ttl=64 time=193 ms
64 bytes from 178.207.179.1: icmp_seq=3 ttl=64 time=23.8 ms

--- 178.207.179.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 23.758/85.655/193.319/76.414 ms
rtr-cod#
```

rtr-a (ecorouter):

Назначение имени на устройство:

- Реализация аналогично rtr-cod, за исключением соответствующего доменного имени:
 - имя устройства должно быть:

```
rtr-a#show hostname
rtr-a
rtr-a#
rtr-a#
```

- доменное имя на устройстве должно быть:

```
rtr-a#show running-config | include domain-name
ip domain-name office.ssa2026.region
rtr-a#
rtr-a#
```

Назначение IP-адресов на устройство:

- Реализация аналогично rtr-cod, за исключением того, что на базе физического порта te1 должны быть созданы интерфейсы и Service instance с целью обработки тегированного трафика для предоставления возможности маршрутизации между VLAN (рассмотрено далее)
 - должен быть создан интерфейс для подключения к Интернет провайдеру ISP:

```
rtr-a#show ip interface brief
Interface          IP-Address      Status           VRF
-----+-----+-----+-----+
isp        178.207.179.28/29    down            default
rtr-a#
```

- должен быть создан Service instance на порту te0
- на созданный Service instance реализована обработка не тегированного трафика
- в созданный Service instance добавлен интерфейс для подключения к Интернет провайдеру

```
rtr-a#show ip interface brief
Interface          IP-Address      Status           VRF
-----+-----+-----+-----+
isp        178.207.179.28/29    up              default
rtr-a#show service-instance brief
  Total instances: 1
  > - Active:
  N - Not connected:
  D - Down:
  PORT          SERVICE INSTANCE   ENDPOINT     ENDPOINT   BRIEF
  NAME          NAME             TYPE         NAME       DESCRIPTION
  > te0          te0/isp          iface        isp        -
rtr-a#
```

- В отличие от **rtr-cod**, на **rtr-a** по заданию нет никаких требований про настройку BGP, а значит для доступа в сеть Интернет, маршрут по умолчанию можно задать вручную:
 - переходим в режим конфигурации (**configure terminal**)
 - с помощью команды **ip route <IP_NETWORK/PREFIX> <NEXTHOP_IP_ADDRESS>** задаём маршрут по умолчанию (шлюз)

```
rtr-a(config)#ip route 0.0.0.0/0 178.207.179.25
```

```
rtr-a(config)#
```

- Проверить назначенный маршрут по умолчанию можно командой **show ip route static** из режима администрирования (**enable**):

```
rtr-a#show ip route static
IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.25 to network 0.0.0.0
S*      0.0.0.0/0 [1/0] via 178.207.179.25,  isp
rtr-a#
```

- Так же стоит проверить доступ в сеть Интернет:

```
rtr-a#ping 77.88.8.8 ←  
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.  
64 bytes from 77.88.8.8: icmp_seq=1 ttl=51 time=104 ms  
64 bytes from 77.88.8.8: icmp_seq=2 ttl=51 time=99.5 ms  
64 bytes from 77.88.8.8: icmp_seq=3 ttl=51 time=38.6 ms  
64 bytes from 77.88.8.8: icmp_seq=4 ttl=51 time=37.6 ms  
--- 77.88.8.8 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3058ms  
rtt min/avg/max/mdev = 37.625/69.814/103.518/31.719 ms  
rtr-a#
```

- Реализуем создание под-интерфейсов для дальнейшем маршрутизации между VLAN-ами:
 - Создаём интерфейсы с произвольными именами для каждого VLAN-а и назначаем на них IP-адреса:

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (**enable**):
 - созданные интерфейсы пока не добавлены в какие-либо **Service instance**, а значит не привязаны и к порту, отсюда и статус **down**

Interface	IP-Address	Status	VRF
isp	178.207.179.28/29	up	default
vl100	172.20.10.254/24	down	default
vl200	172.20.20.254/24	down	default
vl300	172.20.30.254/24	down	default

- на базе физического интерфейса **te1** для каждого VLAN-а создаём **service-instance** с инкапсуляцией соответствующих тегов (**VID**) и подключением необходимых интерфейсов:

Пояснение к листингу команд указанному ниже

Операции над метками в сервисных интерфейсах:

- Есть три варианта операций над метками: **удаление** существующей метки/меток, **добавление** новой метки (меток) и **трансляция** метки/меток из одного значения в другое.

Пояснение команд:

- Указание номера, обрабатываемого VLAN выполняется на service-instance с помощью команды **encapsulation dot1q <VID> exact**
 - опция **exact** показывает, что под это правило попадут кадры только с меткой равной **<VID>**
 - слово **exact** писать не обязательно, так как это поведение по умолчанию и в выводе **show run** это слово не отображается
- Указание выполняемой операции выполняется на service-instance с помощью команды **rewrite pop <№>**
 - ключ **1** показывает, что снимаем только одну, верхнюю метку, на L3 кадр должен поступать без признаков VLAN

```
rtr-a(config)#port te1
rtr-a(config-port)#service-instance te1/vl100
rtr-a(config-service-instance)#encapsulation dot1q 100
rtr-a(config-service-instance)#rewrite pop 1
rtr-a(config-service-instance)#connect in
rtr-a(config-service-instance)#connect ip interface vl100
rtr-a(config-service-instance)#exit
rtr-a(config-port)#
rtr-a(config-port)#service-instance te1/vl200
rtr-a(config-service-instance)#encapsulation dot1q 200
```

```
rtr-a(config-service-instance)#rewrite pop 1
rtr-a(config-service-instance)#connect ip interface vl200
rtr-a(config-service-instance)#exit
rtr-a(config-port)#
rtr-a(config-port)#service-instance te1/vl300
rtr-a(config-service-instance)#encapsulation dot1q 300
rtr-a(config-service-instance)#rewrite pop 1
rtr-a(config-service-instance)#connect ip interface vl300
rtr-a(config-service-instance)#exit
rtr-a(config-port)#exit
rtr-a(config)#write memory
Building configuration...
```

```
rtr-a(config)#
```

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (enable):

```
rtr-a#show ip interface brief
-----+-----+-----+-----+
Interface      IP-Address      Status      VRF
-----+-----+-----+-----+
isp          178.207.179.28/29  up          default
vl100        172.20.10.254/24   up          default
vl200        172.20.20.254/24   up          default
vl300        172.20.30.254/24   up          default
-----+-----+-----+-----+
rtr-a#
```

- Проверить созданные Service instance можно командой **show service-instance brief** из режима администрирования (enable):

```
rtr-a#show service-instance brief
Total instances: 4
> - Active:
N - Not connected:
D - Down:
-----+-----+-----+-----+-----+
PORT      SERVICE INSTANCE      ENDPOINT      ENDPOINT      BRIEF
NAME       NAME             TYPE        NAME        DESCRIPTION
-----+-----+-----+-----+-----+
> te0      te0/isp           iface       isp          -
> te1      te1/vl100          iface       vl100        -
> te1      te1/vl200          iface       vl200        -
> te1      te1/vl300          iface       vl300        -
-----+-----+-----+-----+-----+
rtr-a#
```


Вариант реализации:

fw-cod (ideco):

Создание учетной записи администратора

- Создайте учетную запись администратора после уведомления **Создание аккаунта администратора**:

```
Ideco NGFW 21.5.99
-----
Внимание! Аккаунт администратора отсутствует.
Требуется предварительно его создать.
Создание аккаунта администратора.

Введите новый логин и нажмите Enter.

#
```

- Требования к логину и паролю:
 - Логин:
 - Не должен начинаться с цифры.
 - Не должен содержать специальных и пробельных символов, кроме дефиса -.
 - **Длина логина** - от 1 до 31 символа включительно.
 - Пароль:
 - **Длина пароля** - от 10 до 42 символов.
 - **Содержит строчные и заглавные латинские буквы.**
 - **Содержит цифры.**
 - **Содержит специальные символы** (! # \$ % & ' * + и другие).
- Если пароль не соответствует требованиям политики безопасности, то появится надпись с информацией, что пароль ненадежен. Потребуется ввести новый пароль с учетом требований к нему.
- Создадим пользователя **admin** с паролем **P@ssw0rd1234**:

```
Введите новый логин и нажмите Enter.

# admin1

Введите новый пароль и нажмите Enter.

Введите 'b' и нажмите Enter для возврата.
#2

Повторите пароль и нажмите Enter.

Введите 'b' и нажмите Enter для возврата.

▶ Аккаунт администратора создан успешно.
```

Настройка интерфейса управления

- Для корректной идентификации сетевой карты используйте MAC-адрес сетевой карты
- Для настройки Ideco NGFW Novum через веб-интерфейс настройте Control Plane интерфейс в локальном меню
 - Control Plane - интерфейс администрирования, используется для настройки NGFW Novum через браузер и должен иметь свой выход в Интернет
- Выполните вход из под созданного пользователя **admin** с паролем **P@ssw0rd1234**

Скриншот терминального окна с текстом на русском языке:

```
Нажмите любую клавишу для перехода к локальному меню.  
Вход в локальное меню.  
Введите логин и нажмите Enter.  
# admin  
Введите пароль и нажмите Enter.  
Введите 'b' и нажмите Enter для возврата.  
#  
Управление сервером
```

На скриншоте есть красные выделения надписей "Введите логин и нажмите Enter.", "Введите пароль и нажмите Enter." и "Управление сервером". Курсор виден перед "admin" в строке "# admin".

- Перейдите в раздел **Ethernet-интерфейсы (3) → Создать интерфейс (3)**:

```

1. Консоль
2. Физические порты
3. Ethernet-интерфейсы
4. Отключить все статические маршруты и добавить новый
5. Выбор типа кластерной сети
6. Включить доступ к веб-интерфейсу из внешней сети
7. Включить доступ к серверу по SSH из Интернет
8. Включить доступ к серверу по SSH из локальных сетей
9. Включить режим 'Разрешить Интернет всем'
10. Сбросить блокировки по IP
11. Отключить пользовательский межсетевой экран
12. Создать новый бэкап
13. Восстановить из бэкапа
14. Мгновенно восстановить из бэкапа
15. Включить доступ Удаленного Помощника
16. Контакты технической поддержки
17. Восстановиться на предыдущую версию
18. Перезагрузка сервера
19. Отключить сервер
. Выход

```

Введите номер пункта и нажмите Enter.

3

```

1. Показать список интерфейсов
2. Включить/Отключить интерфейс
3. Создать интерфейс
4. Изменить интерфейс
5. Удалить интерфейс

```

Введите номер пункта и нажмите Enter.

Введите 'c' и нажмите Enter для отмены.

3

Выберите порт:

N	Порт	MAC-адрес	Назначение	Производитель сетевой карты
1	eth0_01	bc:24:11:95:c8:01	Сервис	Red Hat, Inc. Virtio network device
2	eth1_da	bc:24:11:55:66:da	Сервис	Red Hat, Inc. Virtio network device
3	eth2_4f	bc:24:11:02:c4:4f	Сервис	Red Hat, Inc. Virtio network device

Введите номер пункта и нажмите Enter.

Введите 'c' и нажмите Enter для отмены.

#

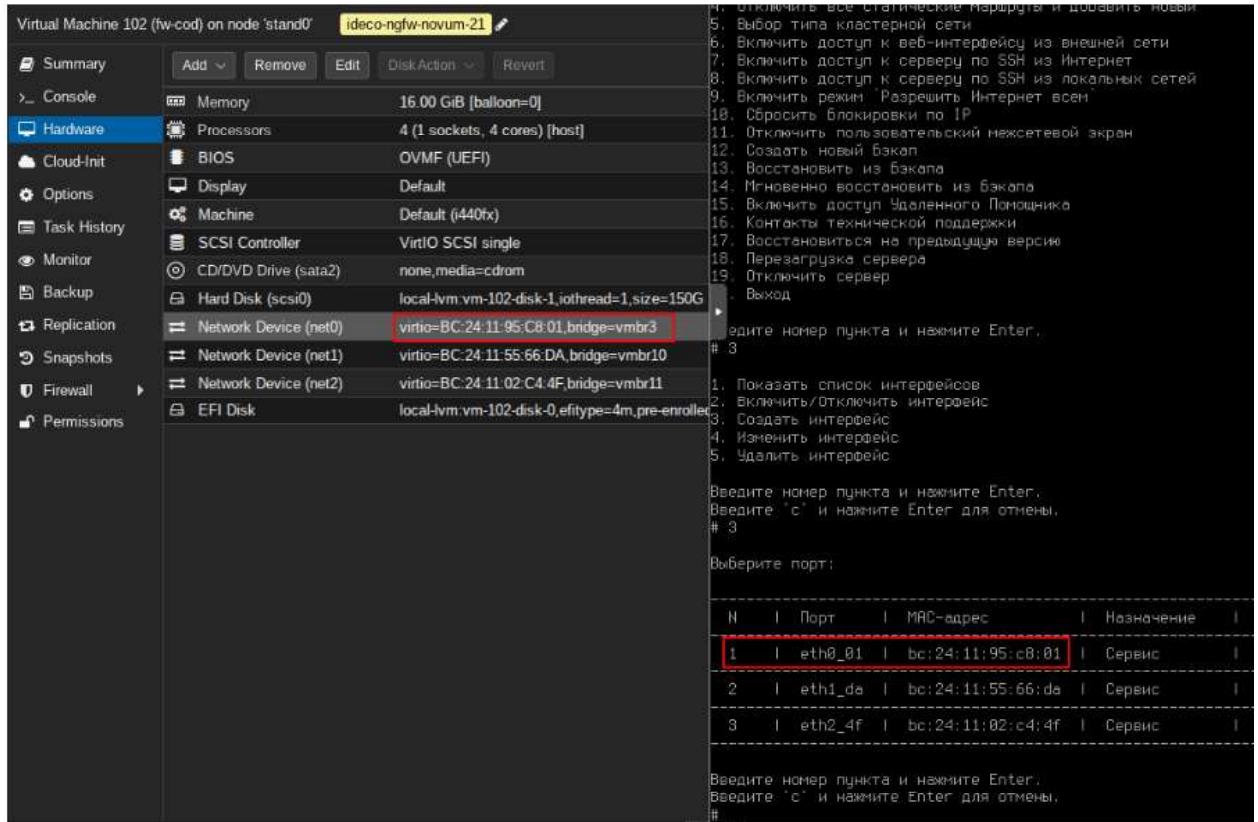
Пояснения в контексте текущего задания:

- Поскольку Ideco рекомендуется настраивать через веб-интерфейс, надо настроить IP-адрес для этого доступа
- Поскольку в cod.ssa2026.region между fw-cod и sw1-cod необходимо организовать агрегированное соединение 802.3ad,
 - а также за маршрутизацию между VLAN-ами по топологии будет отвечать fw-cod,
 - то так называемые под-интерфейсы необходимо реализовывать поверх агрегированного канала 802.3ad,
 - в консоле Ideco - это реализовать не получится, править конфигурационные файлы в ideco плохо, можно лишиться технической поддержки.
- Поэтому как один из способов доступа к веб-интерфейсу это:
 - использование интерфейса в сторону rtr-cod,
 - после реализации туннеля и маршрутизации между rtr-cod и rtr-a,
 - а также коммутации между sw1-a и sw2-a,

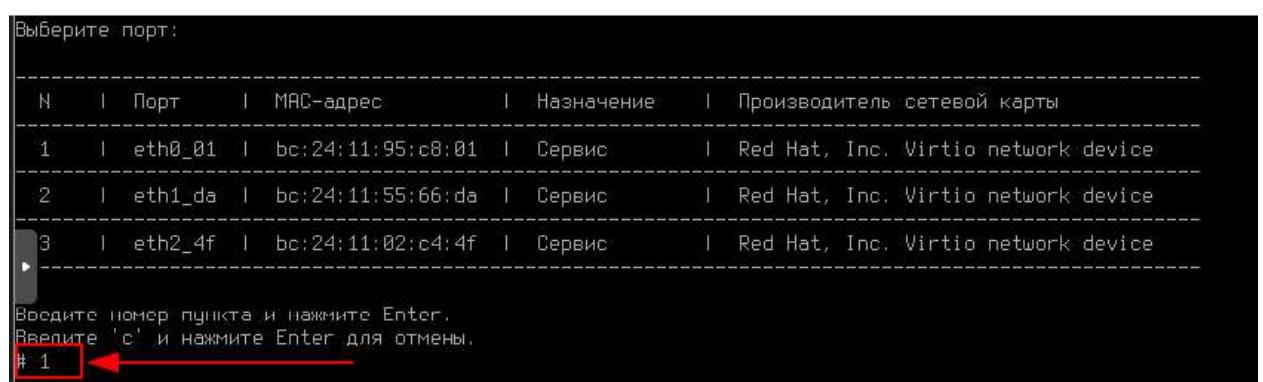
- у нас появится возможность конфигурировать fw-cod через веб-интерфейс, например с cli1-а или cli2-а,
- вследствии чего и полная связность между устройствами cod.ssa2026.region и office.ssa2026.region

Продолжение настройки:

- Выберите порт для доступа к веб-интерфейсу:
 - сравнив MAC-адреса на уровне виртуальной машины



- Выберите порт для доступа к веб-интерфейсу:
 - в текущем случае выбирается порт в сторону виртуальной машины **rtr-cod**



- Введите имя интерфейса:

Введите имя интерфейса (или оставьте пустым) и нажмите Enter.

Введите 'b' и нажмите Enter для возврата.

Введите 'c' и нажмите Enter для отмены.

rtr-cod

- Выберите роль **LAN**:

Выберите роль для интерфейса:

1. CP
2. LAN
3. WAN

Введите номер пункта и нажмите Enter.

Введите 'c' и нажмите Enter для отмены.

2

- Выберите **Корневой контекст**:

Выберите VCE:

1. Системный контекст

Введите номер пункта и нажмите Enter.

Введите 'c' и нажмите Enter для отмены.

1

- Настройте локальную сеть **Вручную**:

Настроить адресацию на интерфейсе

1. Автоматически через DHCP
2. Вручную
3. Без адресации

Введите номер пункта и нажмите Enter.

Введите 'c' и нажмите Enter для отмены.

2

- Ведите локальный IP-адрес и маску подсети в формате ip/маска и нажмите **Enter**:

Введите IP/префикс и нажмите Enter.

Введите 'b' и нажмите Enter для возврата.

Введите 'c' и нажмите Enter для отмены.

172.16.1.2/30

Интерфейс успешно настроен.

- Проверить можно выбрав соответствующий пункт меню:

```

1. Показать список интерфейсов
2. Включить/Отключить интерфейс
3. Создать интерфейс
4. Изменить интерфейс
5. Удалить интерфейс

Введите номер пункта и нажмите Enter.
Введите 'c' и нажмите Enter для отмены.
# 1 ←

Выберите интерфейс:



| N | Состояние | Название | Роль | VCE                | Порт    | IP            | Статус |
|---|-----------|----------|------|--------------------|---------|---------------|--------|
| 1 | Вкл.      | rtr-cod  | LAN  | Системный контекст | eth0_01 | 172.16.1.2/30 | up     |



Введите номер пункта и нажмите Enter.
Введите 'c' и нажмите Enter для отмены.
#

```

- Также при вводе с клавиатуры "с" и нажатие **Enter** или же сочетание клавиш **Ctrl + D**:
 - Можно увидеть адрес и порт для доступа к веб-интерфейсу для дальнейшей настройки

```

Добро пожаловать в панель мониторинга сервера Ideco NGFW 21.5.99!

Название сервера: без названия fddcbdbb-81a5-420d-bece-e00da33673ff
Состояние локальных интерфейсов: Настроены
Доступ Удаленного Помощника: Отключено
Режим Разрешить Интернет всем : Отключено
Доступ в веб-интерфейс: Доступен
Доступ к веб-интерфейсу из внешней сети: Отключено

Адреса веб-интерфейса:
https://172.16.1.2:8443

В случае возникновения ошибок на сервере, пожалуйста,
обратитесь в техподдержку:

Email: help@ideco.ru
Портал тех. поддержки: help.ideco.ru
Время работы тех. поддержки: ideco.ru/support

нажмите любую клавишу для перехода к локальному меню,
less Ctrl+R if you don't see the symbols above.

```

Вариант реализации:

rtr-cod (ecorouter):

Базовая настройка BGP:

- Запустите протокол BGP, указав нужную автономную систему, командой: **router bgp <№>**:

```
rtr-cod(config)#router bgp 64500
```

```
rtr-cod(config-router)#
```

- Указать уникальный идентификатор маршрутизатора в протоколе BGP, командой: **bgp router-id <IP>**:

```
rtr-cod(config-router)#bgp router-id 178.207.179.4
```

```
rtr-cod(config-router)#
```

- Сконфигурируйте BGP соседство с Интернет провайдером **ISP**, указав адрес соседа и номер локальной AS, используя команду: **neighbor <NEIGHBOR_IP> remote-as <\$>**:

```
rtr-cod(config-router)#neighbor 178.207.179.1 remote-as 31133
```

```
rtr-cod(config-router)#exit
```

```
rtr-cod(config)#write memory
```

```
Building configuration...
```

```
rtr-cod(config)#
```

- Проверить состояние всех соединений BGP можно командой **show ip bgp summary** из режима администрирования (**enable**):

```
rtr-cod#show ip bgp summary
BGP router identifier 178.207.179.4, local AS number 64500
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor      V   AS     MsgRcv   MsgSen   TblVer  InQ    OutQ  Up/Down  State/PfxRcd
178.207.179.1  4   31133  7          5        2       0      0      00:01:34   1

Total number of neighbors 1
Total number of Established sessions 1
rtr-cod#
```

- Также по условиям задания **rtr-cod** должен получать маршрут по умолчанию по BGP

- Проверить маршрут по умолчанию можно командой **show ip route** из режима администрирования (**enable**):

```
rtr-cod#show ip route
Codes: C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default

IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.1 to network 0.0.0.0

B*      0.0.0.0/0 [20/0] via 178.207.179.1, isp, 00:00:16
C        172.16.1.0/30 is directly connected, fw-cod
C        178.207.179.0/29 is directly connected, isp
rtr-cod#
```

- Проверить доступ в сеть Интернет:

```
rtr-cod#ping 77.88.8.8 ←
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=51 time=187 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=51 time=89.9 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=51 time=91.9 ms
64 bytes from 77.88.8.8: icmp_seq=4 ttl=51 time=95.2 ms

--- 77.88.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss time 3004ms
rtt min/avg/max/mdev = 89.859/115.988/186.932/41.004 ms
rtr-cod#
```

Вариант реализации:

rtr-cod (ecorouter):

Настройка туннеля GRE между устройствами:

- Создайте интерфейс туннеля с именем **tunnel.0**:

```
rtr-cod(config)#interface tunnel.0
```

```
rtr-cod(config-if-tunnel)#
```

- Назначьте ip адрес в соответствие с требованиями задания:

```
rtr-cod(config-if-tunnel)#ip address 10.10.10.1/30
```

```
rtr-cod(config-if-tunnel)#
```

- Задайте режим работы туннеля GRE и адресов начала (источника - **rtr-cod**) и конца туннеля (назначения - **rtr-a**):

```
rtr-cod(config-if-tunnel)#ip tunnel 178.207.179.4 178.207.179.28 mode gre
```

```
rtr-cod(config-if-tunnel)# exit
```

```
rtr-cod(config)#write memory
```

```
Building configuration...
```

```
rtr-cod(config)#
```

- Для просмотра состояния туннеля используется команда **show interface tunnel.0** из режима администрирования (**enable**):

```
rtr-cod#show interface tunnel.0
Interface tunnel.0 is up
Snmp index: 8
Ethernet address: (port not configured)
MTU: 1476
Tunnel source: 178.207.179.4
Tunnel destination: 178.207.179.28
Tunnel mode: GRE
Tunnel keepalive: disabled
NAT: no
ARP Proxy: disable
ICMP redirects on, unreachable on
IP URPF is disabled
Label switching is disabled
<UP,BROADCAST,RUNNING,NOARP,MULTICAST>
inet 10.10.10.1/30 broadcast 10.10.10.3/30
total input packets 0, bytes 0
total output packets 0, bytes 0
rtr-cod#
```

rtr-a (ecorouter):

Настройка туннеля GRE между устройствами:

- Реализация аналогично rtr-cod, за исключением соответствующего IP-адресов источника и назначения:
 - состояния туннеля должно быть:

```
rtr-a#show interface tunnel.0
Interface tunnel.0 is up
Snmp index: 9
Ethernet address: (port not configured)
MTU: 1476
Tunnel source: 178.207.179.28
Tunnel destination: 178.207.179.4
Tunnel mode: GRE
Tunnel keepalive: disabled
NAT: no
ARP Proxy: disable
ICMP redirects on, unreachable on
IP URPF is disabled
Label switching is disabled
<UP,BROADCAST,RUNNING,NOARP,MULTICAST>
inet 10.10.10.2/30 broadcast 10.10.10.3/30
total input packets 0, bytes 0
total output packets 0, bytes 0
rtr-a#
```

- Должна быть связность по туннелю:

```
rtr-a#show ip interface brief ←  
Interface          IP-Address      Status           VRF  
---  
isp                178.207.179.28/29 up               default  
vl100              172.20.10.254/24 up               default  
vl200              172.20.20.254/24 up               default  
vl300              172.20.30.254/24 up               default  
tunnel.0            10.10.10.2/30  up               default  
  
rtr-a#ping 10.10.10.1 ←  
PING 10.10.10.1 (10.10.10.1) 56(84) bytes of data.  
64 bytes from 10.10.10.1: icmp_seq=1 ttl=64 time=106 ms  
64 bytes from 10.10.10.1: icmp_seq=2 ttl=64 time=96.8 ms  
64 bytes from 10.10.10.1: icmp_seq=3 ttl=64 time=111 ms  
64 bytes from 10.10.10.1: icmp_seq=4 ttl=64 time=93.2 ms  
  
--- 10.10.10.1 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3005ms  
rtt min/avg/max/mdev = 93.181/101.848/111.388/7.220 ms  
rtr-a#
```

Вариант реализации:

rtr-cod (ecorouter):

Настройка dynamic PAT:

- Определяемся с внутренними (**inside**) и внешними (**outside**) интерфейсами с точки зрения технологии NAT:
 - интерфейс **isp** - внешний (**outside**);
 - интерфейс **fw-cod** - внутренний (**inside**)

```
rtr-cod#show ip interface brief
Interface          IP-Address      Status           VRF
-----
tunnel.0          10.10.10.1/30   up               default
isp              178.207.179.4/29  up               default
fw-cod            172.16.1.1/30   up               default
rtr-cod#show interface description
Interface          Status          Protocol        Description
-----
isp                up             up              "Connecting to an ISP provider"
fw-cod              up             up              "Connecting to fw-cod"
tunnel.0           up             up
rtr-cod#
```

- Назначаем интерфейс **isp** как **ip nat outside**:

```
rtr-cod(config)#interface isp
rtr-cod(config-if)#ip nat outside
rtr-cod(config-if)#exit
rtr-cod(config)#

```

- Назначаем интерфейс **fw-cod** как **ip nat inside**:

```
rtr-cod(config)#interface fw-cod
rtr-cod(config-if)#ip nat inside
rtr-cod(config-if)#exit
rtr-cod(config)#

```

- Создаём **nat pool**, чтобы указать диапазоны IP-адресов, который в дальнейшем будут попадать под правила трансляции:
 - необходимо не только указать диапазон IP-адресов из сети между **rtr-cod** и **fw-cod**
 - но и диапазоны IP-адресов из сетей:
 - 192.168.10.0/24 - **vlan100**
 - 192.168.30.0/24 - **vlan300**
 - 192.168.40.0/24 - **vlan400**

- 192.168.50.0/24 - vlan500
- Т.к. в дальнейшем будет реализована маршрутизация между **rtr-cod** и **fw-cod**
- диапазон 192.168.20.0/24 - vlan200 указывать не надо, т.к. по условиям задания
 - "Для «cod» трафик VLAN - DATA не должен маршрутизироваться"

```
rtr-cod(config)#ip nat pool fw-cod 172.16.1.1-172.16.1.2
rtr-cod(config)#ip nat pool vlan100 192.168.10.1-192.168.10.254
rtr-cod(config)#ip nat pool vlan300 192.168.30.1-192.168.30.254
rtr-cod(config)#ip nat pool vlan400 192.168.40.1-192.168.40.254
rtr-cod(config)#ip nat pool vlan500 192.168.50.1-192.168.50.254
rtr-cod(config)#

```

- Создать правило трансляции адресов для каждого созданного **nat pool**, через интерфейс, который с точки зрения NAT **outside**:

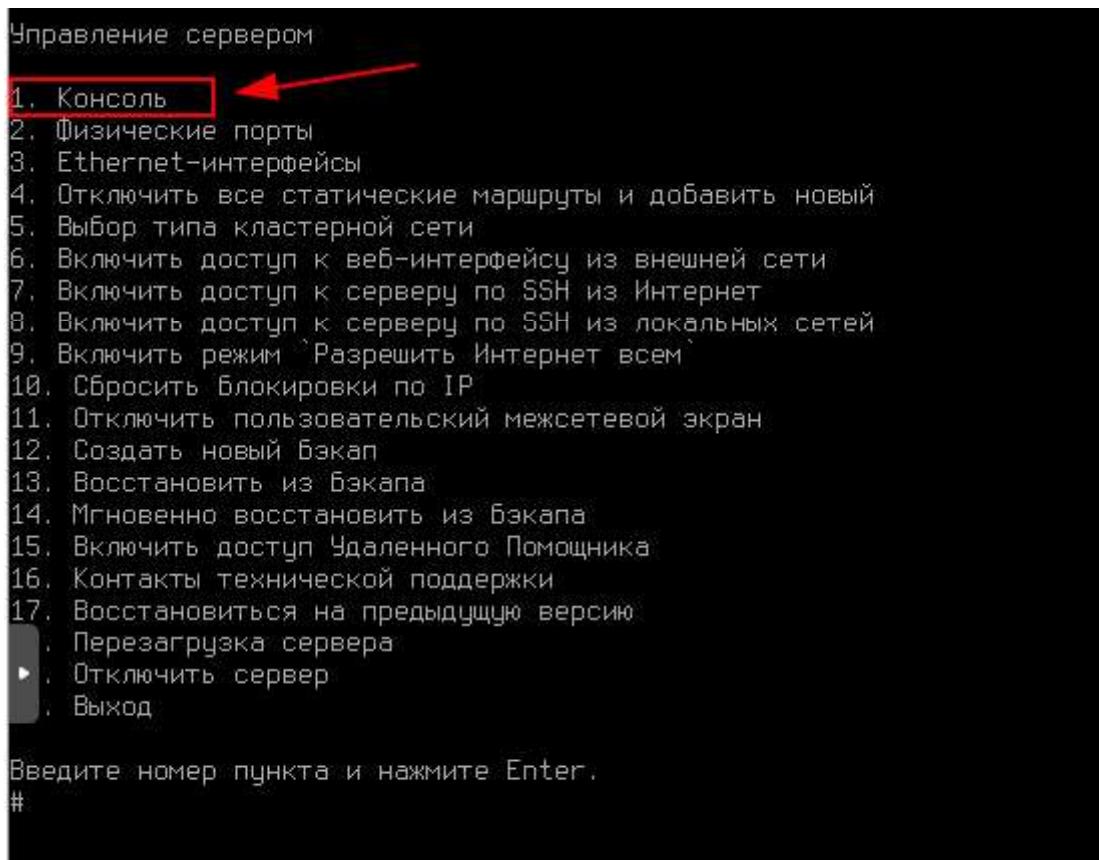
```
rtr-cod(config)#ip nat source dynamic inside pool fw-cod overload interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan100 overload interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan300 overload interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan400 overload interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan500 overload interface isp
rtr-cod(config)#write memory
Building configuration...

```

```
rtr-cod(config)#

```

- Проверить доступ в сеть Интернет с виртуальной машины **fw-cod** используя консоль:



- временно назначаем адрес шлюза по умолчанию, используя команду ip route add:

```
ip route add 0.0.0.0/0 via 172.16.1.1
```

- Проверяем доступ в сеть Интернет:

```
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=50 time=88.8 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=50 time=88.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=50 time=86.9 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 86.943/88.072/88.770/0.805 ms
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]#
```

- На rtr-cod проверяем таблицу трансляции адресов командой show ip nat translations из режима администрирования (enable):

```
rtr-cod#show ip nat translations
Static translations:
Source Translated VRF

Destination Translated VRF

Empty list.
Total: 0

PAT translations:
Source Translated Destination
Time: 9s, Protocol: UDP, VRF: default
IN: 172.16.1.2:21260 178.207.179.4:1024 216.239.32.105:53
OUT: 216.239.32.105:53 172.16.1.2:21260 178.207.179.4:1024

Time: 9s, Protocol: UDP, VRF: default
IN: 172.16.1.2:15566 178.207.179.4:1025 84.201.185.208:53
OUT: 84.201.185.208:53 172.16.1.2:15566 178.207.179.4:1025

Time: 1s, Protocol: ICMP, VRF: default
IN: 172.16.1.2 178.207.179.4 77.88.8.8
OUT: 77.88.8.8 172.16.1.2 178.207.179.4
```

- Так же на данном этапе стоит добавить статические маршруты в локальные сети COD-a:
 - динамическую маршрутизацию, например OSPF между rtr-cod и fw-cod по условиям задания реализовать нельзя, т.к.
 - на rtr-cod в дальнейшем для обеспечения динамической маршрутизации между офисом «а» и «cod»
 - "Все интерфейсы, кроме туннельных, должны быть переведены в пассивный режим"
 - так же не стоит добавлять маршрут в сеть 192.168.20.0/24 (vlan200), т.к.
 - "Для «cod» трафик VLAN - DATA не должен маршрутизоваться"

```
rtr-cod(config)#ip route 192.168.10.0/24 172.16.1.2
rtr-cod(config)#ip route 192.168.30.0/24 172.16.1.2
rtr-cod(config)#ip route 192.168.40.0/24 172.16.1.2
rtr-cod(config)#ip route 192.168.50.0/24 172.16.1.2
rtr-cod(config)#

```

- Проверить таблицу маршрутизации можно командой **show ip route** из режима администрирования (**enable**):

```

rtr-cod#show ip route
Codes: C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default

IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.1 to network 0.0.0.0

B*      0.0.0.0/0 [20/0] via 178.207.179.1, isp, 00:03:17
C       10.10.10.0/30 is directly connected, tunnel.0
C       172.16.1.0/30 is directly connected, fw-cod
C       178.207.179.0/29 is directly connected, isp
S       192.168.10.0/24 [1/0] via 172.16.1.2, fw-cod
S       192.168.30.0/24 [1/0] via 172.16.1.2, fw-cod
S       192.168.40.0/24 [1/0] via 172.16.1.2, fw-cod
S       192.168.50.0/24 [1/0] via 172.16.1.2, fw-cod
rtr-cod#

```

rtr-a (ecorouter):

Настройка dynamic PAT:

- Реализация аналогично rtr-cod, за исключением:
 - интерфейсы с точки зрения NAT должны быть:

```

rtr-a#show interface isp | grep NAT
  NAT: outside
rtr-a#show interface vl100 | grep NAT
  NAT: inside
rtr-a#show interface vl200 | grep NAT
  NAT: inside
rtr-a#show interface vl300 | grep NAT
  NAT: inside
rtr-a#

```

- **nat pool** должны быть:

```

rtr-a#show run | grep pool
ip nat pool vlan100 172.20.10.1-172.20.10.254
ip nat pool vlan200 172.20.20.1-172.20.20.254
ip nat pool vlan300 172.20.30.1-172.20.30.254
rtr-a#

```

- созданные правила трансляции:

```

rtr-a#show run | grep overload
ip nat source dynamic inside-to-outside pool vlan100 overload interface isp
ip nat source dynamic inside-to-outside pool vlan200 overload interface isp
ip nat source dynamic inside-to-outside pool vlan300 overload interface isp
rtr-a#

```

- Доступ в сеть Интернет можно проверить с виртуальной машины **sw1**-
а используя команды временного назначения IP-адресов и шлюза:

- назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **rtr-a (ens19)**,
- создав тегированный подинтерфейс с IP-адресом из подсети для **vlan300**

```
ip link add link ens19 name ens19.300 type vlan id 300
```

```
ip link set dev ens19.300 up
```

```
ip addr add 172.20.30.1/24 dev ens19.300
```

```
ip route add 0.0.0.0/0 via 172.20.30.254
```

- Доступ в сеть Интернет с **sw-1**:

```
[root@localhost ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=50 time=38.6 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=50 time=38.6 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=50 time=39.7 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 38.598/38.981/39.732/0.530 ms
[root@localhost ~]# -
```

- На **rtr-a** проверяем таблицу трансляции адресов командой **show ip nat translations** из режима администрирования (**enable**):

PAT translations:

Source	Translated	Destination
Time: 29s, Protocol: ICMP, VRF: default		
IN: 172.20.30.1	178.207.179.28	77.88.8.8
OUT: 77.88.8.8	172.20.30.1	178.207.179.28
Time: 35s, Protocol: ICMP, VRF: default		
IN: 172.20.30.1	178.207.179.28	77.88.8.8
OUT: 77.88.8.8	172.20.30.1	178.207.179.28

Вариант реализации:

rtr-cod (ecorouter):

Настройка динамической маршрутизации OSPF:

- Перейдите в режим конфигурирования протокола с помощью команды **router ospf <номер процесса>**, где номер в пределах <0-65535> в режиме глобальной конфигурации:

```
rtr-cod(config)#router ospf 1
```

```
rtr-cod(config-router)#
```

- Сконфигурируйте OSPF идентификатор маршрутизатора (необязательный этап)
 - Используйте команду **ospf router-id <значение>**, в качестве значения укажем туннельный IP-адрес маршрутизатора:

```
rtr-cod(config-router)#router-id 10.10.10.1
```

```
rtr-cod(config-router)#
```

- Переводим все интерфейсы в пассивный режим:

```
rtr-cod(config-router)#passive-interface default
```

```
rtr-cod(config-router)#
```

- Исключаем интерфейс **tunnel.0** из пассивного режима для установления соседства и дальнейшего обмена маршрутной информацией:

```
rtr-cod(config-router)#no passive-interface tunnel.0
```

```
rtr-cod(config-router)#
```

- Объявляем сети, которые будут задействованы в процессе маршрутизации:
 - импортировав (**redistribute**) ранее указанные статические маршруты (до локальных сетей COD-а) в процесс OSPF

```
rtr-cod(config-router)#redistribute static
```

```
rtr-cod(config-router)#network 10.10.10.0/30 area 0
```

```
rtr-cod(config-router)#network 172.16.1.0/30 area 0
```

```
rtr-cod(config-router)#exit
```

```
rtr-cod(config)#
```

- Обеспечиваем защиту протокола маршрутизации посредством парольной защиты:

```
rtr-cod(config)#interface tunnel.0
rtr-cod(config-if-tunnel)#ip ospf authentication message-digest
rtr-cod(config-if-tunnel)#ip ospf message-digest-key 1 md5 P@ssw0rd
rtr-cod(config-if-tunnel)#exit
rtr-cod(config)#write memory
Building configuration...
```

```
rtr-cod(config)#[/]
```

- Просмотр данных о состоянии и сконфигурированных настройках на интерфейсах, участвующих в OSPF процессе можно командой **show ip ospf interface** из режима администрирования (**enable**):

```
rtr-cod#show ip ospf interface
tunnel.0 is up, line protocol is up
  Internet Address 10.10.10.1/30, Area 0.0.0.0, MTU 1476
  Process ID 1, VRF (default), Router ID 10.10.10.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1, TE Metric 1
  Designated Router (ID) 10.10.10.1, Interface Address 10.10.10.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:10
  Neighbor Count is 0, Adjacent neighbor count is 0
  Crypt Sequence Number is 2764
  Hello received 0 sent 17, DD received 0 sent 0
  LS-Req received 0 sent 0, LS-Upd received 0 sent 0
  LS-Ack received 0 sent 0, Discarded 0
  Message-digest authentication, using key-id 1
rtr-cod#[/]
```

rtr-a (ecorouter):

Настройка динамической маршрутизации OSPF:

- Реализация аналогично **rtr-cod**, за исключением:
 - должны быть указаны следующие сети:

```
rtr-a#show running-config ospf
!
!
router ospf 1
  ospf router-id 10.10.10.2
  passive-interface default
  no passive-interface tunnel.0
  network 10.10.10.0/30 area 0.0.0.0
  network 172.20.10.0/24 area 0.0.0.0
  network 172.20.20.0/24 area 0.0.0.0
  network 172.20.30.0/24 area 0.0.0.0

  interface tunnel.0
    ip ospf authentication message-digest
    ip ospf message-digest-key 1 md5 0xeb5442f9b26607db
!
rtr-a#
```

- Просмотр сведений о соседских отношениях между OSPF маршрутизаторами можно командой **show ip ospf neighbor** из режима администрирования (enable):

```
rtr-a#show ip ospf neighbor
Total number of full neighbors: 1
OSPF process 1 VRF(default):
Neighbor ID      Pri   State          Dead Time     Address           Interface       Instance ID
10.10.10.1        1   Full/Backup   00:00:34     10.10.10.1     tunnel.0          0
rtr-a#
```

- Проверить таблицу маршрутизации можно командой **show ip route** из режима администрирования (enable):

```
rtr-a#show ip route
Codes: C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default

IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.25 to network 0.0.0.0

S*      0.0.0.0/0 [1/0] via 178.207.179.25, isp
C      10.10.10.0/30 is directly connected, tunnel.0
O      172.16.1.0/30 [110/2] via 10.10.10.1, tunnel.0, 00:00:13
C      172.20.10.0/24 is directly connected, v100
C      172.20.20.0/24 is directly connected, v1200
C      172.20.30.0/24 is directly connected, v1300
C      178.207.179.24/29 is directly connected, isp
O E2    192.168.10.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:13
O E2    192.168.30.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:13
O E2    192.168.40.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:13
O E2    192.168.50.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:13
rtr-a#
```

- Аналогично на rtr-cod:

```
rtr-cod#show ip ospf neighbor
Total number of full neighbors: 1
OSPF process 1 VRF(default):
Neighbor ID      Pri  State            Dead Time    Address          Interface          Instance ID
10.10.10.2        1   Full/DR          00:00:37    10.10.10.2    tunnel.0           0
rtr-cod#show ip route
Codes: C - connected, S - static, R - RIP, B - BGP
      0 - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default
IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.1 to network 0.0.0.0

B*   0.0.0.0/0 [20/0] via 178.207.179.1, isp, 00:04:57
C     10.10.10.0/30 is directly connected, tunnel.0
C     172.16.1.0/30 is directly connected, fw-cod
O     172.20.10.0/24 [110/2] via 10.10.10.2, tunnel.0, 00:00:43
O     172.20.20.0/24 [110/2] via 10.10.10.2, tunnel.0, 00:03:22
O     172.20.30.0/24 [110/2] via 10.10.10.2, tunnel.0, 00:03:22
C     178.207.179.0/29 is directly connected, isp
S     192.168.10.0/24 [1/0] via 172.16.1.2, fw-cod
S     192.168.30.0/24 [1/0] via 172.16.1.2, fw-cod
S     192.168.40.0/24 [1/0] via 172.16.1.2, fw-cod
S     192.168.50.0/24 [1/0] via 172.16.1.2, fw-cod
rtr-cod#
```

- Также можно проверить связность между **sw1-a** и **fw-cod**:
 - о если ранее указанный временный маршрут по умолчанию на **fw-cod** всё ещё присутствует,
 - о на **sw1-a** ранее указанный временный IP-адрес присутствует

```
[root@localhost ~]# ping -c3 172.16.1.2
PING 172.16.1.2 (172.16.1.2) 56(84) bytes of data.
64 bytes from 172.16.1.2: icmp_seq=1 ttl=62 time=130 ms
64 bytes from 172.16.1.2: icmp_seq=2 ttl=62 time=130 ms
64 bytes from 172.16.1.2: icmp_seq=3 ttl=62 time=130 ms

--- 172.16.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 129.508/129.920/130.158/0.292 ms
[root@localhost ~]# tracepath -n 172.16.1.2
??: [LOCALHOST]                                pmtu 1500
1: 172.20.30.254                               72.228ms
1: 172.20.30.254                               21.991ms
2: 172.20.30.254                               64.922ms pmtu 1476
2: 10.10.10.1                                  97.804ms
3: 172.16.1.2                                    99.717ms reached
Resume: pmtu 1476 hops 3 back 3
[root@localhost ~]#
```

Вариант реализации:

sw1-a (alt-server):

Назначение имени на устройство:

- Для назначения имени устройства согласно топологии используем следующую команду:

```
hostnamectl set-hostname sw1-a.office.ssa2026.region; exec bash
```

- Так же рекомендуется указать имя в файле **/etc/sysconfig/network**:

```
vim /etc/sysconfig/network
```

- указать имя в параметре **HOSTNAME**:

```
# When set to no, this may cause most daemons' initscripts skip starting.
NETWORKING=yes

# Used by hotplug/pcmcia/ifplugd scripts to detect current network config
# subsystem.
CONFMETHOD=etcnet

# Used by rc.susinit to setup system hostname at boot.
HOSTNAME=sw1-a.office.ssa2026.region

# This is used by ALTLinux ppp-common to decide if we want to install
# nameserver lines into /etc/resolv.conf or not.
RESOLV_MODS=yes
```

- Проверить можно с помощью команды **hostname** с ключём **-f**:

```
[root@sw1-a ~]# hostname -f
sw1-a.office.ssa2026.region
[root@sw1-a ~]#
```

Установка пакета Open vSwitch:

- Для установки пакета **openvswitch** необходим доступ в сеть Интернет, для этого необходимо на виртуальной машине **sw1-a**:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **rtr-a (ens19)**,
 - тегированный подинтерфейс с IP-адресом из подсети для **vlan300**, а также шлюзом по умолчанию и DNS

```
ip link add link ens19 name ens19.300 type vlan id 300
```

```
ip link set dev ens19.300 up
```

```
ip addr add 172.20.30.1/24 dev ens19.300
```

```
ip route add 0.0.0.0/0 via 172.20.30.254
```

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- После чего обновляем список пакетов и устанавливаем **openvswitch**:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку **openvswitch**:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g" /etc/net/ifaces/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные в ручную через пакет **iproute2**:

```
reboot
```

- Проверяем интерфейсы и определяемся какой к кому направлен:
 - таким образом, имеем:
 - **ens19** - интерфейс в сторону **rtr-a**;
 - **ens20** - интерфейс в сторону **dc-a**;
 - **ens21** - интерфейс в сторону **sw2-a**;
 - **ens22** - интерфейс в сторону **sw2-a**;

```
[root@localhost ~]# ip -c -br a
lo      UNKNOWN
ens19    UP
ens20    DOWN
ens21    DOWN
ens22    DOWN
[root@localhost ~]#
```

- Поднимаем физические интерфейсы, создавая директорию для каждого интерфейса в **/etc/net/ifaces** и описывая файл **options**:
 - если в файле **options** для порты **ens19** есть параметры **TYPE=eth** и **BOOTPROTO=static**,
 - то можно рекурсивно выполнить копирование **ens19** во все необходимые каталоги

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens20cp -r /etc/net/ifaces/ens19
```

```
/etc/net/ifaces/ens21cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens22
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить, что интерфейсы перешли из статуса **DOWN** в статус **UP**:

```
[root@sw1-a ~]# ip -c -br a
lo          UNKNOWN
ens19        UP
ens20        UP
ens21        UP
ens22        UP
[root@sw1-a ~]#
```

- Чтобы на **sw2-a** была возможность установить пакет **openvswitch**:
 - временно сосдадим простой коммутатор с именем, например **br0**
 - и добавим в него интерфейсы **ens19** в сторону **rtr-a** и **ens21** в сторону **sw2-a**

```
ovs-vsctl add-br br0ovs-vsctl add-port br0 ens19ovs-vsctl add-port br0 ens21
```

sw2-a (alt-server):

Назначение имени на устройство:

- Реализация аналогично **sw1-a**:

```
[root@sw2-a ~]# hostname -f
sw2-a.office.ssa2026.region
[root@sw2-a ~]#
```

- Для установки пакета **openvswitch** необходим доступ в сеть Интернет, для этого необходимо на виртуальной машине **sw2-a**:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **sw1-a (ens19)**,
 - тегированный подинтерфейс с IP-адресом из подсети для **vlan300**, а также шлюзом по умолчанию и DNS

```
ip link add link ens19 name ens19.300 type vlan id 300
```

```
ip link set dev ens19.300 up
```

```
ip addr add 172.20.30.2/24 dev ens19.300
```

```
ip route add 0.0.0.0/0 via 172.20.30.254
```

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- После чего обновляем список пакетов и устанавливаем **openvswitch**:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку **openvswitch**:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g" /etc/net/ifaces/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные в ручную через пакет **iproute2**:

```
reboot
```

sw1-a (alt-server):

Настройка коммутации:

- Удаляем ранее созданный временный коммутатор с именем **br0**:

```
ovs-vsctl del-br br0
```

- Создадим коммутатор с именем **sw1-a**:

```
ovs-vsctl add-br sw1-a
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
(root@sw1-a ~) # ovs-vsctl show
245b4b82-d2c3-4e24-a664-0df54cf70786
    Bridge sw1-a
        Port sw1-a
            Interface sw1-a
                type: internal
                ovs_version: "3.3.2"
(root@sw1-a ~) #
```

- Добавим интерфейс, направленный в сторону **dc-a** (ens20) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 100**:

```
ovs-vsctl add-port sw1-a ens20 tag=100
```

- Интерфейс в сторону **rtr-a** (ens19) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200 и 300)

```
ovs-vsctl add-port sw1-a ens19 trunk=100,200,300
```

- Интерфейсы в стороны **sw2-a** (ens21, ens22) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:

- также разрешаем пропуск только требуемых VLAN (100,200 и 300)

```
ovs-vsctl add-port sw1-a ens21 trunk=100,200,300
ovs-vsctl add-port sw1-a ens22
trunk=100,200,300
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-a ~]# ovs-vsctl show
245b4b82-d2c3-4e24-a664-0df54cf70786
  Bridge sw1-a
    Port ens19
      trunks: [100, 200, 300]
      Interface ens19
    Port sw1-a
      Interface sw1-a
        type: internal
    Port ens21
      trunks: [100, 200, 300]
      Interface ens21
    Port ens22
      trunks: [100, 200, 300]
      Interface ens22
    Port ens20
      tag: 100
      Interface ens20
  ovs_version: "3.3.2"
[root@sw1-a ~]#
```

- Включаем модуль ядра отвечающий за тегированный трафик (802.1Q):

```
modprobe 8021q
echo "8021q" | tee -a /etc/modules
```

- Запускаем процесс STP на коммутаторе:
 - в режиме **802.1w** (rstp)

```
ovs-vsctl set bridge sw1-a rstp_enable=true
ovs-vsctl set bridge sw1-a other_config:stp-protocol=rstp
```

- Задаём данному коммутатору наименьший приоритет:

```
ovs-vsctl set bridge sw1-a other_config:rstp-priority=0
```

- Проверить заданный приоритет можно с помощью команды **ovs-appctl rstp/show**:

```
[root@sw1-a ~]# ovs-appctl rstp/show
---- sw1-a ----
Root ID:
  stp-priority      0
  stp-system-id    bc:24:11:11:4c:4a
  stp-hello-time   2s
  stp-max-age     20s
  stp-fwd-delay   15s
  This bridge is the root

Bridge ID:
  stp-priority      0
  stp-system-id    bc:24:11:11:4c:4a
  stp-hello-time   2s
  stp-max-age     20s
  stp-fwd-delay   15s

Interface  Role      State       Cost      Pri.Nbr
-----  -----
ens19      Designated Forwarding 2000      128.4
ens20      Designated Forwarding 2000      128.1
ens22      Designated Forwarding 2000      128.2
ens21      Designated Forwarding 2000      128.3
```

- Проверить заданный протокол и приоритет можно с помощью команды **ovs-vsctl list bridge**:

```
[root@sw1-a ~]# ovs-vsctl list bridge
_uuid                  : caa77584-1e9b-4fe9-8b62-f7c383f45010
auto_attach            : []
controller            : []
datapath_id           : "0000bc2411114c4a"
datapath_type          : ""
datapath_version       : "<unknown>"
external_ids           : {}
fail_mode              : []
flood_vlans            : []
flow_tables            : {}
ipfix                  : []
mcast_snooping_enable: false
mirrors                : []
name                   : sw1-a
netflow                 : []
other_config            : {rstp-priority="0", stp-priority="0",
ports                 : [6d31f883-3097-49af-9c46-13e6a1f5e7ed,
-619310b3d52b, e0675f57-022e-4143-87c6-d06816585df0]
protocols              : []
rstp_enable             : true
rstp_status             : {rstp_bridge_id="0.000.bc2411114c4a",
rstp_root_id="0.000.bc2411114c4a", rstp_root_path_cost="0"}
sflow                  : []
status                 : {}
stp_enable              : false
```

- Сетевая подсистема **etcnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления

- создаём каталог для management интерфейса с именем **mgmt**:

```
mkdir /etc/net/ifaces/mgmt
```

- Описываем файл **options** для создания management интерфейса с именем **mgmt**:

```
vim /etc/net/ifaces/mgmt/options
```

- где:

- **TYPE** - тип интерфейса (**internal**);
- **BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
- **CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
- **BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
- **VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw1-a
VID=300
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "172.20.30.1/24" > /etc/net/ifaces/mgmt/ipv4address
echo "default via 172.20.30.254" >
/etc/net/ifaces/mgmt/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```
[root@sw1-a ~]# ip -c -br -4 a
lo      UNKNOWN    127.0.0.1/8
mgmt    UNKNOWN    172.20.30.1/24
[root@sw1-a ~]# ip -c r
default via 172.20.30.254 dev mgmt
```

- Проверить назначенный IP-адрес шлюза по умолчанию можно командой **ip r**:

```
[root@sw1-a ~]# ip -c r
default via 172.20.30.254 dev mgmt
172.20.30.0/24 dev mgmt proto kernel scope link src 172.20.30.1
[root@sw1-a ~]#
```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw1-a**:

```
[root@sw1-a ~]# ovs-vsctl show  
245b4b82-d2c3-4e24-a664-0df54cf70786  
    Bridge sw1-a  
        Port ens19  
            trunks: [100, 200, 300]  
            Interface ens19  
        Port sw1-a  
            Interface sw1-a  
                type: internal  
        Port ens21  
            trunks: [100, 200, 300]  
            Interface ens21  
        Port mgmt  
            tag: 300  
            Interface mgmt  
                type: internal  
        Port ens22  
            trunks: [100, 200, 300]  
            Interface ens22  
        Port ens20  
            tag: 100  
            Interface ens20  
    ovs_version: "3.3.2"  
[root@sw1-a ~]# -
```

- Помимо того, что интерфейс **mgmt** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt**:

```
[root@sw1-a ~]# ovs-vsctl list port mgmt  
_uuid : d44ab8d5-5565-4894-8510-64bf7bc85bad  
bond_active_slave : []  
bond_downdelay : 0  
bond_fake_iface : false  
bond_mode : []  
bond_updelay : 0  
culans : []  
external_ids : {}  
fake_bridge : false  
interfaces : [f8e44b2f-9e4d-410d-8d80-38de806f9df1]  
lacp : []  
mac : []  
name : mgmt  
other_config : {}  
protected : false  
qos : []  
rstp_statistics : {}  
rstp_status : {}  
statistics : {}  
status : {}  
tag : 300  
trunks : []  
vlan_mode : native-untagged  
[root@sw1-a ~]#
```

sw2-a (alt-server):

- Проверяем интерфейсы и определяемся какой к кому направлен:
 - таким образом, имеем:
 - **ens19** - интерфейс в сторону **sw-1**;
 - **ens20** - интерфейс в сторону **sw-1**;
 - **ens21** - интерфейс в сторону **cli2-a**;
 - **ens22** - интерфейс в сторону **cli1-a**;

```
[root@sw2-a ~]# ip -c -br a
lo          UNKNOWN
ens19        UP
ens20        DOWN
ens21        DOWN
ens22        DOWN
[root@sw2-a ~]#
```

- Поднимаем физические интерфейсы, создавая директорию для каждого интерфейса в **/etc/net/ifaces** и описывая файл **options**:
 - если в файле **options** для порты **ens19** есть параметры **TYPE=eth** и **BOOTPROTO=static**,
 - то можно рекурсивно выполнить копирование **ens19** во все необходимые каталоги

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens20cp -r /etc/net/ifaces/ens19
/etc/net/ifaces/ens21cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens22
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить, что интерфейсы перешли из статуса **DOWN** в статус **UP**:

```
[root@sw2-a ~]# ip -c -br a
lo          UNKNOWN
ens19        UP
ens20        UP
ens21        UP
ens22        UP
[root@sw2-a ~]#
```

Настройка коммутации:

- Создадим коммутатор с именем **sw2-a**:

```
ovs-vsctl add-br sw2-a
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-a ~]# ovs-vsctl show  
3c07bdb7-8007-4fa1-b4dd-3df4faee806e  
    Bridge sw2-a  
        Port sw2-a  
            Interface sw2-a  
                type: internal  
        ovs_version: "3.3.2"  
[root@sw2-a ~]# _
```

- Добавим интерфейс, направленный в сторону **cli1-a** (ens22) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw2-a ens22 tag=200
```

- Добавим интерфейс, направленный в сторону **cli2-a** (ens21) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw2-a ens21 tag=200
```

- Интерфейсы в стороны **sw1-a** (ens19, ens20) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200 и 300)

```
ovs-vsctl add-port sw1-a ens19 trunk=100,200,300ovs-vsctl add-port sw1-a ens20
```

```
trunk=100,200,300
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-a ~]# ovs-vsctl show  
3c07bdb7-8007-4fa1-b4dd-3df4faee806e  
    Bridge sw2-a  
        Port ens22  
            tag: 200  
            Interface ens22  
        Port ens20  
            trunks: [100, 200, 300]  
            Interface ens20  
        Port sw2-a  
            Interface sw2-a  
                type: internal  
        Port ens21  
            tag: 200  
            Interface ens21  
        Port ens19  
            trunks: [100, 200, 300]  
            Interface ens19  
        ovs_version: "3.3.2"  
[root@sw2-a ~]# _
```

- Включаем модуль ядра отвечающий за тегированный трафик (**802.1Q**):

```
modprobe 8021qecho "8021q" | tee -a /etc/modules
```

- Запускаем процесс STP на коммутаторе:
 - в режиме **802.1w** (rstp)

```
ovs-vsctl set bridge sw2-a rstp_enable=true
ovs-vsctl set bridge sw2-a other_config:stp-protocol=rstp
```

- Проверить можно с помощью команды **ovs-appctl rstp/show**:

```
[root@sw2-a ~]# ovs-appctl rstp/show
---- sw2-a ----
Root ID:
  stp-priority    0
  stp-system-id   bc:24:11:11:4c:4a
  stp-hello-time  2s
  stp-max-age     20s
  stp-fwd-delay   15s
  root-port       ens20
  root-path-cost  2000

Bridge ID:
  stp-priority    32768
  stp-system-id   bc:24:11:5d:92:6d
  stp-hello-time  2s
  stp-max-age     20s
  stp-fwd-delay   15s

  Interface  Role      State        Cost      Pri.Nbr
  _____
  ens21      Designated Forwarding 2000      128.4
  ▶ ens20      Root        Forwarding 2000      128.1
  ens22      Designated Forwarding 2000      128.2
  ens19      Alternate   Discarding 2000      128.3

[root@sw2-a ~]#
```

- Проверить заданный протокол и приоритет можно с помощью команды **ovs-vsctl list bridge**:

```
[root@sw2-a ~]# ovs-vsctl list bridge
_uuid                  : aec1d564-ade1-461e-97f8-2e6635225138
_auto_attach           : []
_controller            : []
_datapath_id           : "0000bc24115d926d"
_datapath_type          :
_datapath_version       : "<unknown>"
_external_ids          : {}
_fail_mode              : []
_flood_vlans           : []
_flow_tables            : {}
_ipfix                  :
_mcast_snooping_enable: false
_mirrors                :
_name                   : sw2-a
_netflow                 :
_other_config            : {stp-protocol=rstp}
_ports                  : [5eb84cec-c0fe-42ef-90d4-67ec06e20403, 61b6
-001cc189acf3, eed4acda-c2b4-4828-85ed-68c38e27cec8]
_protocols               :
_rstp_enable             : true
_rstp_status              : {rstp_bridge_id="8.000.bc24115d926d", rstp_
/rstp_root_id="0.000.bc2411114c4a", rstp_root_path_cost="2000"}
_sfflow                  :
_status                  : {}
_stp_enable               : false
[root@sw2-a ~]#
```

- Сетевая подсистема **etcnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления
 - создаём каталог для management интерфейса с именем **mgmt**:

`mkdir /etc/net/ifaces/mgmt`

- Описываем файл **options** для создания management интерфейса с именем **mgmt**:

`vim /etc/net/ifaces/mgmt/options`

- где:
 - TYPE** - тип интерфейса (**internal**);
 - BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
 - CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
 - BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
 - VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw2-a
VID=300
~
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "172.20.30.2/24" > /etc/net/iface/mgmt/ipv4address
echo "default via 172.20.30.254" >
/etc/net/iface/mgmt/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```
[root@sw2-a ~]# ip -c -br -4 a
lo          UNKNOWN      127.0.0.1/8
mgmt        UNKNOWN      172.20.30.2/24
[root@sw2-a ~]#
```

- Проверить назначенный IP-адрес шлюза по умолчанию можно команжой **ip r**:

```
[root@sw2-a ~]# ip -c r
default via 172.20.30.254 dev mgmt
172.20.30.0/24 dev mgmt proto kernel scope link src 172.20.30.2
[root@sw2-a ~]#
```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw2-a**:

```
[root@sw2-a ~]# ovs-vsctl show
3c07bdb7-8007-4fa1-b4dd-3df4faee806e
  Bridge sw2-a
    Port mgmt
      tag: 300
      Interface mgmt
        type: internal
    Port ens22
      tag: 200
      Interface ens22
    Port ens20
      trunks: [100, 200, 300]
      Interface ens20
    Port sw2-a
      Interface sw2-a
        type: internal
    Port ens21
      tag: 200
      Interface ens21
    Port ens19
      trunks: [100, 200, 300]
      Interface ens19
  ovs_version: "3.3.2"
[root@sw2-a ~]#
```

- Помимо того, что интерфейс **mgmt** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt**:

```
[root@sw2-a ~]# ovs-vsctl list port mgmt
_uuid : 3c29bbb7-8223-4892-b8a3-7a3c671eaf30
bond_active_slave : []
bond_downdelay : 0
bond_fake_iface : false
bond_mode : []
bond_updelay : 0
culans : []
external_ids : {}
fake_bridge : false
interfaces : [9150af57-a5e1-4c74-b0e1-358639b9dd3e]
lacp : []
mac : []
name : mgmt
other_config : {}
protected : false
qos : []
rstp_statistics : {}
rstp_status : {}
statistics : {}
status : {}
tag : 300
trunks : []
vlan_mode : native-untagged
[root@sw2-a ~]
```

- Проверить доступность коммутаторов **sw1-a** и **sw2-a** можно как с маршрутизатора **rtr-a**:

```
rtr-a#ping 172.20.30.1 ←
PING 172.20.30.1 (172.20.30.1) 56(84) bytes of data.
64 bytes from 172.20.30.1: icmp_seq=1 ttl=64 time=211 ms
64 bytes from 172.20.30.1: icmp_seq=2 ttl=64 time=22.4 ms
64 bytes from 172.20.30.1: icmp_seq=3 ttl=64 time=205 ms
64 bytes from 172.20.30.1: icmp_seq=4 ttl=64 time=29.1 ms

--- 172.20.30.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 22.421/116.800/210.773/91.076 ms
rtr-a#ping 172.20.30.2 ←
PING 172.20.30.2 (172.20.30.2) 56(84) bytes of data.
64 bytes from 172.20.30.2: icmp_seq=1 ttl=64 time=81.4 ms
64 bytes from 172.20.30.2: icmp_seq=2 ttl=64 time=24.0 ms
64 bytes from 172.20.30.2: icmp_seq=3 ttl=64 time=73.7 ms
64 bytes from 172.20.30.2: icmp_seq=4 ttl=64 time=198 ms

--- 172.20.30.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 24.005/94.301/198.166/63.879 ms
rtr-a#
```

- так и с маршрутизатора **rtr-cod**:

```
rtr-cod#ping 172.20.30.1 ←
PING 172.20.30.1 (172.20.30.1) 56(84) bytes of data.
64 bytes from 172.20.30.1: icmp_seq=1 ttl=64 time=121 ms
64 bytes from 172.20.30.1: icmp_seq=2 ttl=64 time=86.3 ms
64 bytes from 172.20.30.1: icmp_seq=3 ttl=64 time=109 ms
64 bytes from 172.20.30.1: icmp_seq=4 ttl=64 time=184 ms

--- 172.20.30.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 86.327/125.213/184.125/36.243 ms
rtr-cod#ping 172.20.30.2 ←
PING 172.20.30.2 (172.20.30.2) 56(84) bytes of data.
64 bytes from 172.20.30.2: icmp_seq=1 ttl=64 time=197 ms
64 bytes from 172.20.30.2: icmp_seq=2 ttl=64 time=98.9 ms
64 bytes from 172.20.30.2: icmp_seq=3 ttl=64 time=105 ms
64 bytes from 172.20.30.2: icmp_seq=4 ttl=64 time=103 ms

--- 172.20.30.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 98.929/126.174/197.339/41.150 ms
rtr-cod#
```

- Также с коммутаторов **sw1-a** и **sw2-a** должна быть доступна сеть Интернет:

```
[root@sw1-a ~]# ping -c3 77.88.8.8 ←
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=90.7 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=92.3 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=88.6 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 88.647/90.537/92.284/1.488 ms
[root@sw1-a ~]#
```

```
[root@sw2-a ~]# ping -c3 77.88.8.8 ←
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=64.9 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=61.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=60.6 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 60.613/62.352/64.948/1.870 ms
[root@sw2-a ~]#
```

Вариант реализации:

fw-cod (ideco):

Проверка наличия временного маршрута по умолчанию:

- Выполнить вход на fw-cod из-под ранее созданного пользователя и перейти в консоль:

```
Ideco NGFW 21.5.99
-----
Вход в локальное меню.

Введите логин и нажмите Enter.
# admin

Введите пароль и нажмите Enter.

Введите 'b' и нажмите Enter для возврата.
# 

Управление сервером

1. Консоль
2. Физические порты
3. Ethernet-интерфейсы
4. Отключить все статические маршруты и добавить новый
Выбор типа кластерной сети
▶ Включить доступ к веб-интерфейсу из внешней сети
▶ Включить доступ к серверу по SSH из Интернет
8. Включить доступ к серверу по SSH из локальных сетей
9. Включить режим Разрешить Интернет всем
10. Сбросить блокировки по IP
11. Отключить пользовательский межсетевой экран
12. Создать новый бэкап
13. Восстановить из бэкапа
14. Мгновенно восстановить из бэкапа
15. Включить доступ Удаленного Помощника
16. Контакты технической поддержки
17. Восстановиться на предыдущую версию
18. Перезагрузка сервера
19. Отключить сервер
20. Выход

Введите номер пункта и нажмите Enter.
# 1_
```

- Проверить таблицу маршрутов можно с помощью команды ip r:

```
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# ip -c r  
169.254.1.0/29 dev 1b_local_in proto kernel scope link src 169.254.1.1  
169.254.1.0/29 dev 1b_local_out proto kernel scope link src 169.254.1.3  
169.254.1.4/30 dev Lvpn0 proto kernel scope link src 169.254.1.6  
169.254.1.5 dev Lvpn0 scope link  
172.16.1.0/30 dev Leth2 proto kernel scope link src 172.16.1.2  
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]#
```

- Если отсутствует маршрут по умолчанию, его необходимо временно добавить средствами команды **ip route add ...:**

```
ip route add 0.0.0.0/0 via 172.16.1.1
```

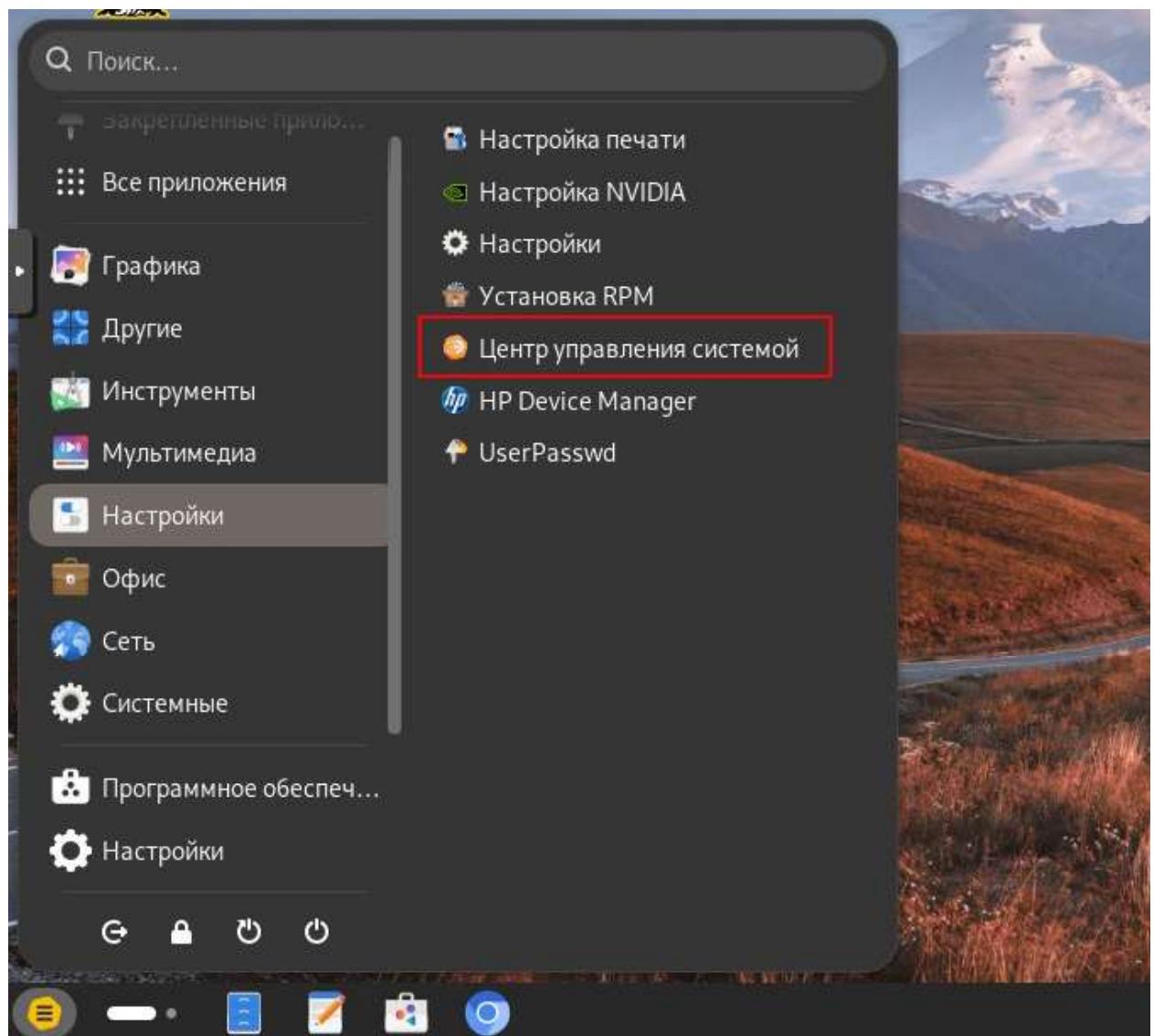
- Повторно приверить таблицу маршрутов:

```
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# ip -c r  
default via 172.16.1.1 dev Leth2  
169.254.1.0/29 dev 1b_local_in proto kernel scope link src 169.254.1.1  
169.254.1.0/29 dev 1b_local_out proto kernel scope link src 169.254.1.3  
169.254.1.4/30 dev Lvpn0 proto kernel scope link src 169.254.1.6  
169.254.1.5 dev Lvpn0 scope link  
172.16.1.0/30 dev Leth2 proto kernel scope link src 172.16.1.2  
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# _
```

cli1-a (alt-workstation):

Базовая настройка устройства:

- Перейдём в Центр Управления Системой (ЦУС/acc):



- В ЦУС перейдём в раздел Ethernet-интерфейсы:

 Обновить Переключиться на старую версию  Справка

Система

Информация об установленной системе и её настройка

[Дата и время](#) [Информация о дистрибутиве](#) [Лицензионное соглашение](#) [Настройка zram-swap](#)[Настройка нескольких рабочих мест](#) [Настройка ограничений](#) [Обновление системы](#)[Обновление ядра](#) [Сетевые каталоги](#) [Системные журналы](#) [Системные ограничения](#)[Системные службы](#) [Управление ключами SSL](#)

Пользователи

Управление пользователями системы

[Администратор системы](#) [Аутентификация](#) [Использование диска](#) [Локальные группы](#)[Локальные учётные записи](#)

Брандмауэр

Брандмауэр

[Внешние сети](#) [Перенаправление портов](#) [Список блокируемых хостов](#)

Сеть

Настройка подключения к сети

[Ethernet-интерфейсы](#) [OpenVPN-соединения](#) [PPPoE-соединения](#) [PPTP-соединения](#)[Прокси-сервер](#)

Графический интерфейс

Настройка устройств ввода-вывода

[Дисплей](#) [Клавиатура](#)

- Назначим имя на устройство в соответствие с топологией:

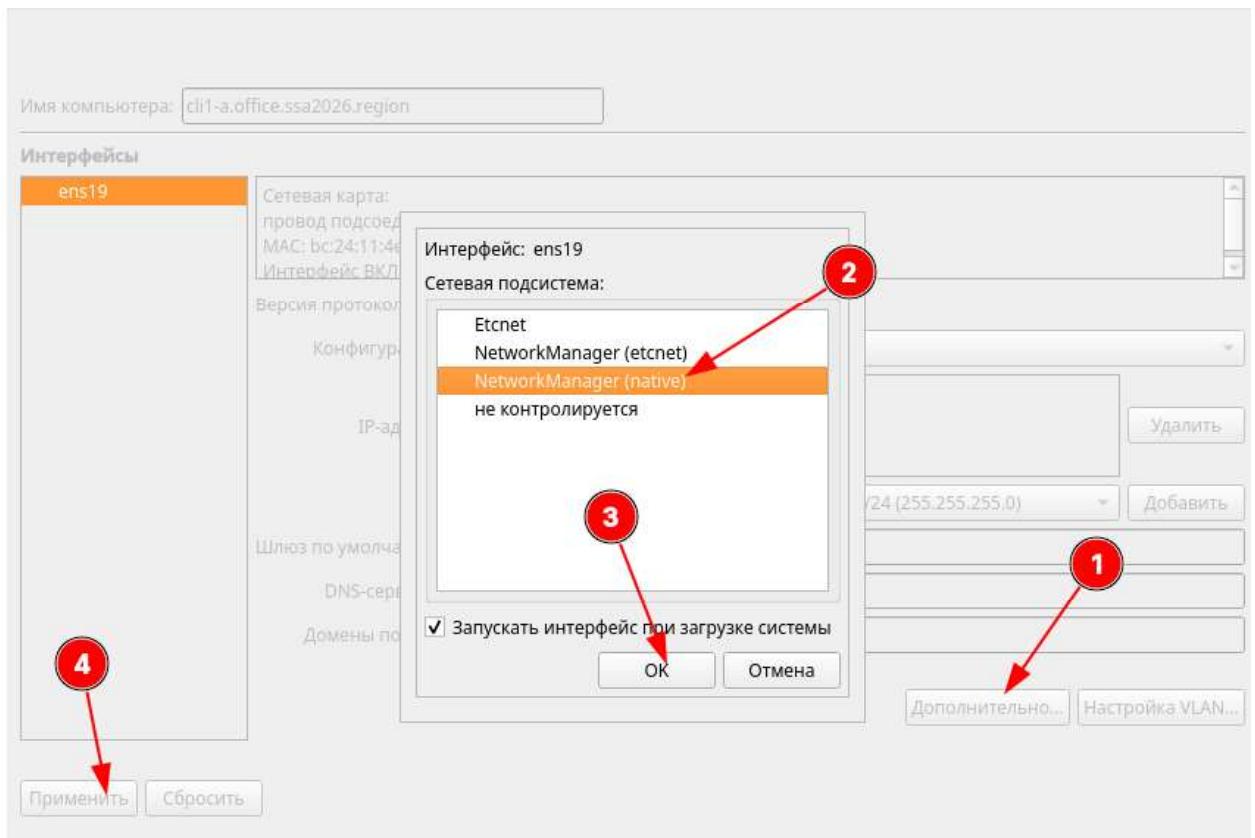
Имя компьютера:

Интерфейсы

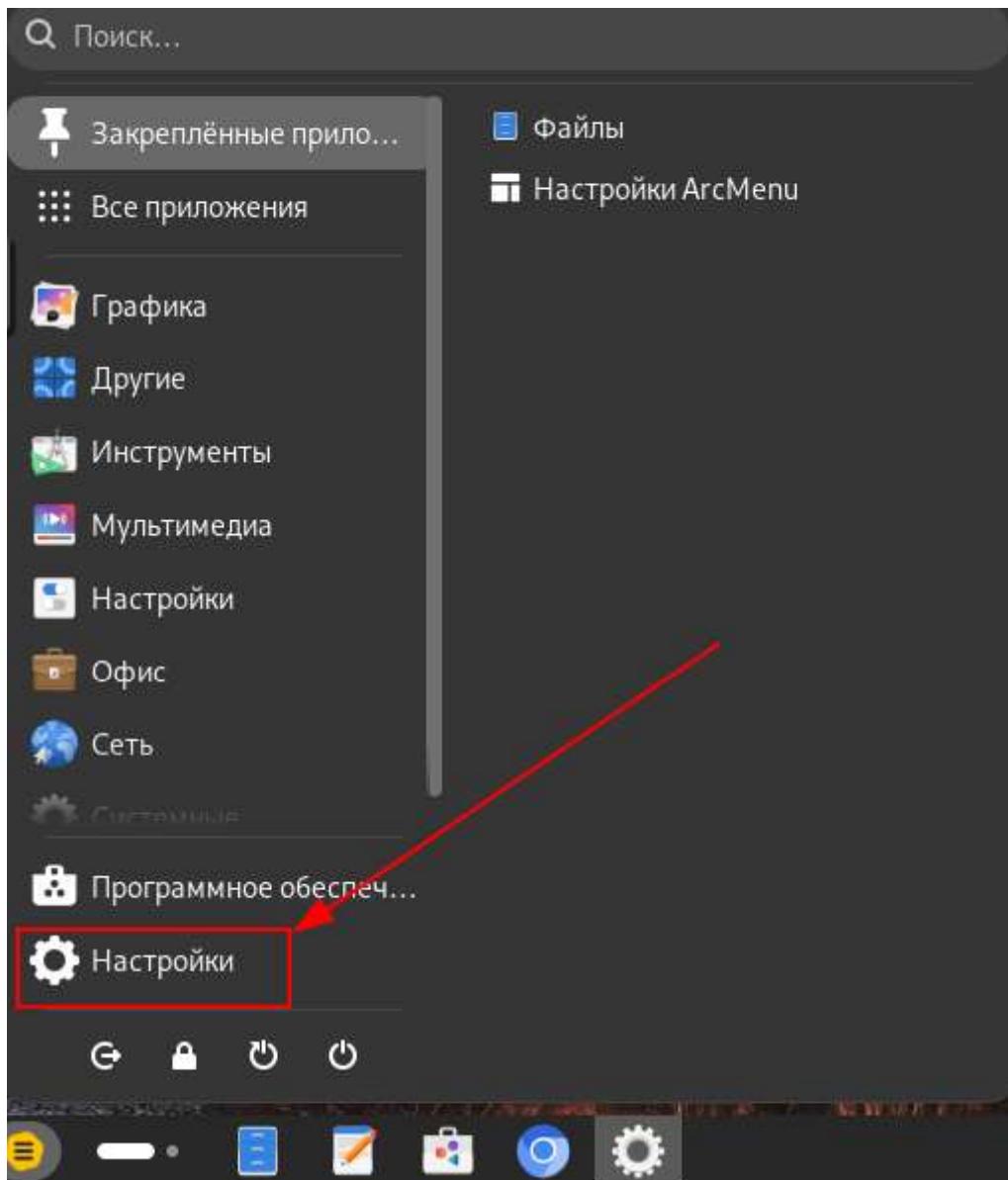
ens19	Сетевая карта: провод подсоединен MAC: bc:24:11:4e:3c:26 Интерфейс ВКЛЮЧЕН
	Версия протокола IP: <input type="button" value="IPv4"/> <input checked="" type="checkbox"/> Включить
	Конфигурация: <input type="button" value="Использовать DHCP"/>
	IP-адреса:
	Добавить IP: <input type="text"/>
	Шлюз по умолчанию: <input type="text"/>
	DNS-серверы: <input type="text"/>
	Домены поиска: <input type="text"/>
	(несколько значений записываются через пр)



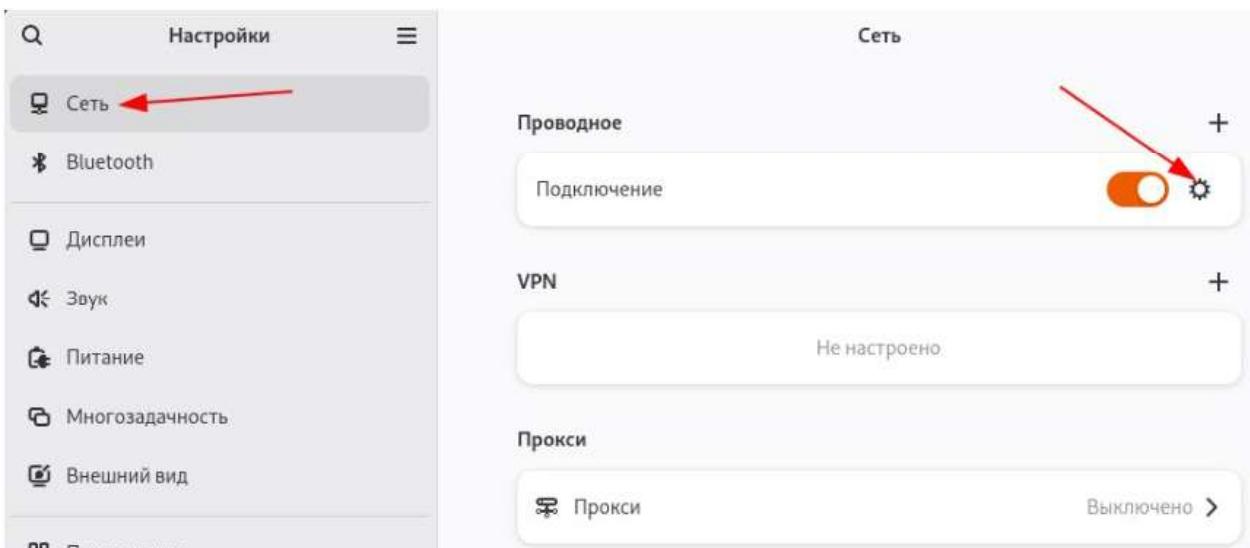
- В качестве режима работы сетевой подсистемы выберем **NetworkManager (native)**:



- Перейдём в обычный **Настройки** для назначения сетевых параметров:



- В разделе **Сеть** перейдём к настройке сетевого подключения:



- Задаём **Вручную** и нажимаем **Применить**:

- Адрес
- Маску сети
- Шлюз
- DNS (в качестве DNS-сервера указываем IP-адрес dc-a)

Отменить **Проводное подключение** **Применить**

Подробности	Идентификация	IPv4	IPv6	Безопасность
-------------	---------------	------	------	--------------

Метод IPv4

<input type="radio"/> Автоматический (DHCP)	<input checked="" type="radio"/> Вручную	<input type="radio"/> Только для локальной сети
<input type="radio"/> Общий доступ другим компьютерам		<input type="radio"/> Выключить

Адреса

Адрес	Маска сети	Шлюз
172.20.20.1	255.255.255.0	172.20.20.254
		<input type="button" value="x"/>

DNS

172.20.10.10	<input checked="" type="checkbox"/> Автоматически
--------------	---

- Проверить можно на вкладке Подробности:

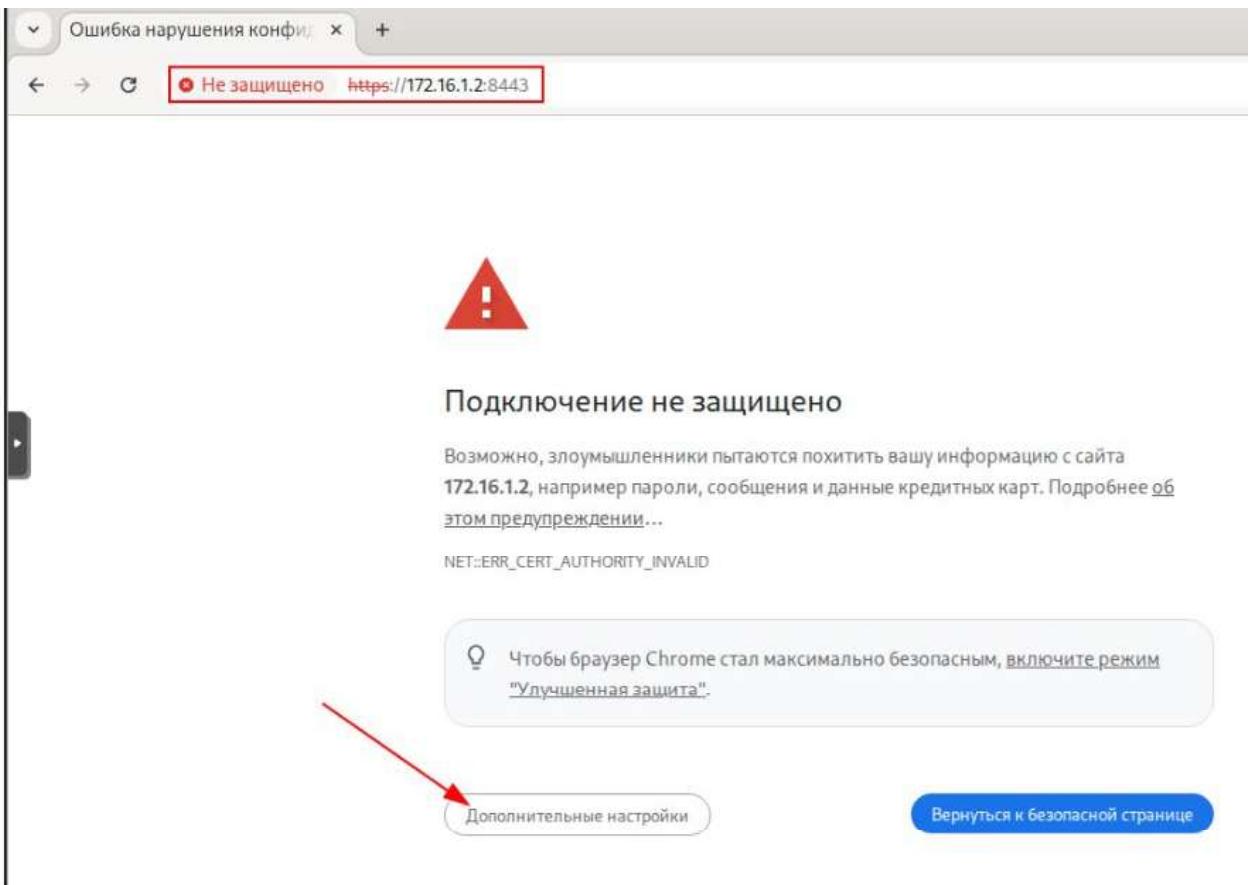
Отменить **Проводное подключение** **Применить**

Подробности	Идентификация	IPv4	IPv6	Безопасность
-------------	---------------	------	------	--------------

Адрес IPv4 172.20.20.1
 Адрес IPv6 fe80::9eb7:43db:b272:14f3
 Аппаратный адрес BC:24:11:4E:3C:26
 Маршрут по умолчанию 172.20.20.254
 DNS 172.20.10.10

Доступ к веб-интерфейсу управления fw-cod:

- Открываем браузер и переходим по <https://172.16.1.2:8443> (IP-адрес fw-cod):



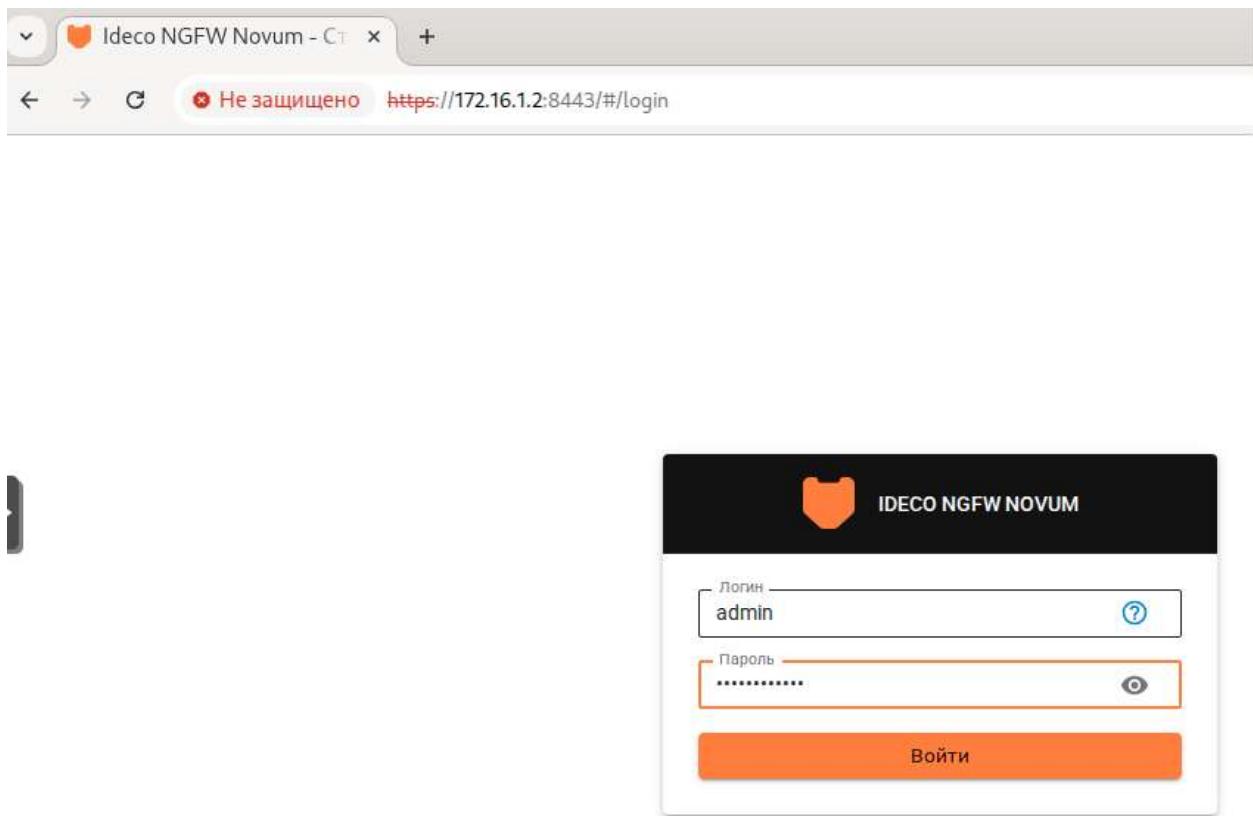
[Скрыть подробности](#)

[Вернуться к безопасной странице](#)

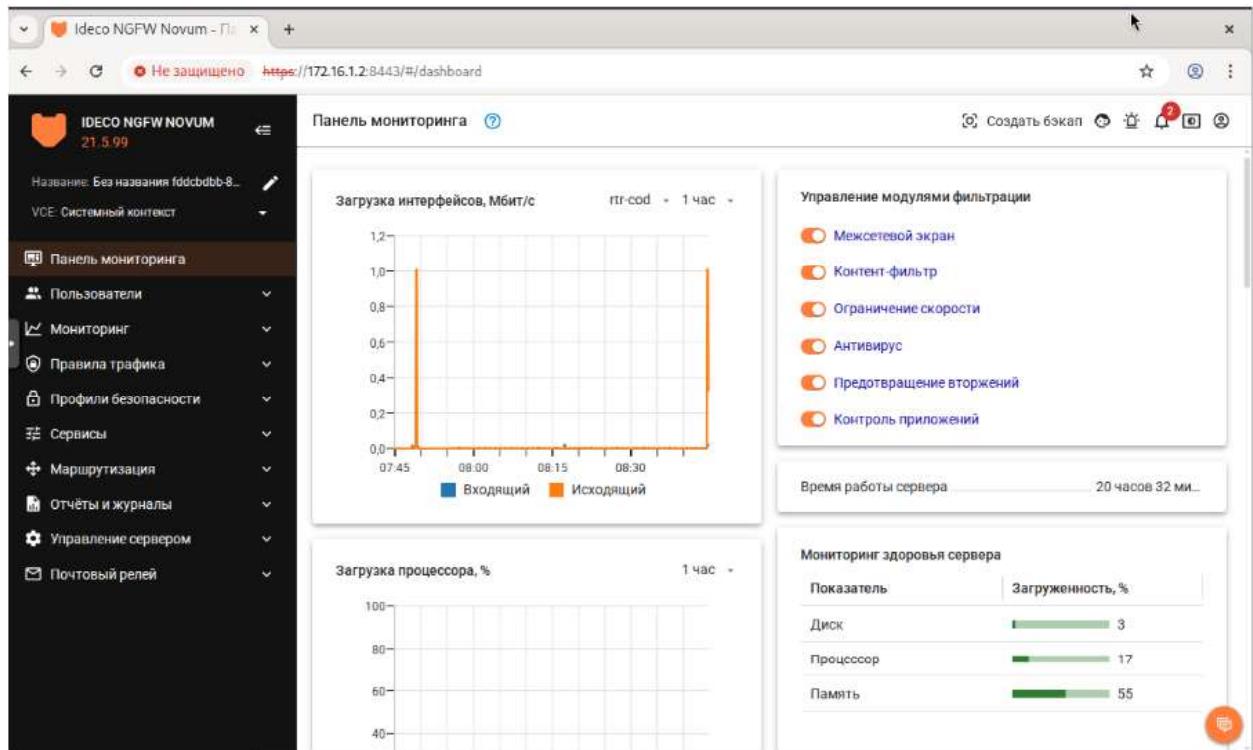
Не удалось подтвердить, что это сервер **172.16.1.2**. Операционная система компьютера не доверяет его сертификату безопасности. Возможно, сервер настроен неправильно или кто-то пытается перехватить ваши данные.

[Перейти на сайт 172.16.1.2 \(небезопасно\)](#)

- Выполняем вход в веб-интерфейс управления fw-cod:



- Результат успешного входа:



Базовая настройка fw-cod через веб-интерфейс управления:

- Зададим имя устройству в соответствие с топологией:

The screenshot shows the Ideco NGFW Novum dashboard. On the left, there's a sidebar with the device logo, name 'IDECO NGFW NOVUM 21.5.99', and a red box highlighting the 'Название: fw-cod.cod.ssa2026.region' field. A red arrow points from this box to the top right where the title 'Панель мониторинга' is located. Below it, a chart titled 'Загрузка интерфейсов, N' shows a single data point at approximately 1.0.

- Зададим адрес маршрута по умолчанию на постоянной основе, т.к. для первого доступа задавался временный:
 - для начала создадим **Объект** с типом **IP-адрес**, где и укажем значение, которое в дальнейшем будет использовано в качестве **шлюза**:

The screenshot shows the 'Объекты' (Objects) page. At the top, there are buttons for '+ Добавить' (Add), 'Фильтры' (Filters), and 'Отображение' (Display). A red arrow points from the '+ Добавить' button to the 'Добавить' button in the sidebar. Another red arrow points from the 'Добавить' button to the '3' value in the 'Порты' row. In the sidebar, the 'Мониторинг' section is expanded, with the 'Объекты' item highlighted by a red circle labeled '2'. A red arrow points from this circle to the 'Добавить' button on the main page.

Добавление объекта

Тип IP-адрес

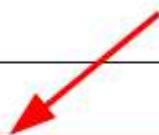
Название gateway

Значение 172.16.1.1

Комментарий

0/256

Добавить **Отмена**

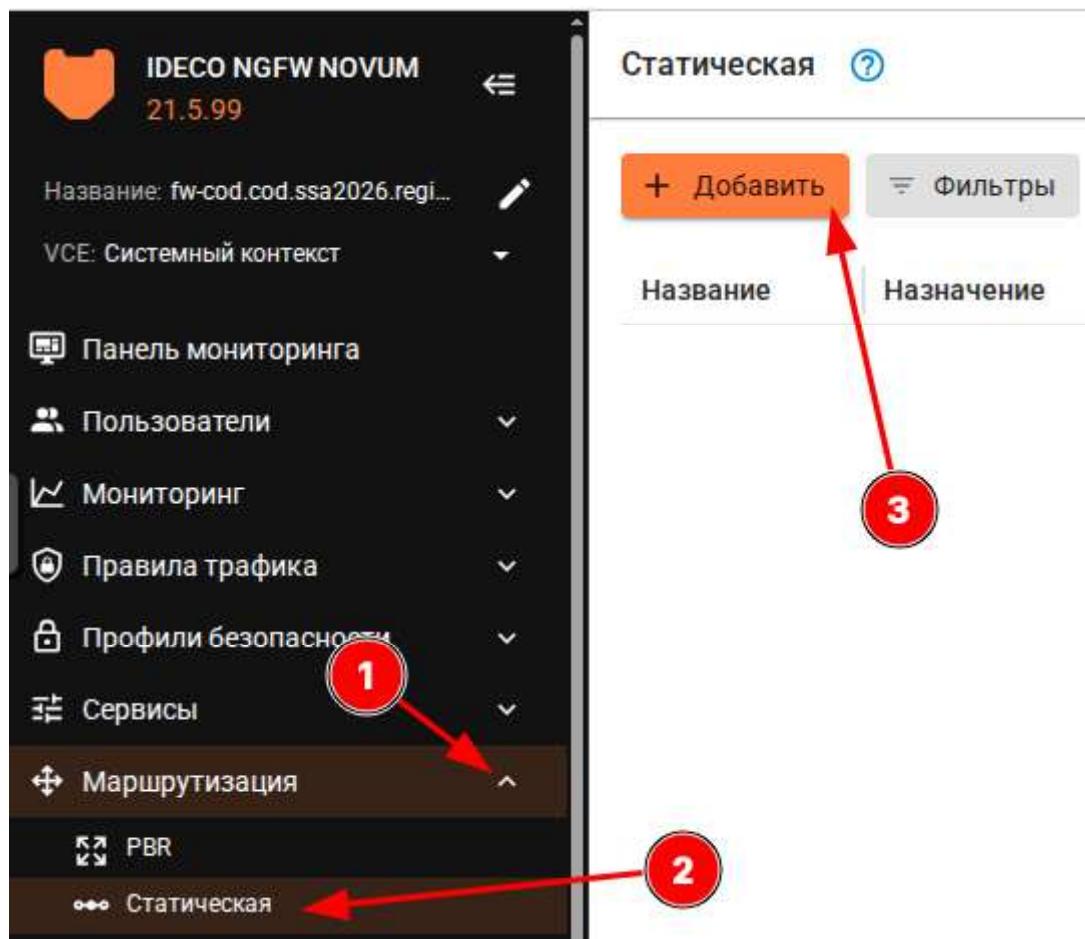


- Результат успешного добавления **Объекта** с типом **IP-адрес**:

Объекты ?

+ Добавить	Фильтры	Отображение	Поиск
Название		Значение	
IP-адрес · 1			
gateway		172.16.1.1	
Порты · 6			
Порт · 14			
Время · 1			

- после чего можем создать статический маршрут:



- Заполняем форму, указав:
 - произвольное имя в поле **Название**;
 - выбрав в качестве **Назначения** - **Любой**;
 - выбрав в качестве **Интерфейса** - единственный (на текущий момент) созданный интерфейс в сторону **rtr-cod**;
 - выбрав в поле **Шлюз** - ранее созданный объект с именем **gateway** (в значение этого объекта - IP-адрес шлюза по умолчанию, т.е. **rtr-cod**)

Статическая ?

Добавление маршрута

Название Поле необязательное

Назначение X ▼

интерфейс X ▼

Шлюз X ▼

Поле необязательное. Укажите шлюз, если выбран интерфейс с режимом статической IP-конфигурации.

Приоритет

Ведите число от 1 до 255. Чем меньше приоритет, тем приоритетнее маршрут

Дополнительно

Комментарий

0/256

Добавить Отмена

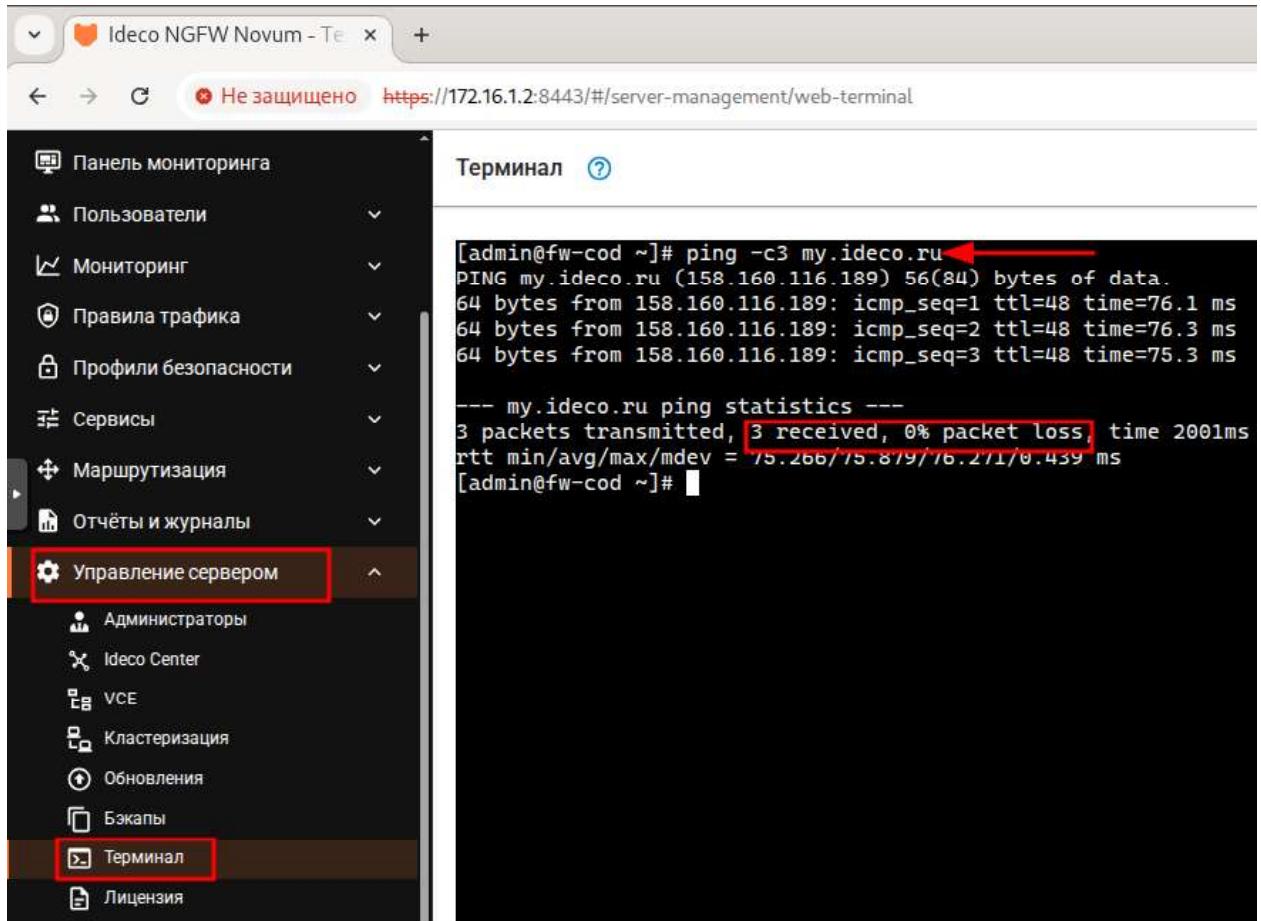


- Результат успешного добавления статического маршрута:

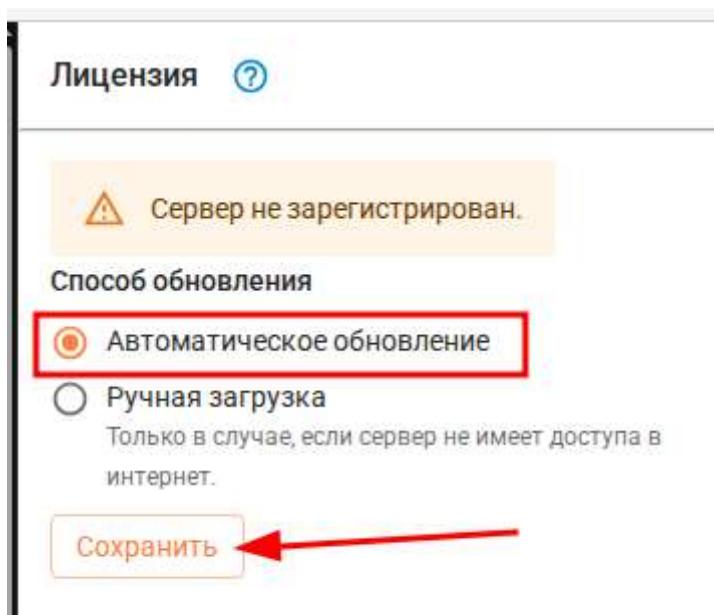
Статическая ?							Создать бэкап ? ? ? ? ?	
Добавить Фильтры Отображение Поиск								
Название	Назначение	Интерфейс	Шлюз	Приоритет	Статус	Комментарий	Управление	
default-gateway	* Любой	rtr-cod	172.16.1.1	1	Используется		? ? ?	

Получение лицензии на fw-cod через веб-интерфейс управления:

- Для получения лицензии у fw-cod должен быть доступ в сеть Интернет, а именно до **my.ideco.ru**:



- Перейдите в **Управление сервером** → **Лицензия**, выберите **Автоматическое обновление** в качестве способа обновления, нажмите **Сохранить**:



- Перейдите в **MY.IDECO**, нажав **Зарегистрировать**:

- Для того, чтобы имя **my.ideco.ru** с виртуальной машины **cli1-a** корректно было преобразовано DNS-сервером:



Не удается получить доступ к сайту

Не удалось найти IP-адрес сервера **my.ideco.ru**.

Попробуйте сделать следующее:

[Проверьте настройки прокси-сервера, брандмауэра и DNS.](#)

DNS_PROBE_FINISHED_BAD_CONFIG

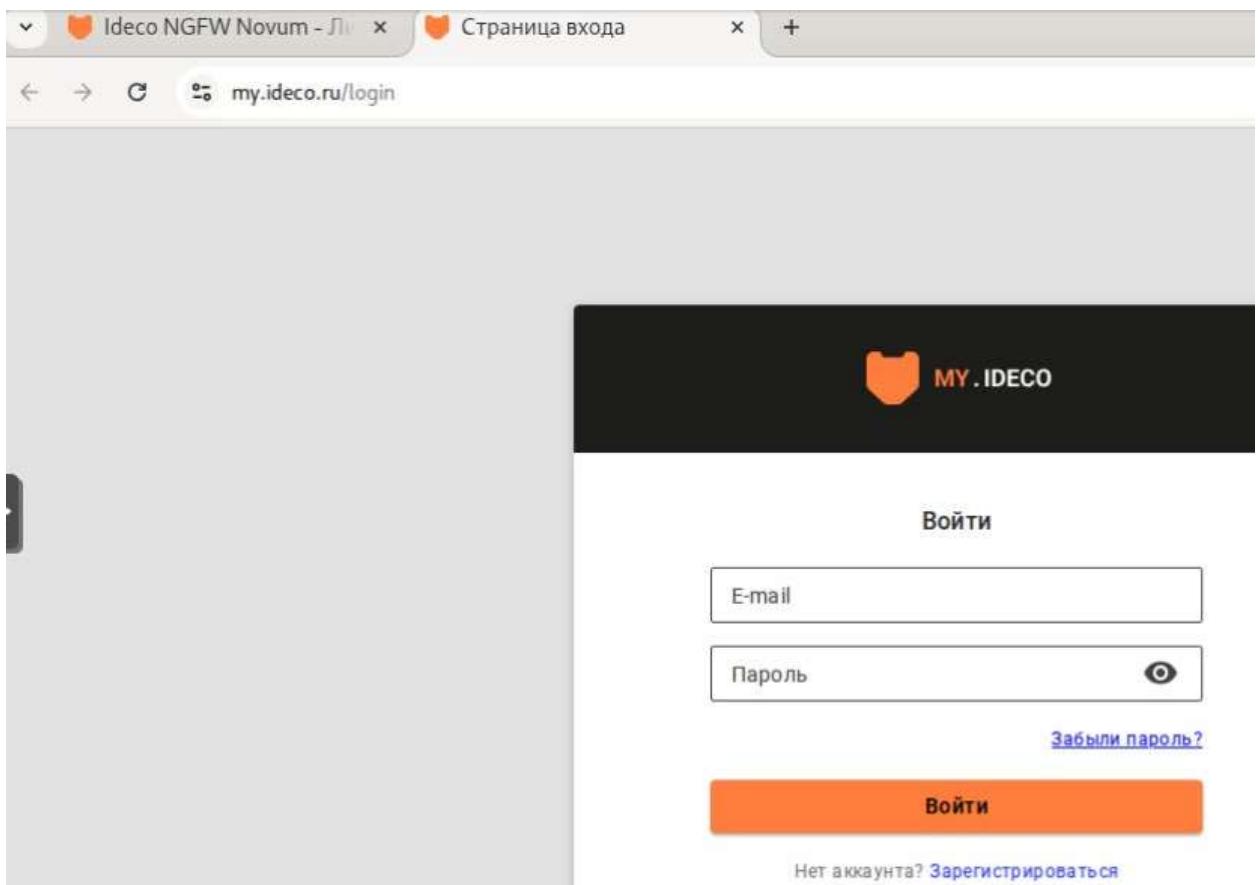
Сведения

Перезагрузить

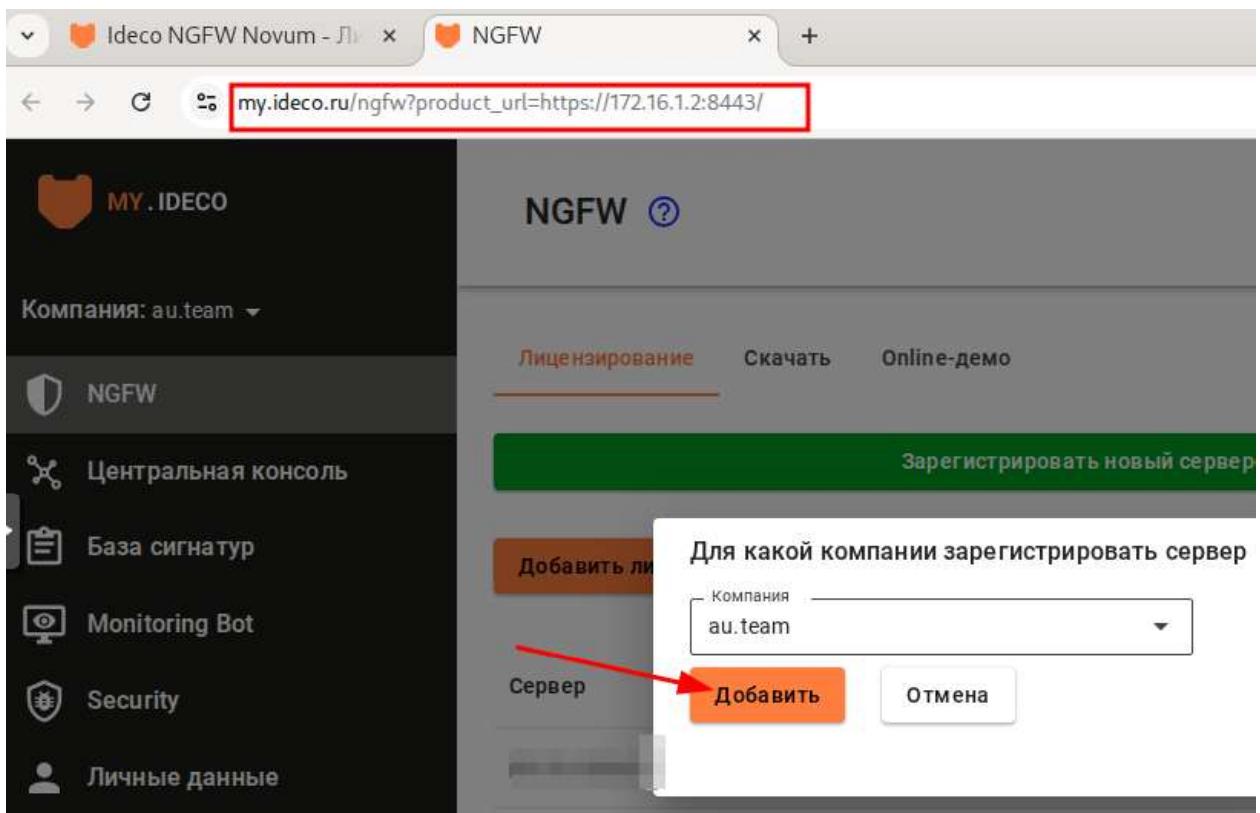
- необходимо временно назначить в качестве DNS-сервера, какой-либо публичный DNS, т.к. на **dc-a** пока DNS не реализован:

```
Q root@cli1-a: /root + ⌂ ×
[user@cli1-a ~]$ su -
Password:
[root@cli1-a ~]# echo "nameserver 77.88.8.8" > /etc/resolv.conf
[root@cli1-a ~]#
```

- Результат, выполните вход в личный кабинет MY.IDECO для дальнейшего лицензирования:



- В открывшемся окне выберите компанию и нажмите **Добавить**
 - после добавления нажмите **Обновить** информацию о лицензии для проверки состояния лицензии:



- Результат успешного лиценсирования fw-cod:

Не защищено <https://172.16.1.2:8443/#/modules/about/license>

Лицензия [?](#)

Способ обновления

Автоматическое обновление

Ручная загрузка
Только в случае, если сервер не имеет доступа в интернет.

Сохранить

Управление лицензией осуществляется в [личном кабинете](#).

Информация о лицензии:

Номер лицензии	LIC-2584284227
Тип лицензии	enterprise-demo
Начало действия лицензии	15 секунд назад, четверг, 13 ноября 2025 г., 9:37
Окончание лицензии	через 1 месяц, суббота, 27 декабря 2025 г., 9:37
Окончание обновлений	через 1 месяц, суббота, 27 декабря 2025 г., 9:37
Окончание технической поддержки	через 1 месяц, суббота, 27 декабря 2025 г., 9:37
Количество пользователей	0 из 10 000
Название компании	au.team
Название сервера	unnamed

Вариант реализации:

cli1-a (alt-workstation):

Создание интерфейсов типа VLAN на fw-cod для маршрутизации между VLAN

- Открываем браузер и переходим по <https://172.16.1.2:8443> (IP-адрес fw-cod), выполняем вход в веб-интерфейс управления fw-cod
- Перейдёт в Сервисы -> Сетевые интерфейсы -> Добавить -> VLAN:

The screenshot shows the IDECO NGFW Novum web interface. On the left, there's a sidebar with various service icons and names like 'Панель мониторинга', 'Пользователи', 'Мониторинг', 'Правила трафика', 'Профили безопасности', 'Сервисы' (which is highlighted with a red box), and 'Сетевые интерфейсы'. The main right panel is titled 'Сетевые интерфейсы' and has tabs for 'ИНТЕРФЕЙСЫ' (selected) and 'ФИЗИЧЕСКИЕ'. Below these tabs is an orange 'Добавить' button. A list of interface types is shown: Ethernet (highlighted with a red box), VLAN, LACP, SPAN, GRE, L2TP, PPPoE, and PPTP. The status for each interface type is listed as 'Настройка' (Configuration).

- Заполняем форму **Добавление Ethernet-интерфейса**:
 - задаём произвольное имя;
 - роль **LAN**;
 - физический порт, направленный в сторону **fw1-cod**;
 - режим - **Без конфигурации**, т.к. IP-адреса будут назначаться на интерфейсы **VLAN**, созданные на базе данного интерфейса

Добавление Ethernet-интерфейса

Название fw1-cod

Настройки

Роль LAN

Зона

Поле необязательное

VCE Системный контекст

Виртуальный контекст (VCE), в котором будет использоваться интерфейс

Физический порт eth1_da

IP-конфигурация

Режим Без конфигурации

- Нажимаем Добавить, результат добавления интерфейса:

Сетевые интерфейсы

ИНТЕРФЕЙСЫ ФИЗИЧЕСКИЕ ПОРТЫ

Тип	Статус	Название	Роль	Зона	VCE	Интерфейс/порт	Ter VL...	IP-адр...	Комм...	Управление
Ethern...	■■■■■	rtr-cod	LAN	—	● Сис...	eth0_01	—	172.1...		⊕ ⚒ ━━
Ethern...	■■■■■	fw1-cod	LAN	—	● Сис...	eth1_da	—	—		⊕ ⚒ ━━

- Создаём интерфейсы типа VLAN:

IDECO NGFW NOVUM
21.5.99

Название: fw-cod.cod.ssa2026.reg...
VCE: Системный контекст

Панель мониторинга
Пользователи
Мониторинг
Правила трафика
Профили безопасности
Сервисы
Сетевые интерфейсы
IGMP Proxy

Сетевые интерфейсы

ИНТЕРФЕЙСЫ ФИЗИЧЕСКИЕ ПОРТЫ

+ Добавить Фильтры

	Статус	Настройки	Назначение	Роль
Ethernet	OK			LAN
VLAN	OK		rtr-cod	LAN
LACP	OK			
SPAN	OK			
GRE				
L2TP				
PPPoE				
PPTP				

- Заполняем форму **Добавление VLAN-интерфейса**:
 - задаём произвольное имя;
 - выбираем ранее созданный интерфейс в сторону **sw1-cod**;
 - указываем в качестве тега **vlan 100**;
 - задаём статическую конфигурацию IP-адреса и нажимаем добавить

Добавление VLAN-интерфейса

Название
srv-cod

Настройки

Роль
LAN

Зона

Поле необязательное

VCE

Системный контекст

Виртуальный контекст (VCE), в котором будет
использоваться интерфейс

Интерфейс
fw1-cod

На выбранном интерфейсе создаётся VLAN

Tag VLAN
100

1094

IP-конфигурация

Режим	Статический
IP-адрес/маска	192.168.10.254/24

+ Добавить IP-адрес с маской

- ⓘ Для доступа к сети интернет настройте Балансировку и резервирование.

Дополнительно

Индекс интерфейса для Netflow	0
-------------------------------	---

Целое число от 0 до 65 535

Комментарий	0/256
-------------	-------

Добавить

Отмена

- Результат добавления интерфейса типа VLAN:

Тип	Статус	Название	Интерфейс/порт	Ter VLAN	IP-адрес/маска	Управление
Ethernet	OK	rtr-cod	eth0_01	-	172.16.1.2/30	
Ethernet	OK	fw1-cod	eth1_da	-	-	
VLAN	OK	srv-cod	fw1-cod	100	192.168.10.254/24	

- Аналогичным образом необходимо создать интерфейсы типа VLAN для:
 - **vlan300**
 - **vlan400**
 - **vlan500**
 - для **vlan200** создавать интерфейс нет необходимости, т.к. по условиям задания трафик данного vlan-а не должен маршрутизоваться

Тип	Статус	Название	Интерфейс/порт	Ter VLAN	IP-адрес/маска	Управление
Ethernet	зеленый	rtr-cod	eth0_01	—	172.16.1.2/30	
Ethernet	зеленый	fw1-cod	eth1_da	—	—	
VLAN	зеленый	srv-cod	fw1-cod	100	192.168.10.254/24	
VLAN	зеленый	mgmt-cod	fw1-cod	300	192.168.30.254/24	
VLAN	зеленый	cli	fw1-cod	400	192.168.40.254/24	
VLAN	зеленый	voip	fw1-cod	500	192.168.50.254/24	

- Доступ в сеть Интернет можно проверить с виртуальной машины **sw1-cod** используя команды временного назначения IP-адресов и шлюза:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **fw-cod (ens19)**,
 - создав тегированный подинтерфейс с IP-адресом из подсети для **vlan300**

```
ip link add link ens19 name ens19.300 type vlan id 300
```

```
ip link set dev ens19 up
```

```
ip link set dev ens19.300 up
```

```
ip addr add 192.168.30.1/24 dev ens19.300
```

```
ip route add 0.0.0.0/0 via 192.168.30.254
```

- Доступ в сеть Интернет с **sw1-cod**:

```
[root@localhost ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=92.1 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=89.3 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=91.4 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 89.266/90.936/92.147/1.220 ms
[root@localhost ~]#
```

Вариант реализации:

sw1-cod (alt-server):

Назначение имени на устройство:

- Для назначения имени устройства согласно топологии используем следующую команду:

```
hostnamectl set-hostname sw1-cod.cod.ssa2026.region; exec bash
```

- Так же рекомендуется указать имя в файле **/etc/sysconfig/network**:

```
vim /etc/sysconfig/network
```

- указать имя в параметре **HOSTNAME**:

```
# When set to no, this may cause most daemons' initscripts skip starting.
NETWORKING=yes

# Used by hotplug/pcmcia/ifplugd scripts to detect current network config
# subsystem.
CONFMETHOD=ethtool

# Used by rc.susinit to setup system hostname at boot.
HOSTNAME=sw1-cod.cod.ssa2026.region

# This is used by ALTLinux ppp-common to decide if we want to install
# nameserver lines into /etc/resolv.conf or not.
RESOLV_MODS=yes
```

- Проверить можно с помощью команды **hostname** с ключом **-f**:

```
[root@sw1-cod ~]# hostname -f
sw1-cod.cod.ssa2026.region
[root@sw1-cod ~]# -
```

cli1-a (alt-workstation):

Настройка авторизации на fw-cod для доступа в сеть Интернет:

- Авторизация - необходимое условие для доступа пользователя в интернет. Для работы в пределах локальной сети авторизация не требуется
- Для доступа сетевого устройства (хоста) в интернет через NGFW Novum с возможностью контроля трафика, хост должен быть авторизован в системе под учетной записью пользователя
- Открываем браузер и переходим по <https://172.16.1.2:8443> (IP-адрес fw-cod), выполняем вход в веб-интерфейс управления fw-cod

- Для авторизации необходимо создать пользователя, для этого перейдём в Пользователи -> Учётные записи и нажмём Добавить пользователя:

The screenshot shows the 'User Accounts' section of the Ideco NGFW Novum web interface. On the left, there's a sidebar with links: 'Monitoring Panel', 'Users' (which is highlighted with a red box), 'User Accounts' (also highlighted with a red box), and 'External Catalogs'. The main panel has tabs for 'LOCAL' and 'ACTIVE DIRECTORY'. Below the tabs is a search bar with 'Search' and a magnifying glass icon. Under the 'LOCAL' tab, there's a section titled 'All local' with a '+' button and a red arrow pointing to it. Another entry 'Ideco Device VPN' is listed below.

- В форме **Добавления пользователя** заполняем только **Имя пользователя** (произвольное) и **Логин** (произвольный)
 - пароль можно оставить сгенерированный случайным образом, он нам не потребуется

Добавить пользователя в группу «Все локальные»

Основные настройки

Имя пользователя	network
Логин	network
Пароль
Повторите пароль

Рекомендуется использовать комбинацию цифр и

- Результат успешного добавления пользователя:

Учётные записи (?)

ЛОКАЛЬНЫЕ ACTIVE DIRECTORY

Поиск

- Все локальные [папка] [+]
- [закрыт] [открыто] Ideco Device VPN [+]
- [пользователь] network [+]

- Далее перейдёт в Пользователи -> Авторизация -> ПО ПОДСЕТЯД и нажмём Добавить:

IDEKO NGFW NOVUM
21.5.99

Название: fw-cod.cod.ssa2026.regi... [редактировать]

VCE: Системный контекст

Панель мониторинга

[пользователь] Пользователи

Учётные записи

Внешние каталоги

Аутентификация

[пользователь] Авторизация

VPN-подключения

Ideco Client

Авторизация (?)

НАСТРОЙКИ IP И MAC по подсетям

+ Добавить Фильтры Отображение

Подсеть Пользователь

- В форме **Добавление правила авторизации** указываем только что созданного пользователя и подсеть
 - в данном случае в сеть **192.168.0.0/16** входят все подсети **COD-a** (именно 192.168.10.0/24, 192.168.20.0/24, 192.168.30.0/24, 192.168.40.0/24, 192.168.50.0/24);

Авторизация ?

НАСТРОЙКИ IP И MAC **ПО ПОДСЕТЯМ**

Добавление правила авторизации

Пользователь: network

Подсеть: 192.168.0.0/16

Комментарий:

0/256

Добавить **Отмена**

- Результат успешного добавления Авторизации по подсетям:

Авторизация ?

? Создать бэкап OK Удалить Обновить Сохранить Сохранить как

НАСТРОЙКИ IP И MAC **ПО ПОДСЕТЯМ**

+ Добавить

Фильтры

Отображение

Поиск

Подсеть

Пользователь

Каталог

Комментарий

Управление

192.168.0.0/16

network

Локальные пользо...

Создать Изменить Удалить

Настройка коммутации:

- Проверяем интерфейсы и определяемся какой к кому направлен (сверка производится по MAC-адресам):
 - таким образом, имеем (в данном конкретном случае):
 - **ens19** - интерфейс в сторону **fw-cod**;
 - **ens21** - интерфейс в сторону **sw2-cod**;
 - **ens22** - интерфейс в сторону **sw2-cod**;
 - **enp2s1** - интерфейс в сторону **srv2-cod**;
 - **enp2s29** - интерфейс в сторону **admin-cod**;
 - **enp3s12** - интерфейс в сторону **srv2-cod**

```
[root@sw1-cod ~]# ip -c -br a
lo          UNKNOWN
ens19        DOWN
ens21        DOWN
ens22        DOWN
enp2s1       UP
enp2s29     DOWN
enp3s12      UP
[root@sw1-cod ~]# -
```

- Для каждого интерфейса необходимо создать директорию по пути `/etc/net/ifaces/` с помощью команды `mkdir`:

```
[root@sw1-cod ~]# ls /etc/net/ifaces/
default  enp2s1  enp2s29  enp3s12  ens19  ens20  ens21  ens22  lo  unknown
[root@sw1-cod ~]#
```

- Для каждого интерфейса в директории `/etc/net/ifaces/ <ИМЯ_ИНТЕРФЕЙСА>` необходимо создать файл `options`
 - указав в нём два основных параметра:
 - `TYPE=eth`
 - `BOOTPROTO=static`
 - после чего необходимо перезагрузить службу `network`
 - все интерфейсы должны перейти в статус `UP`:

```
[root@sw1-cod ~]# ip -c -br a
lo          UNKNOWN
ens19        UP
ens21        UP
ens22        UP
enp2s1       UP
enp2s29     UP
enp3s12      UP
[root@sw1-cod ~]#
```

- Временно на базе интерфейса в сторону `fw-cod` создадим подинтерфейс с указанием `vlan300` для дальнейшей установки пакета `openvswitch`:

```
ip link add link ens19 name ens19.300 type vlan id 300
```

```
ip link set dev ens19.300 up
```

```
ip addr add 192.168.30.1/24 dev ens19.300
```

```
ip route add 0.0.0.0/0 via 192.168.30.254
```

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- После чего обновляем список пакетов и устанавливаем `openvswitch`:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку `openvswitch`:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g" /etc/net/iface/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные в ручную через пакет **iproute2**:

```
reboot
```

- Чтобы на **sw2-cod** была возможность установить пакет **openvswitch**:
 - временно сосдадим простой коммутатор с именем, например **br0**
 - и добавим в него интерфейсы **ens19** в сторону **fw-cod** и **ens21** в сторону **sw2-cod**

```
ovs-vsctl add-br br0 ovs-vsctl add-port br0 ens19 ovs-vsctl add-port br0 ens21
```

sw2-cod (alt-server):

Назначение имени на устройство:

- Реализация аналогично **sw1-cod**:

```
[root@sw2-cod ~]# hostname -f
sw2-cod.cod.ssa2026.region
[root@sw2-cod ~]#
```

- Для установки пакета **openvswitch** необходим доступ в сеть Интернет, для этого необходимо на виртуальной машине **sw2-cod**:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **sw1-cod (ens19)**,
 - тегированный подинтерфейс с IP-адресом из подсети для **vlan300**, а также шлюзом по умолчанию и DNS

```
ip link add link ens19 name ens19.300 type vlan id 300
```

```
ip link set up ens19
```

```
ip link set up ens19.300
```

```
ip addr add 192.168.30.2/24 dev ens19.300
```

```
ip route add 0.0.0.0/0 via 192.168.30.254
```

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- Почле чего обновляем список пакетов и устанавливаем **openvswitch**:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку **openvswitch**:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g" /etc/net/ifaces/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные в ручную через пакет **iproute2**:

```
reboot
```

sw1-cod (alt-server):

Настройка коммутации:

- Удаляем ранее созданный временный коммутатор с именем **br0**:

```
ovs-vsctl del-br br0
```

- Создадим коммутатор с именем **sw1-cod**:

```
ovs-vsctl add-br sw1-cod
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-cod ~]# ovs-vsctl show
ca12e05a-0291-425e-8290-ac709fdaf53f
  Bridge sw1-cod
    Port sw1-cod
      Interface sw1-cod
        type: internal
        ovs_version: "3.3.2"
[root@sw1-cod ~]# -
```

- Добавим интерфейс, направленный в сторону **admin-cod** (enp2s29) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 300**:

```
ovs-vsctl add-port sw1-cod enp2s29 tag=300
```

- Добавим интерфейс, направленный в сторону **srv2-cod** (`enp2s1`) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 100**:

```
ovs-vsctl add-port sw1-cod enp2s1 tag=100
```

- Добавим интерфейс, направленный в сторону **srv2-cod** (`enp3s12`) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw1-cod enp3s12 tag=200
```

- Интерфейс в сторону **fw-cod** (`ens19`) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200,300,400 и 500)

```
ovs-vsctl add-port sw1-cod ens19 trunk=100,200,300,400,500
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-cod ~]# ovs-vsctl show
ca12e05a-0291-425e-8290-ac709fdaf53f
  Bridge sw1-cod
    Port enp3s12
      tag: 200
      Interface enp3s12
    Port enp2s1
      tag: 100
      Interface enp2s1
    Port sw1-cod
      Interface sw1-cod
        type: internal
    Port ens19
      trunks: [100, 200, 300, 400, 500]
      Interface ens19
    Port enp2s29
      tag: 300
      Interface enp2s29
  ovs_version: "3.3.2"
[root@sw1-cod ~]#
```

- Включаем модуль ядра отвечающий за тегированный трафик (802.1Q):

```
modprobe 8021q echo "8021q" | tee -a /etc/modules
```

- На базе интерфейсов **ens21** и **ens22**, направленных в сторону **sw2-cod**, создадим **bond**-интерфейс в режиме **active-backup**:

```
ovs-vsctl add-bond sw1-cod bond0 ens21 ens22 bond_mode=active-backup
```

- Интерфейс типа **bond** в сторону **sw2-cod** (**bond0**) настраиваем как магистральный (trunk) порт:

- также разрешаем пропуск только требуемых VLAN (100,200,300,400 и 500)

```
ovs-vsctl set port bond0 trunk=100,200,300,400,500
```

- Проверить можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-cod ~]# ovs-vsctl show  
ca12e05a-0291-425e-8290-ac709fdaf53f  
    Bridge sw1-cod  
        Port enp3s12  
            tag: 200  
            Interface enp3s12  
        Port enp2s1  
            tag: 100  
            Interface enp2s1  
        Port bond0  
            trunks: [100, 200, 300, 400, 500]  
            Interface ens22  
            Interface ens21  
        Port sw1-cod  
            Interface sw1-cod  
                type: internal  
        Port ens19  
            trunks: [100, 200, 300, 400, 500]  
            Interface ens19  
        Port enp2s29  
            tag: 300  
            Interface enp2s29  
    ovs_version: "3.3.2"  
[root@sw1-cod ~]
```

- Режим работы **bond** интерфейса проверить можно с помощью команды **ovs-appctl bond/show**:

```
[root@sw1-cod ~]# ovs-appctl bond/show  
---- bond0 ----  
bond_mode: active-backup  
bond may use recirculation: no, Recirc-ID : -1  
bond-hash-basis: 0  
lb_output action: disabled, bond-id: -1  
updelay: 0 ms  
downdelay: 0 ns  
lacp_status: off  
lacpFallback_ab: false  
active-backup primary: <none>  
active member mac: bc:24:11:81:8e:b2(ens22)  
  
member ens21: enabled  
    may_enable: true  
  
member ens22: enabled  
    active member  
    may_enable: true  
[root@sw1-cod ~]
```

- Сетевая подсистема **etcnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления
 - создаём каталог для management интерфейса с именем **mgmt-cod**:

```
mkdir /etc/net/ifaces/mgmt-cod
```

- Описываем файл **options** для создания management интерфейса с именем **mgmt-cod**:
 - где:
 - TYPE** - тип интерфейса (**internal**);
 - BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
 - CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
 - BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
 - VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw1-cod
VID=300
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "192.168.30.1/24" > /etc/net/ifaces/mgmt-cod/ipv4address
echo "default via 192.168.30.254" > /etc/net/ifaces/mgmt-cod/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```
[root@sw1-cod ~]# ip -c -br -4 a
lo      UNKNOWN    127.0.0.1/8
mgmt-cod UNKNOWN    192.168.30.1/24
[root@sw1-cod ~]# -
```

- Проверить назначенный IP-адрес шлюза по умолчанию можно команжой **ip r**:

```
[root@sw1-cod ~]# ip -c r
default via 192.168.30.254 dev mgmt-cod
192.168.30.0/24 dev mgmt-cod proto kernel scope link src 192.168.30.1
[root@sw1-cod ~]#
```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw1-cod**:

```
[root@sw1-cod ~]# ovs-vsctl show  
ca12e05a-0291-425e-8290-ac709fdaf53f  
  Bridge sw1-cod  
    Port enp3s12  
      tag: 200  
      Interface enp3s12  
    Port mgmt-cod  
      tag: 300  
      Interface mgmt-cod  
        type: internal  
    Port enp2s1  
      tag: 100  
      Interface enp2s1  
    Port bond0  
      trunks: [100, 200, 300, 400, 500]  
      Interface ens22  
      Interface ens21  
    Port sw1-cod  
      Interface sw1-cod  
        type: internal  
    Port ens19  
      trunks: [100, 200, 300, 400, 500]  
      Interface ens19  
    Port enp2s29  
      tag: 300  
      Interface enp2s29  
  ovs_version: "3.3.2"  
[root@sw1-cod ~]#
```

- Помимо того, что интерфейс **mgmt-cod** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt-cod vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt-cod**:

```
[root@sw1-cod ~]# ovs-vsctl list port mgmt-cod
    _uuid           : 1434ac51-88f6-4ea8-8718-15a0bbec2714
    bond_active_slave   : []
    bond_downdelay     : 0
    bond_fake_iface    : false
    bond_mode          : []
    bond_updelay       : 0
    cvlans             : []
    external_ids       : {}
    fake_bridge        : false
    interfaces         : [b865de0e-0251-4c5a-942a-14710cd69fa5]
    lacp               : []
    mac                : []
    name               : mgmt-cod
    other_config       : {}
    protected          : false
    qos                : []
    rstp_statistics    : {}
    rstp_status        : {}
    statistics         : {}
    status              : {}
    tag                : 300
    trunks             : []
    vlan_mode          : native-untagged
[root@sw1-cod ~]#
```

sw2-cod (alt-server):

- Проверяем интерфейсы и определяемся какой к кому направлен:
 - таким образом, имеем (в данном конкретном случае):
 - **ens19** - интерфейс в сторону **sw1-cod**;
 - **ens20** - интерфейс в сторону **sw1-cod**;
 - **ens21** - интерфейс в сторону **srv1-cod**;
 - **ens22** - интерфейс в сторону **srv1-cod**;
 - **enp2s29** - интерфейс в сторону **sip-cod**;
 - **enp3s12** - интерфейс в сторону **cli-cod**;

```
[root@sw2-cod ~]# ip -c -br a
lo          UNKNOWN
ens19        DOWN
ens20        DOWN
ens21        DOWN
ens22        DOWN
enp2s29      UP
enp3s12      DOWN
[root@sw2-cod ~]#
```

- Для каждого интерфейса в директории **/etc/net/iface**/
<ИМЯ_ИНТЕРФЕЙСА>/ необходимо создать файл **options**
 - указав в нём два основных параметра:
 - **TYPE=eth**
 - **BOOTPROTO=static**

- после чего необходимо перезагрузить службу **network**
- все интерфейсы должны перейти в статус **UP**:

```
[root@sw2-cod ~]# ip -c -br a
lo          UNKNOWN
ens19        UP
ens20        UP
ens21        UP
ens22        UP
enp2s29      UP
enp3s12      UP
[root@sw2-cod ~]#
```

- Создадим коммутатор с именем **sw2-cod**:

```
ovs-vsctl add-br sw2-cod
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-cod ~]# ovs-vsctl show
a83d6bb0-91b6-49ec-be50-3c19857b3947
  Bridge sw2-cod
    Port sw2-cod
      Interface sw2-cod
        type: internal
        ovs_version: "3.3.2"
[root@sw2-cod ~]#
```

- Добавим интерфейс, направленный в сторону **srv1-cod** (ens21) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 100**:

```
ovs-vsctl add-port sw2-cod ens21 tag=100
```

- Добавим интерфейс, направленный в сторону **srv1-cod** (ens22) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw2-cod ens22 tag=200
```

- Добавим интерфейс, направленный в сторону **sip-cod** (enp2s29) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 500**:

```
ovs-vsctl add-port sw2-cod enp2s29 tag=500
```

- Добавим интерфейс, направленный в сторону **cli-cod** (enp3s12) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 400**:

```
ovs-vsctl add-port sw2-cod enp3s12 tag=400
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-cod ~]# ovs-vsctl show  
a83d6bb0-91b6-49ec-be50-3c19857b3947  
  Bridge sw2-cod  
    Port ens21  
      tag: 100  
      Interface ens21  
    Port enp2s29  
      tag: 500  
      Interface enp2s29  
    Port enp3s12  
      tag: 400  
      Interface enp3s12  
    Port sw2-cod  
      Interface sw2-cod  
        type: internal  
    Port ens22  
      tag: 200  
      Interface ens22  
  ovs_version: "3.3.2"  
[root@sw2-cod ~]# -
```

- Включаем модуль ядра отвечающий за тегированный трафик (**802.1Q**):

```
modprobe 8021q echo "8021q" | tee -a /etc/modules
```

- На базе интерфейсов **ens19** и **ens20**, направленных в сторону **sw1-cod**, создадим **bond**-интерфейс в режиме **active-backup**:

```
ovs-vsctl add-bond sw2-cod bond0 ens19 ens20 bond_mode=active-backup
```

- Интерфейс типа **bond** в сторону **sw2-cod (bond0)** настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200,300,400 и 500)

```
ovs-vsctl set port bond0 trunk=100,200,300,400,500
```

- Проверить можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-cod ~]# ovs-vsctl show  
a83d6bb0-91b6-49ec-be50-3c19857b3947  
    Bridge sw2-cod  
        Port ens21  
            tag: 100  
            Interface ens21  
        Port enp2s29  
            tag: 500  
            Interface enp2s29  
        Port enp3s12  
            tag: 400  
            Interface enp3s12  
        Port sw2-cod  
            Interface sw2-cod  
                type: internal  
        Port ens22  
            tag: 200  
            Interface ens22  
    Port bond0  
        trunks: [100, 200, 300, 400, 500]  
        Interface ens19  
        Interface ens20  
ovs_version: "3.3.2"  
[root@sw2-cod ~]#
```

- Режим работы **bond** интерфейса проверить можно с помощью команды **ovs-appctl bond/show**:

```
[root@sw2-cod ~]# ovs-appctl bond/show  
---- bond0 ----  
bond_mode: active-backup  
bond may use recirculation: no, Recirc-ID : -1  
bond-hash-basis: 0  
lb_output action: disabled, bond-id: -1  
updelay: 0 ms  
downdelay: 0 ms  
lacp_status: off  
lacpFallback_ab: false  
active-backup primary: <none>  
active member mac: bc:24:11:2b:a9:69(ens20)  
  
member ens19: enabled  
    may_enable: true  
  
member ens20: enabled  
    active member  
    may_enable: true  
  
[root@sw2-cod ~]
```

- Сетевая подсистема **etcnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления
 - создаём каталог для management интерфейса с именем **mgmt-cod**:

```
mkdir /etc/net/ifaces/mgmt-cod
```

- Описываем файл **options** для создания management интерфейса с именем **mgmt-codt**:

```
vim /etc/net/ifaces/mgmt-cod/options
```

- где:

- **TYPE** - тип интерфейса (**internal**);
- **BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
- **CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
- **BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
- **VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw2-cod
VID=300
"
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "192.168.30.2/24" > /etc/net/ifaces/mgmt-cod/ipv4addressecho "default via
192.168.30.254" > /etc/net/ifaces/mgmt-cod/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```
[root@sw2-cod ~]# ip -c -br -4 a
lo      UNKNOWN  127.0.0.1/8
mgmt-cod UNKNOWN  192.168.30.2/24
[root@sw2-cod ~]# -
```

- Проверить назначенный IP-адрес шлюза по умолчанию можно команжой **ip r**:

```
[root@sw2-cod ~]# ip -c r
default via 192.168.30.254 dev mgmt-cod
192.168.30.0/24 dev mgmt-cod proto kernel scope link src 192.168.30.2
[root@sw2-cod ~]#
```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw1-cod**:

```
[root@sw2-cod ~]# ovs-vsctl show  
a83d6bb0-91b6-49ec-be50-3c19857b3947  
  Bridge sw2-cod  
    Port ens21  
      tag: 100  
      Interface ens21  
    Port mgmt-cod  
      tag: 300  
      Interface mgmt-cod  
        type: internal  
    Port emp2s29  
      tag: 500  
      Interface emp2s29  
    Port emp3s12  
      tag: 400  
      Interface emp3s12  
    Port sw2-cod  
      Interface sw2-cod  
        type: internal  
    Port ens22  
      tag: 200  
      Interface ens22  
    Port bond0  
      trunks: [100, 200, 300, 400, 500]  
      Interface ens19  
      Interface ens20  
  ovs_version: "3.3.2"  
[root@sw2-cod ~]#
```

- Помимо того, что интерфейс **mgmt-cod** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt-cod vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt-cod**:

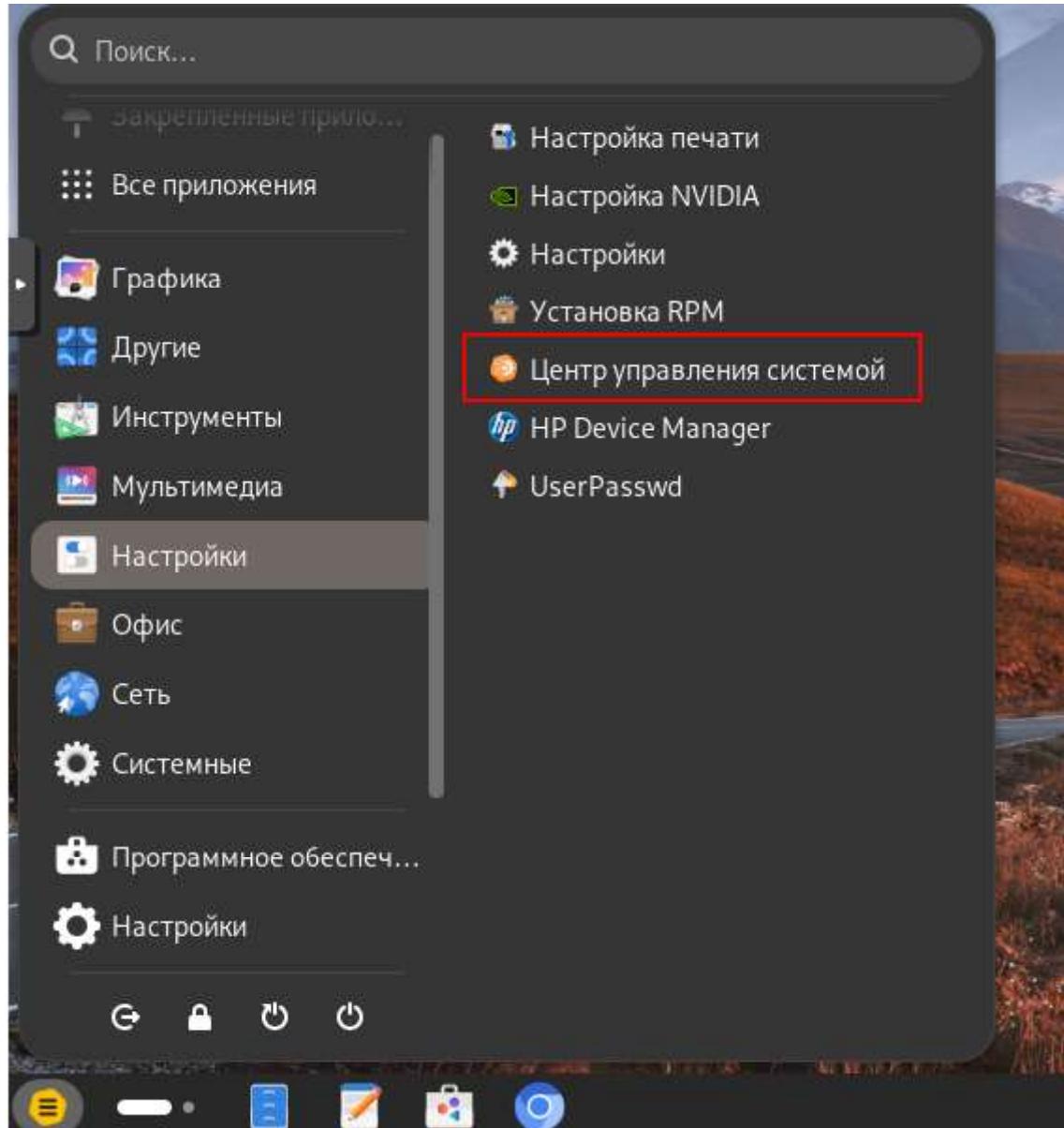
```
[root@sw2-cod ~]# ovs-vsctl list port mgmt-cod  
_uuid : 4db47dc6-c94c-491d-9b53-8642eb526930  
bond_active_slave : []  
bond_downdelay : 0  
bond_fake_iface : false  
bond_mode : []  
bond_updelay : 0  
culans : []  
external_ids : {}  
fake_bridge : false  
interfaces : [69720e2f-9ba0-44f0-8cfb-e460f6f23b94]  
lacp : []  
mac : []  
name : mgmt-cod  
other_config : {}  
protected : false  
qos : []  
rstp_statistics : {}  
rstp_status : {}  
statistics : {}  
status : {}  
tag : 300  
trunks : []  
vlan_mode : native-untagged  
[root@sw2-cod ~]#
```

Вариант реализации:

cli-cod (alt-workstation):

Базовая настройка устройства:

- Перейдём в Центр Управления Системой (ЦУС/acc):



- В ЦУС перейдём в раздел Ethernet-интерфейсы:

 Обновить Переключиться на старую версию  Справка

Система

Информация об установленной системе и её настройка

[Дата и время](#) [Информация о дистрибутиве](#) [Лицензионное соглашение](#) [Настройка zram-swap](#)[Настройка нескольких рабочих мест](#) [Настройка ограничений](#) [Обновление системы](#)[Обновление ядра](#) [Сетевые каталоги](#) [Системные журналы](#) [Системные ограничения](#)[Системные службы](#) [Управление ключами SSL](#)

Пользователи

Управление пользователями системы

[Администратор системы](#) [Аутентификация](#) [Использование диска](#) [Локальные группы](#)[Локальные учётные записи](#)

Брандмауэр

Брандмауэр

[Внешние сети](#) [Перенаправление портов](#) [Список блокируемых хостов](#)

Сеть

Настройка подключения к сети

[Ethernet-интерфейсы](#) [OpenVPN-соединения](#) [PPPoE-соединения](#) [PPTP-соединения](#)[Прокси-сервер](#)

Графический интерфейс

Настройка устройств ввода-вывода

[Дисплей](#) [Клавиатура](#)

- Назначим имя на устройство в соответствие с топологией:

Имя компьютера: cli-cod.cod.ssa2026.region

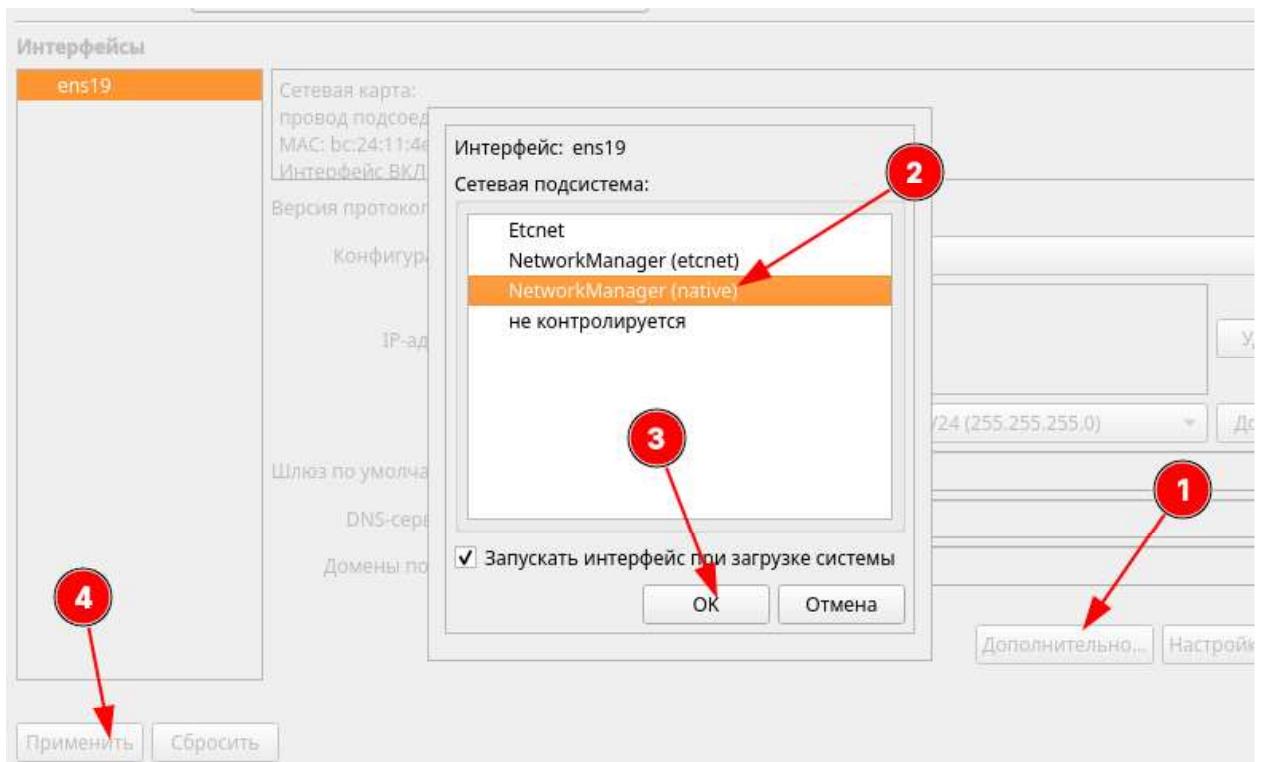
Интерфейсы

enp519	Сетевая карта: провод подсоединен MAC: bc:24:11:c8:17:aa <u>Интерфейс ВКЛЮЧЕН</u>
	Версия протокола IP: <input type="text"/> IPv4
	Конфигурация: <input type="text"/> И
	IP-адреса: <input type="text"/>
	До: <input type="text"/>
	Шлюз по умолчанию: <input type="text"/>
	DNS-серверы: <input type="text"/>
	Домены поиска: <input type="text"/> [не установлено]

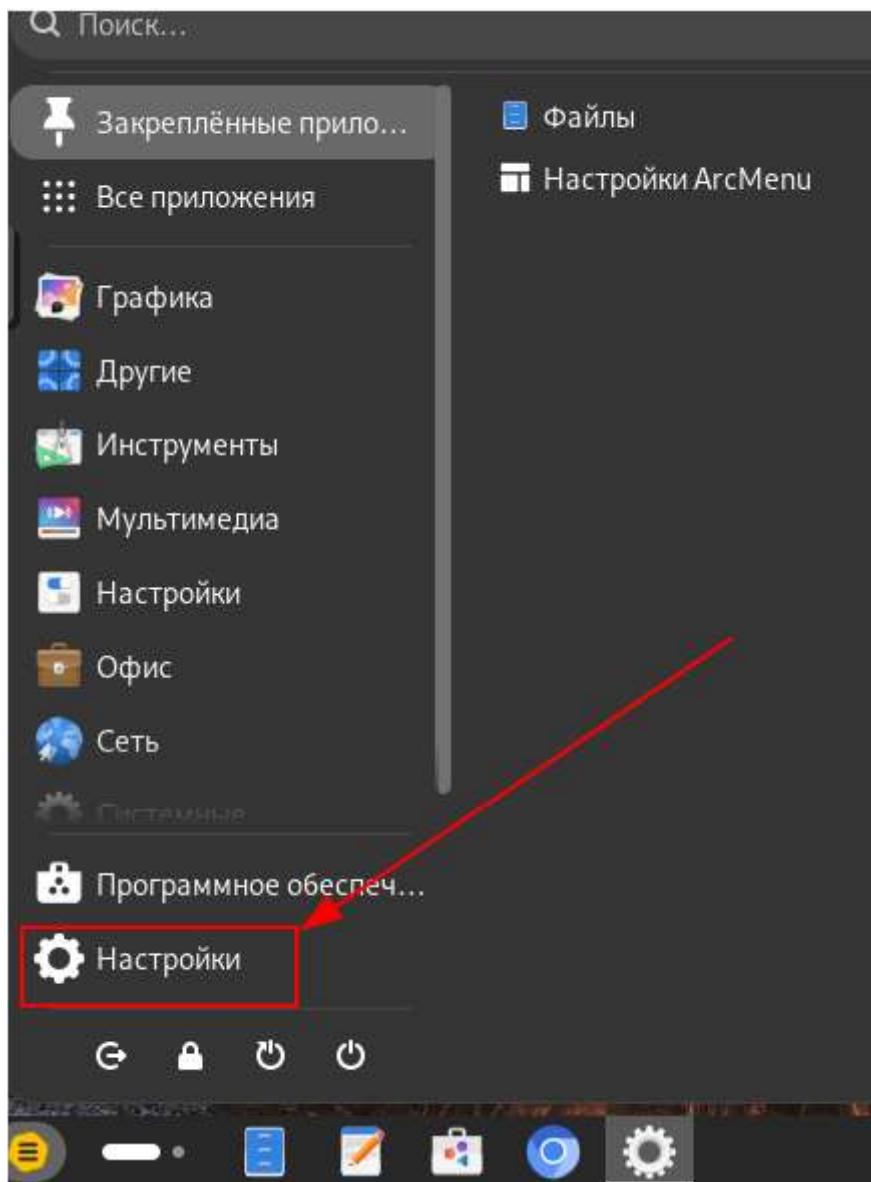
Применить **Сбросить**



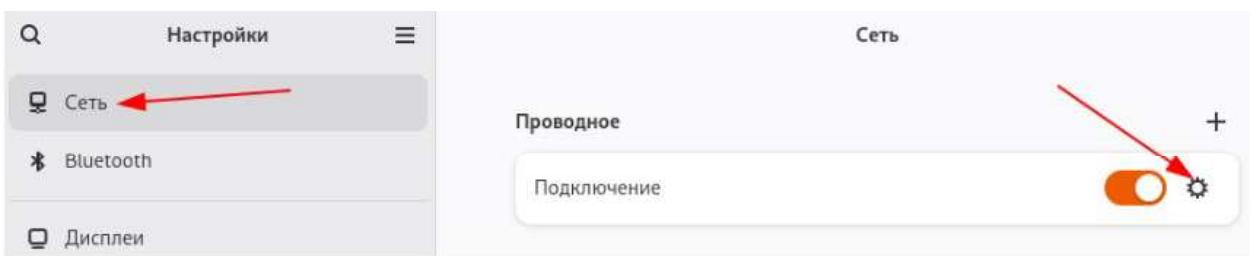
- В качестве режима работы сетевой подсистемы выберем **NetworkManager (native)**:



- Перейдём в обычный **Настройки** для назначения сетевых параметров:



- В разделе **Сеть** перейдём к настройке сетевого подключения:



- Задаём **Вручную** и нажимаем **Применить**:
 - Адрес
 - **Маску сети**
 - Шлюз
 - DNS (в качестве DNS-сервера указываем IP-адрес **srv1-cod**)

Подробности Идентификация IPv4 IPv6 Безопасность

Метод IPv4

- Автоматический (DHCP)
- Вручную
- Общий доступ другим компьютерам

Адреса

Адрес	Маска сети	Шлюз
192.168.40.40	255.255.255.0	192.168.40.254
		(x)

DNS

Автоматически

192.168.10.1

Отделяйте IP-адреса запятыми

- Проверить можно на вкладке **Подробности**:

Отменить Проводное подключение

Подробности Идентификация IPv4 IPv6 Безопасность

Адрес IPv4 192.168.40.40
 Адрес IPv6 fe80::c921:51f0:524:6ee3
 Аппаратный адрес BC:24:11:C8:17:AA
 Маршрут по умолчанию 192.168.40.254
 DNS 192.168.10.1

- Также можно проверить доступ в сеть Интернет:

```
[user@cli-cod ~]$ ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=188 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=83.9 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=86.4 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 83.874/119.478/188.147/48.567 ms
[user@cli-cod ~]$
```

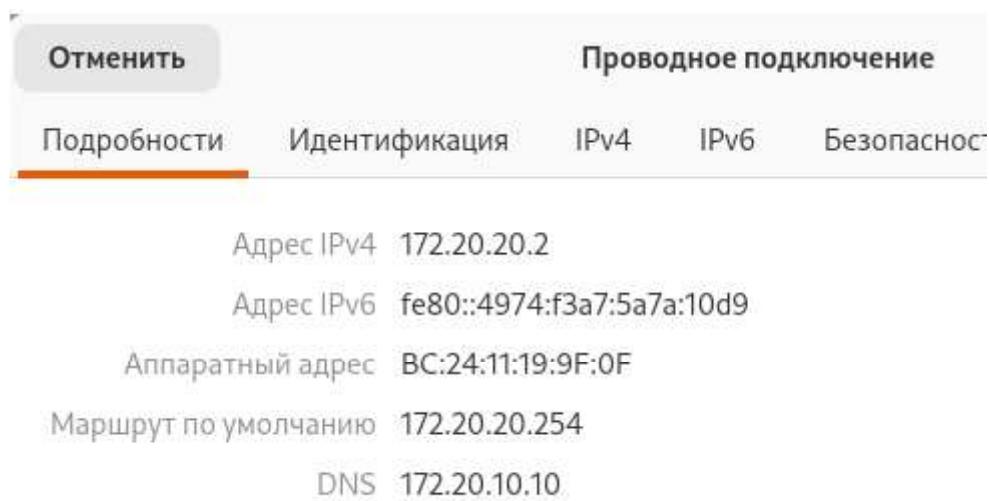
cli2-a (alt-workstation):

Базовая настройка устройства:

- Реализация аналогично **cli-cod**:
 - имя должно быть:

```
[user@cli2-a ~]$ hostname -f  
cli2-a.office.ssa2026.region  
[user@cli2-a ~]$ █
```

- сетевые параметры должны быть:



- доступ в сеть Интернет должен быть:

```
[user@cli2-a ~]$ ping -c3 77.88.8.8 ←  
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.  
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=48.3 ms  
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=46.5 ms  
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=43.5 ms  
  
--- 77.88.8.8 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2002ms  
rtt min/avg/max/mdev = 43.456/46.097/48.345/2.015 ms  
[user@cli2-a ~]$ █
```

- связность с устройствами **COD-а** должна быть, например с **cli-cod**:

```

[user@cli2-a ~]$ ping -c3 192.168.40.40 ←
PING 192.168.40.40 (192.168.40.40) 56(84) bytes of data.
64 bytes from 192.168.40.40: icmp_seq=1 ttl=61 time=104 ms
64 bytes from 192.168.40.40: icmp_seq=2 ttl=61 time=107 ms
64 bytes from 192.168.40.40: icmp_seq=3 ttl=61 time=106 ms

--- 192.168.40.40 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 103.655/105.400/106.921/1.342 ms
[user@cli2-a ~]$ tracepath -n 192.168.40.40 ←
 1?: [LOCALHOST] pmtu 1500
 1: 172.20.20.254 13.304ms
 1: 172.20.20.254 68.337ms
 2: 172.20.20.254 14.964ms pmtu 1476
 2: 10.10.10.1 100.561ms
 3: 169.254.1.1 104.682ms asymm 4
 4: 192.168.40.40 108.025ms reached

Resume: pmtu 1476 hops 4 back 4
[user@cli2-a ~]$
```

admin-cod (alt-workstation):

Базовая настройка устройства:

- Реализация аналогично cli-cod:

```

[user@admin-cod ~]$ hostname -f
admin-cod.cod.ssa2026.region
[user@admin-cod ~]$ ip -c -br -4 a
lo      UNKNOWN    127.0.0.1/8
ens19    UP        192.168.30.30/24
[user@admin-cod ~]$ ip -c r
default via 192.168.30.254 dev ens19 proto static metric 100
192.168.30.0/24 dev ens19 proto kernel scope link src 192.168.30.30 metric 100
[user@admin-cod ~]$ cat /etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search cod.ssa2026.region
nameserver 192.168.10.1
[user@admin-cod ~]$ ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=81.1 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=84.7 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=79.9 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 79.890/81.915/84.728/2.052 ms
[user@admin-cod ~]$
```

srv1-cod (alt-server):

Базовая настройка устройства:

- Для назначения имени устройства согласно топологии используем следующую команду:

```
hostnamectl set-hostname srv1-cod.cod.ssa2026.region; exec bash
```

- Так же рекомендуется указать имя в файле **/etc/sysconfig/network**:

```
vim /etc/sysconfig/network
```

- указать имя в параметре **HOSTNAME**:

```
# When set to no, this may cause most daemons' initscripts skip starting.
NETWORKING=yes

# Used by hotplug/pcmcia/ifplugd scripts to detect current network config
# subsystem.
CONFMETHOD=etcnnet

# Used by rc.susinit to setup system hostname at boot.
HOSTNAME=srv1-cod.cod.ssa2026.region

# This is used by ALTLinux ppp-common to decide if we want to install
# nameserver lines into /etc/resolv.conf or not.
RESOLV_MODS=yes
```

- Проверить можно с помощью команды **hostname** с ключём **-f**:

```
[root@srv1-cod ~]# hostname -f
srv1-cod.cod.ssa2026.region
[root@srv1-cod ~]#
```

Проверяем интерфейсы и определяемся какой к кому направлен (сверка производится по MAC-адресам):

- таким образом, имеем (в данном конкретном случае):
 - **ens19** - интерфейс в сторону **sw2-cod (vlan100)**;
 - **ens21** - интерфейс в сторону **sw2-cod (vlan200)**;

Для каждого интерфейса в директории **/etc/net/ifaces/ <ИМЯ_ИНТЕРФЕЙСА>/** необходимо создать файл **options**

- указав в нём два основных параметра:
 - **TYPE=eth**
 - **BOOTPROTO=static**

```
[root@srv1-cod ~]# ls /etc/net/ifaces/
default ens19 ens20 lo unknown
[root@srv1-cod ~]# cat /etc/net/ifaces/ens19/options
TYPE=eth
BOOTPROTO=static
[root@srv1-cod ~]# cat /etc/net/ifaces/ens20/options
TYPE=eth
BOOTPROTO=static
[root@srv1-cod ~]#
```

- Задаём IP-адрес на интерфейс **ens19 (vlan100)**:

```
echo "192.168.10.1/24" > /etc/net/ifaces/ens19/ipv4address
```

- Задаём IP-адрес шлюза по умолчанию для интерфейса **ens19 (vlan100)**:

```
echo "default via 192.168.10.254" > /etc/net/ifaces/ens19/ipv4route
```

- Задаём IP-адрес на интерфейс **ens20 (vlan200)**:
 - шлюз не задаётся, по заданию данный **vlan** не должен маршрутизоваться;

```
echo "192.168.20.1/24" > /etc/net/ifaces/ens20/ipv4address
```

- Перезагружаем службу **network** для применения всех настроек:

```
systemctl restart network
```

- Проверить:

```
[root@srv1-cod ~]# ip -c -br -4 a
lo          UNKNOWN    127.0.0.1/8
ens19        UP         192.168.10.1/24
ens20        UP         192.168.20.1/24
[root@srv1-cod ~]# ip -c r
default via 192.168.10.254 dev ens19
192.168.10.0/24 dev ens19 proto kernel scope link src 192.168.10.1
192.168.20.0/24 dev ens20 proto kernel scope link src 192.168.20.1
[root@srv1-cod ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=47.2 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=40.6 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=42.7 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 40.648/43.502/47.177/2.727 ms
[root@srv1-cod ~]#
```

srv2-cod (alt-server):

Базовая настройка устройства:

- Реализация аналогично **srv1-cod**:

```

[root@srv2-cod ~]# hostname -f
srv2-cod.cod.ssa2026.region
[root@srv2-cod ~]# ip -c -br -4 a
lo      UNKNOWN    127.0.0.1/8
ens19    UP         192.168.20.2/24
ens20    UP         192.168.10.2/24
[root@srv2-cod ~]# ip -c r
default via 192.168.10.254 dev ens20
192.168.10.0/24 dev ens20 proto kernel scope link src 192.168.10.2
192.168.20.0/24 dev ens19 proto kernel scope link src 192.168.20.2
[root@srv2-cod ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=77.7 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=72.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=72.7 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 72.536/74.323/77.719/2.402 ms
[root@srv2-cod ~]#

```

dc-a (alt-server):

Базовая настройка устройства:

- Реализация аналогично srv1-cod:

```

[root@dc-a ~]# hostname -f
dc-a.office.ssa2026.region
[root@dc-a ~]# ip -c -br -4 a
lo      UNKNOWN    127.0.0.1/8
ens19    UP         172.20.10.10/24
[root@dc-a ~]# ip -c r
default via 172.20.10.254 dev ens19
172.20.10.0/24 dev ens19 proto kernel scope link src 172.20.10.10
[root@dc-a ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=53.0 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=55.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=50.4 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 50.419/52.974/55.536/2.089 ms
[root@dc-a ~]#

```

Вариант реализации:

srv1-cod (alt-server):

- Временно для возможности установки необходимых пакетов зададим публичный DNS-сервер:

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- Устанавливаем пакеты **freeradius** и **freeradius-utils**:

```
apt-get update && apt-get install -y freeradius freeradius-utils
```

- Включаем службу **radiusd** в автозагрузку и запускаем ее:

```
systemctl enable --now radiusd
```

- Создаем клиентов добавляя в файле **/etc/raddb/clients.conf** следующее содержимое:
 - пример конфигурации файла **/etc/raddb/clients.conf** для любого клиента

```
client ALL {
    ipaddr = 0.0.0.0
    netmask = 0
    secret = P@ssw0rd
}
```

- Редактируем файл с пользователями **/etc/raddb/users** добавляя в самый конец следующее содержимое:

```
#####
# You should add test accounts to the TOP of this file! #
# See the example user "bob" above.                         #
#####
netuser Cleartext-Password := "P@ssw0rd"
    Service-Type = Administrative-User,
    Cisco-AVPair = "shell:roles=admin"
```

- Перезапускаем службу **radiusd**:

```
systemctl restart radiusd
```

rtr-cod (ecorouter):

- Выполняем подключение к RADIUS-серверу:

```
rtr-cod(config)#security none
```

```
rtr-cod(config)#aaa radius-server 192.168.10.1 port 1812 secret P@ssw0rd auth
```

```
rtr-cod(config)#
```

- Задаём приоритет:

```
rtr-cod(config)#aaa precedence local radius
```

```
rtr-cod(config)#write memory
```

```
Building configuration...
```

```
rtr-cod(config)#
```

- Проверить возможность входа из-под пользователя **netuser** с паролем **P@ssw0rd**:

```
rtr-cod login: netuser
Password:
User Access Verification

EcoRouterOS version Jasmine 26/12/2024 23:46:47
rtr-cod>enable
rtr-cod#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
rtr-cod(config)#[
```

- Проверить вход под локальной учетной записью даже при доступности RADIUS-сервера:

```
rtr-cod login: admin
Password:
User Access Verification

EcoRouterOS version Jasmine 26/12/2024 23:46:47
rtr-cod>[
```

- Проверить доступ по **SSH** из-под пользователя **netuser** с паролем **P@ssw0rd**:
 - например с **srv1-cod**

```

root@srv1-cod ~]# ssh netuser@172.16.1.1
The authenticity of host '172.16.1.1 (172.16.1.1)' can't be established.
ED25519 key fingerprint is SHA256:k5mMnJ+uMSuWI00/QWzwU/CJFUsu/i46KD3nYxI0JjI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.16.1.1' (ED25519) to the list of known hosts.
netuser@172.16.1.1's password:

User Access Verification

EcoRouterOS version Jasmine 26/12/2024 23:46:47
rtr-cod>enable
rtr-cod#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
rtr-cod(config)#

```

sw1-cod u sw2-cod (alt-server):

- Временно для возможности установки необходимых пакетов зададим публичный DNS-сервер:

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- Устанавливаем пакет **pam_radius**:

```
apt-get update && apt-get install -y pam_radius
```

- Редактируем конфигурационный файл **/etc/pam_radius_auth.conf**:

```

# server[:port]      shared secret    timeout (s)   source_ip          vrf
192.168.10.1      P@ssw0rd        3
#127.0.0.1          secret          3
#other-server       other-secret     5           192.168.1.10        vrf-blue
#[2001:0db8:85a3::4]:1812  other6-secret  3           [2001:0db8:85a3::3]  vrf-red
#other-other-server other-other-secret 5           0
#
# having localhost in your radius configuration is a Good Thing.
#

```

- Редактируем конфигурационный файл **/etc/pam.d/sshd**:

```

#%PAM-1.0
auth      required      pam_userpass.so
auth      sufficient    pam_radius_auth.so
auth      include       common-login-use_first_pass
account  include       common-login
password include       common-login
session  include       common-login

```

- Редактируем конфигурационный файл **/etc/pam.d/system-auth-local**:

```
#%PAM-1.0
auth      sufficient  pam_radius_auth.so
auth      include    system-auth-local-only
auth      include    system-auth-common
account   include    system-auth-local-only
account   include    system-auth-common
password  include    system-auth-local-only
password  include    system-auth-common
session   include    system-auth-local-only
session   include    system-auth-common
```

- Также в случае с Linux, данного пользователя необходимо создать локально:

```
useradd netuser
```

- Проверить возможность входа из-под пользователя **netuser** с паролем **P@ssw0rd**:

```
Welcome to ALT Server 11.0 (Mendelevium)!

Hostname: sw2-cod.cod.ssa2026.region
IP: 192.168.30.2
sw2-cod login: netuser
Password:
[netuser@sw2-cod ~]$
```

- Проверить доступ по **SSH** из-под пользователя **netuser** с паролем **P@ssw0rd**:
 - например с **srv1-cod**

```
Last login: Mon Nov 17 13:08:16 MSK 2025 on tt01
[root@srv1-cod ~]# ssh netuser@192.168.30.2
The authenticity of host '192.168.30.2 (192.168.30.2)' can't be established.
ED25519 key fingerprint is SHA256:A8TkadFGEBXdLuLNzB5Bgm2Jtf6g0UdSDsYusrCg05w.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.30.2' (ED25519) to the list of known hosts.
netuser@192.168.30.2's password:
Last login: Mon Nov 17 13:08:04 2025
[netuser@sw2-cod ~]$
```

Вариант реализации:

rtr-cod и rtr-a (ecorouter):

- Устанавливаем корректный часовой пояс:

```
rtr-cod(config)#ntp timezone utc+3
```

```
rtr-cod(config)#
```

- Устанавливаем синхронизацию времени с **100.100.100.100**:

```
rtr-cod(config)#ntp server 100.100.100.100
```

```
rtr-cod(config)#write memory
```

```
Building configuration...
```

```
rtr-cod(config)#
```

- Проверить текущий часовой пояс можно командой **show ntp timezone** из режима администрирования (**enable**):

```
rtr-cod#show ntp timezone
System Time zone: Europe/Moscow
rtr-cod#
```

- Проверить адреса ntp-серверов для синхронизации можно командой **show ntp status** из режима администрирования (**enable**):

```
rtr-cod#show ntp status
Status  Description
*      best
+      sync
-      failed
?      unknown

-----
Status |     VR name    |     Server      | Stratum |   Delay   | Version |
-----|-----|-----|-----|-----|-----|-----|
*| default | 100.100.100.100 |      5 | 0.2069 |      4 |
rtr-cod#
```

sw1-cod, sw2-cod, cli-cod, srv1-cod, srv2-cod, admin-cod, sw1-a, sw2-a, dc-a, cli1-a u cli2-a (alt-server, alt-workstation):

- Устанавливаем корректный часовой пояс:

```
timedatectl set-timezone Europe/Moscow
```

- Редактируем конфигурационный файл /etc/chrony.conf:

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (https://www.pool.ntp.org/join.html
#pool pool.ntp.org iburst
pool 100.100.100.100 iburst

# Record the rate at which the system clock gains/losses time.
```

- Перезапускаем службу chronyd:

```
systemctl restart chronyd
```

- Проверяем с каким сервером синхронизировалось время с помощью команды chronyc tracking:

```
[root@sw1-cod ~]# chronyc tracking
Reference ID      : 64646464 (100.100.100.100)
Stratum          : 6
Ref time (UTC)   : Mon Nov 17 10:15:41 2025
System time      : 0.000004234 seconds slow of NTP time
Last offset      : +0.004109129 seconds
RMS offset       : 0.004109129 seconds
Frequency        : 9.536 ppm slow
Residual freq    : -2878.252 ppm
Skew             : 11.929 ppm
Root delay       : 0.096411645 seconds
Root dispersion  : 0.071875237 seconds
Update interval  : 2.0 seconds
Leap status       : Normal
[root@sw1-cod ~]# -
```

- Проверяем часовой пояс с помощью команды timedatectl:

```
[root@sw1-cod ~]# timedatectl
          Local time: Mon 2025-11-17 13:16:43 MSK
          Universal time: Mon 2025-11-17 10:16:43 UTC
                    RTC time: Mon 2025-11-17 10:16:43
                   Time zone: Europe/Moscow (MSK, +0300)
System clock synchronized: yes
          NTP service: active
     RTC in local TZ: no
[root@sw1-cod ~]# -
```

admin-cod (alt-workstation):

- Открываем браузер и переходим по <https://192.168.30.254:8443> (IP-адрес fw-cod):
 - выполняем вход в веб-интерфейс управления fw-cod
 - переходим Сервисы -> NTP-сервер и нажимаем Добавить:

The screenshot shows the web-based management interface of the Ideco NGFW Novum firewall. The URL in the browser is <https://192.168.30.254:8443/#/scontrol/ntp>. The interface is in Russian. On the left, there's a sidebar with various service icons and names. The 'Сервисы' (Services) icon is highlighted with a red box. Within 'Сервисы', the 'NTP-сервер' (NTP Server) icon is also highlighted with a red box. The main content area on the right is titled 'NTP-сервер' and shows the status 'Работает' (Working). It displays the current server time as '17 ноября 2025 г., 13:29:11'. There are two toggle switches: 'NTP-сервер на всех локальных интерфейсах' (NTP server on all local interfaces) and 'Перехват NTP-запросов' (NTP request interception), both of which are off. At the bottom, there's a large orange 'Добавить' (Add) button with a '+' icon, and a red arrow points from this button to the 'Адрес NTP-сервера' (NTP server address) input field.

- Указываем 100.100.100.100 и нажимаем Добавить:

NTP-сервер

Работает



Добавление NTP-сервера

NTP-сервер
100.100.100.100

IP-адрес или доменное имя

Комментарий

0/256

Добавить

Отмена

- Результат успешного добавления NTP-сервера:

NTP-сервер

Работает



Время на сервере: 17 ноября 2025 г., 13:30:29 (Москва)

- NTP-сервер на всех локальных интерфейсах (порт 123/UDP)
 Перехват NTP-запросов

+ Добавить

Фильтры

Отображение

Адрес NTP-сервера

100.100.100.100

Комментарий

- Проверяем с каким сервером синхронизировалось время с помощью команды `chronyc tracking`:

Терминал [?](#)

```
[admin@fw-cod ~]# chronyc tracking
Reference ID      : 64646464 (100.100.100.100)
Stratum          : 6
Ref time (UTC)   : Mon Nov 17 10:31:03 2025
System time      : 0.007790932 seconds fast of NTP
Last offset      : +0.008523416 seconds
RMS offset       : 0.008937536 seconds
Frequency        : 10.231 ppm slow
Residual freq    : +358.057 ppm
Skew              : 5.477 ppm
Root delay       : 0.080065280 seconds
Root dispersion  : 0.027800325 seconds
Update interval  : 16.4 seconds
Leap status       : Normal
[admin@fw-cod ~]#
```

- Проверяем часовой пояс с помощью команды `timedatectl`:

Терминал [?](#)

```
[admin@fw-cod ~]# timedatectl
          Local time: Пн 2025-11-17 13:32:00 MSK
          Universal time: Пн 2025-11-17 10:32:00 UTC
                  RTC time: Пн 2025-11-17 10:32:00
                    Time zone: Europe/Moscow (MSK, +0300)
System clock synchronized: yes
          NTP service: active
      RTC in local TZ: no
[admin@fw-cod ~]#
```

Вариант реализации:

srv1-cod (alt-server):

- Установим пакет bind и bind-utils:

```
apt-get install -y bind bind-utils
```

- Отредактируем конфигурационный файл /etc/bind/options.conf:

```
options {  
    version "unknown";  
    directory "/etc/bind/zone";  
    dump-file "/var/run/named/named_dump.db";  
    statistics-file "/var/run/named/named.stats";  
    recursing-file "/var/run/named/named.recurising";  
    secroots-file "/var/run/named/named.secroots";  
  
    // disables the use of a PID file  
    pid-file none;  
  
    /*  
     * Oftenly used directives are listed below.  
     */  
  
    listen-on { any; };  
    listen-on-v6 { none; };  
  
    /*  
     * If the forward directive is set to "only", the server will only  
     * query the forwarders.  
     */  
    forward first;  
    forwarders { 100.100.100.100; };  
  
    /*  
     * Specifies which hosts are allowed to ask ordinary questions.  
     */  
    allow-query { any; };  
  
    /*  
     * This lets "allow-query" be used to specify the default zone access  
     * level rather than having to have every zone override the global  
     * value. "allow-query-cache" can be set at both the options and view  
     * levels. If "allow-query-cache" is not set then "allow-recursion" is  
     * used if set, otherwise "allow-query" is used if set unless  
     * "recursion no;" is set in which case "none;" is used, otherwise the  
     * default (localhost; localnets;) is used.  
     */  
    allow-query-cache { any; };  
  
    /*  
     * Specifies which hosts are allowed to make recursive queries  
     * through this server. If not specified, the default is to allow  
     * recursive queries from all hosts. Note that disallowing recursive  
     * queries for a host does not prevent the host from retrieving data  
     * that is already in the server's cache.  
     */  
    allow-recursion { any; };
```

- Добавить в конфигурационный файл **/etc/bind/rfc1912.local** информацию о файлах зон прямого и обратного просмотра:

```
zone "cod.ssa2026.region" {
    type master;
    file "cod.ssa2026.region";
    allow-transfer { 172.20.10.10; };
};

zone "168.192.in-addr.arpa" {
    type master;
    file "168.192.in-addr.arpa";
    allow-transfer { 172.20.10.10; };
};

zone "office.ssa2026.region" {
    type forward;
    forward only;
    forwarders { 172.20.10.10; };
};

zone "20.172.in-addr.arpa" {
    type forward;
    forward only;
    forwarders { 172.20.10.10; };
};
```

- Скопировать файл шаблона для зоны прямого просмотра:

```
cp /etc/bind/zone/localhost /etc/bind/zone/cod.ssa2026.region
```

- Выдать права на файл зоны прямого просмотра:

```
chown root:named /etc/bind/zone/cod.ssa2026.region
```

- Привести файл **/etc/bind/zone/cod.ssa2026.region** зоны прямого просмотра к следующему виду:

\$TTL	1D					
#		IN	SOA	cod.ssa2026.region.	root.cod.ssa2026.region.	(
				2025110500	; serial	
				12H	; refresh	
				1H	; retry	
				1W	; expire	
				1H	; ncache	
)		
		IN	NS	cod.ssa2026.region.		
		IN	A	192.168.10.1		
rtr-cod		IN	A	172.16.1.1		
fw-cod		IN	A	192.168.10.254		
sw1-cod		IN	A	192.168.30.1		
sw2-cod		IN	A	192.168.30.2		
cli-cod		IN	A	192.168.40.40		
srv1-cod		IN	A	192.168.10.1		
srv2-cod		IN	A	192.168.10.2		
sip-cod		IN	A	192.168.50.50		
admin-cod		IN	A	192.168.30.30		
monitoring		IN	CNAME	srv1-cod.ssa2026.region.		

- Скопировать файл шаблона для зоны обратного просмотра:

```
cp /etc/bind/zone/localhost /etc/bind/zone/168.192.in-addr.arpa
```

- Выдать права на файл зоны прямого просмотра:

```
chown root:named /etc/bind/zone/168.192.in-addr.arpa
```

- Привести файл **/etc/bind/zone/168.192.in-addr.arpa** зоны обратного просмотра к следующему виду:

```
$TTL 1D
@ IN SDA    cod.ssa2026.region. root.cod.ssa2026.region. (
    2025110500 ; serial
    12H        ; refresh
    1H        ; retry
    1W        ; expire
    1H        ; ncache
)
254.18 IN NS   cod.ssa2026.region.
1.30  IN PTR  fw-cod.cod.ssa2026.region.
2.30  IN PTR  sw1-cod.cod.ssa2026.region.
40.40 IN PTR  sw2-cod.cod.ssa2026.region.
1.10  IN PTR  cli-cod.cod.ssa2026.region.
2.10  IN PTR  srv1-cod.cod.ssa2026.region.
50.50 IN PTR  srv2-cod.cod.ssa2026.region.
30.30 IN PTR  sip-cod.cod.ssa2026.region.
~
```

- Включить и добавить в автозагрузку службу **bind**:

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

- Задаём DNS-сервер и домен поиска:

```
cat <<EOF > /etc/net/ifaces/ens19/resolv.conf
```

```
search cod.ssa2026.region
```

```
nameserver 127.0.0.1
```

```
EOF
```

- Перезагружаем сервер:

```
reboot
```

- Проверяем записи типа A:

```
[root@srv1-cod ~]# host fw-cod
fw-cod.cod.ssa2026.region has address 192.168.10.254
[root@srv1-cod ~]# host sw1-cod
sw1-cod.cod.ssa2026.region has address 192.168.30.1
[root@srv1-cod ~]# host sw2-cod
sw2-cod.cod.ssa2026.region has address 192.168.30.2
[root@srv1-cod ~]# host cli-cod
cli-cod.cod.ssa2026.region has address 192.168.40.40
[root@srv1-cod ~]# host srv1-cod
srv1-cod.cod.ssa2026.region has address 192.168.10.1
[root@srv1-cod ~]# host srv2-cod
srv2-cod.cod.ssa2026.region has address 192.168.10.2
[root@srv1-cod ~]# host sip-cod
sip-cod.cod.ssa2026.region has address 192.168.50.50
[root@srv1-cod ~]# host admin-cod
admin-cod.cod.ssa2026.region has address 192.168.30.30
[root@srv1-cod ~]# -
```

- Проверяем записи типа PTR:

sw1-cod u sw2-cod (alt-server):

- Задаём в качестве DNS-сервера **srv1-cod**:

```
cat <<EOF > /etc/net/ifaces/mgmt-cod/resolv.conf
search cod.ssa2026.region
nameserver 192.168.10.1
EOF
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить:

```

root@sw1-cod ~# cat /etc/resolv.conf ←
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search cod.ssa2026.region
nameserver 192.168.10.1
root@sw1-cod ~# ping -c3 cod.ssa2026.region ←
PING cod.ssa2026.region (192.168.10.1) 56(84) bytes of data.
64 bytes from srv1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=1 ttl=63 time=2.28 ms
64 bytes from srv1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=2 ttl=63 time=2.97 ms
64 bytes from srv1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=3 ttl=63 time=2.93 ms

--- cod.ssa2026.region ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 2.281/2.724/2.966/0.313 ms
root@sw1-cod ~# ping -c3 ya.ru ←
PING ya.ru (77.88.55.242) 56(84) bytes of data.
64 bytes from ya.ru (77.88.55.242): icmp_seq=1 ttl=49 time=98.4 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=2 ttl=49 time=95.9 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=3 ttl=49 time=94.9 ms

--- ya.ru ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 94.859/96.375/98.362/1.468 ms
root@sw1-cod ~#

```

srv2-cod (alt-server):

- Задаём в качестве DNS-сервера **srv1-cod**:

```

cat <<EOF > /etc/net/ifaces/ens20/resolv.conf

search cod.ssa2026.region

nameserver 192.168.10.1

EOF

```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить:

```
[root@sru2-cod ~]# cat /etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search cod.ssa2026.region
nameserver 192.168.10.1
[root@sru2-cod ~]# ping -c3 cod.ssa2026.region
PING cod.ssa2026.region (192.168.10.1) 56(84) bytes of data.
64 bytes from sru1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=1 ttl=64 time=1.41 ms
64 bytes from sru1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=2 ttl=64 time=2.49 ms
64 bytes from sru1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=3 ttl=64 time=2.10 ms
--- cod.ssa2026.region ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.413/2.001/2.493/0.446 ms
[root@sru2-cod ~]# ping -c3 ya.ru
PING ya.ru (77.88.44.242) 56(84) bytes of data.
64 bytes from ya.ru (77.88.44.242): icmp_seq=1 ttl=47 time=44.4 ms
64 bytes from ya.ru (77.88.44.242): icmp_seq=2 ttl=47 time=43.4 ms
64 bytes from ya.ru (77.88.44.242): icmp_seq=3 ttl=47 time=41.5 ms
--- ya.ru ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 41.497/43.098/44.352/1.191 ms
[root@sru2-cod ~]#
```

Вариант реализации:

dc-a (alt-server):

- Временно для возможности установки необходимых пакетов зададим публичный DNS-сервер:

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- Устанавливаем пакеты **task-samba-dc**, **bind** и **bind-utils**:

```
apt-get update && apt-get install -y task-samba-dc bind bind-utils
```

- Если при установке системы в настройках сети было указано полное имя домена (например, dc-a.office.ssa2026.region), система может автоматически создать зону office.ssa2026.region, что приведёт к конфликту с Samba при запуске bind
- Для решения проблемы необходимо закомментировать все строки в файле **/etc/bind/local.conf**, это предотвратит автозагрузку конфликтующих зон:

```
//include "/etc/bind/rfc1912.conf";  
  
// Consider adding the 1918 zones here,  
// if they are not used in your organization.  
//      include "/etc/bind/rfc1918.conf";  
  
// Add other zones here
```

- Отключите chroot:

```
control bind-chroot disabled
```

- Отключите KRB5RCACHETYPE:

```
echo 'KRB5RCACHETYPE="none"' >> /etc/sysconfig/bind
```

- Подключите плагин BIND_DLZ:

```
echo 'include "/var/lib/samba/bind-dns/named.conf";' >> /etc/bind/named.conf
```

- Отредактируйте файл **/etc/bind/options.conf**:

- в раздел **options** добавьте строки:

- **tkey-gssapi-keytab** — путь к ключевой таблице для GSS-API (интеграция с Kerberos);
 - **minimal-responses** — уменьшает объём ответов;
 - **listen-on** — IP-адреса, на которых принимаются запросы;
 - **allow-query** — разрешённые подсети для DNS-запросов;
 - **allow-recursion** — подсети, которым разрешены рекурсивные запросы;
 - **forwarders** — внешние DNS-серверы для пересылки;;

- **forward first** — сначала пересыпать, затем кешировать;

```

dump-file "/var/run/named/named_dump.db";
statistics-file "/var/run/named/named.stats";
recursing-file "/var/run/named/named.reCURsing";
secroots-file "/var/run/named/named.secroots";

tkey-gssapi-keytab "/var/lib/samba/bind-dNS/dns.keytab";
minimal-responses yes;

// disables the use of a PID file
pid-file none;

/*
 * Oftenly used directives are listed below.
 */

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

/*
 * If the forward directive is set to "only", the server will only
 * query the forwarders.
 */
forward first;
forwarders { 100.100.100.100; };

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

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

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

```

- В раздел **logging** добавьте строку:

- **logging** — подавление предупреждений о «lame servers»

```

logging {
    category lame-servers { null; };
    // The default_debug channel has the special property that it only
    // produces output when the server's debug level is non-zero. It
    // normally writes to a file called named.run in the server's working
    // directory.
}

```

- Восстановить к начальному состоянию Samba:

```

rm -f /etc/samba/smb.conf{,n} -f /etc/samba/smb.conf{,n} -rf /var/cache/sambamkdir -p
/var/lib/samba/sysvol

```

- Интерактивное создание домена:

samba-tool domain provision

- в качестве DNS backend указать BIND9_DLZ

```
[root@dc-a ~]# samba-tool domain provision ←
Realm [OFFICE.SSA2026.REGION]:
Domain [OFFICE]:
Server Role (dc, member, standalone) [dc]:
DNS backend (SAMBA_INTERNAL, BIND9_FLATFILE, BIND9_DLZ, NONE) [SAMBA_INTERNAL]: BIND9_DLZ
Administrator password:
Retype password: _
```

- результат успешного создания домена в интерактивном режиме:

```
97 Server Role:      active directory domain controller
98 Hostname:        dc-a
99 NetBIOS Domain:  OFFICE
00 DNS Domain:     office.ssa2026.region
01 DOMAIN SID:     S-1-5-21-2055079406-4042076010-175
```

- Включить в автозагрузку службы **samba** и **bind**, также запустить их:

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

- При создании домена Samba автоматически генерирует корректный файл **krb5.conf** для домена в каталоге **/var/lib/samba/private/**
 - Можно просто заменить этим файлом файл, находящийся в каталоге **/etc/**:

```
cp /var/lib/samba/private/krb5.conf /etc/krb5.conf
```

- Перезагрузить службу **samba**:

```
systemctl restart samba
```

- Просмотр общей информации о домене можно выполнить с помощью команды **samba-tool domain info 127.0.0.1**:

```
[root@dc-a ~]# samba-tool domain info 127.0.0.1
Forest          : office.ssa2026.region
Domain          : office.ssa2026.region
Netbios domain  : OFFICE
DC name         : dc-a.office.ssa2026.region
DC netbios name : DC-A
Server site     : Default-First-Site-Name
Client site    : Default-First-Site-Name
```

- Просмотр предоставляемых служб можно выполнить с помощью команды **smbclient -L localhost -Uadministrator**:
 - netlogon** и **sysvol** создаются автоматически и необходимы для работы контроллера домена

```
[root@dc-a ~]# smbclient -L localhost -U administrator  
Password for [OFFICE\administrator]:  
  
Sharename      Type      Comment  
-----  
sysvol          Disk  
netlogon        Disk  
IPC$           IPC       IPC Service (Samba 4.21.9-alt1)  
SMB1 disabled -- no workgroup available  
[root@dc-a ~]#
```

- Проверка конфигурации DNS:
 - проверка наличия nameserver 127.0.0.1 в /etc/resolv.conf:

```
[root@dc-a ~]# cat /etc/resolv.conf  
# Generated by resolvconf  
# Do not edit manually, use  
# /etc/net/ifaces/<interface>/resolv.conf instead.  
search office.ssa2026.region  
nameserver 127.0.0.1  
[root@dc-a ~]#
```

- проверка имён хостов "_kerberos._udp.":

```
[root@dc-a ~]# host -t SRV _kerberos._udp.office.ssa2026.region.  
_kerberos._udp.office.ssa2026.region has SRV record 0 100 88 dc-a.office.ssa2026.region.  
[root@dc-a ~]#
```

- проверка имён хостов "_ldap._tcp.":

```
[root@dc-a ~]# host -t SRV _ldap._tcp.office.ssa2026.region.  
_ldap._tcp.office.ssa2026.region has SRV record 0 100 389 dc-a.office.ssa2026.region.  
[root@dc-a ~]#
```

- проверка имён хостов "адрес хоста.":

```
[root@dc-a ~]# host -t A dc-a.office.ssa2026.region.  
dc-a.office.ssa2026.region has address 172.20.10.10  
[root@dc-a ~]#
```

- Проверка Kerberos-аутентификации (имя домена должно быть в верхнем регистре):

```
[root@dc-a ~]# kinit administrator@OFFICE.SSA2026.REGION  
Password for administrator@OFFICE.SSA2026.REGION:  
Warning: Your password will expire in 41 days on Wed Dec 31 09:22:12 2025  
[root@dc-a ~]#
```

- Просмотр полученного билета:

```
root@dc-a:~# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: administrator@OFFICE.SSA2026.REGION

Valid starting     Expires            Service principal
11/19/25 09:49:33  11/19/25 19:49:33  krbtgt/OFFICE.SSA2026.REGION@OFFICE.SSA2026.REGION
                  renew until 11/20/25 09:45:24
root@dc-a:~#
```

cli1-a и cli2-a (alt-workstation):

- Для ввода в домен установим пакет **task-auth-ad-sssd**:

```
apt-get update && apt-get install -y task-auth-ad-sssd
```

- Для ввода компьютера в домен в ЦУС необходимо выбрать пункт **Пользователи → Аутентификация**
- В окне модуля **Аутентификация** следует выбрать пункт **Домен Active Directory**, заполнить поля (Домен, Рабочая группа, Имя компьютера), выбрать пункт SSSD (в единственном домене) и нажать кнопку **Применить**

Аутентификация

Локальная база пользователей

Домен ALT Linux или Astra Linux Directory

Домен:

Кэшировать аутентификацию при недоступности сервера домена

Домен Active Directory

Домен:

Рабочая группа:

Имя компьютера:

SSSD (в единственном домене)

Winbind (в сложных доменах)

Домен FreeIPA

Внимание! Не установлен пакет task-auth-freeipa. Аутентификация в домене FreeIPA не

Домен:

Имя компьютера:

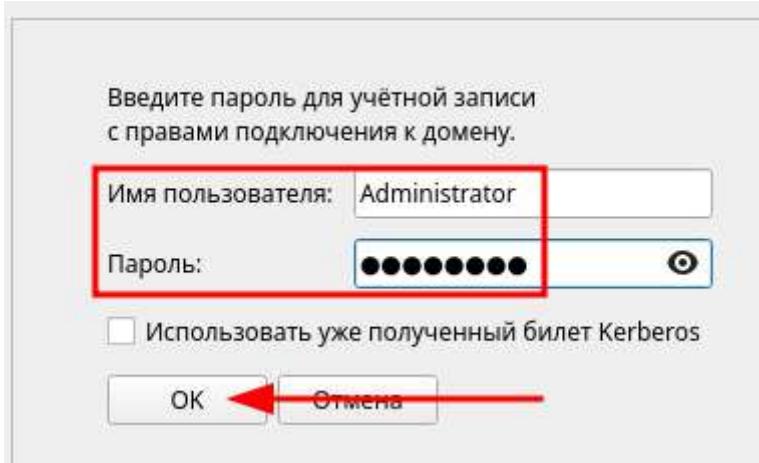
[Настройки SSSD...](#)

Внимание!

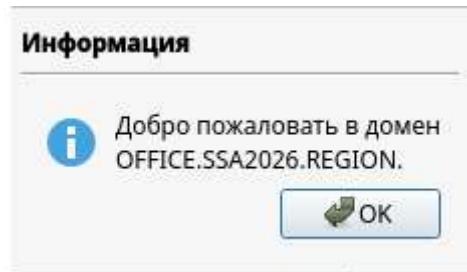
Изменение домена заработает только после перезагрузки компьютера

Восстановить файлы конфигурации по умолчанию (smb.conf, krb5.conf, sssd.conf).

- В открывшемся окне необходимо ввести имя пользователя, имеющего право вводить машины в домен, и его пароль и нажать кнопку ОК:



- Результат успешного присоединения к домену:



- Перезагрузить рабочую станцию для применения всех настроек

cli1-a (alt-workstation):

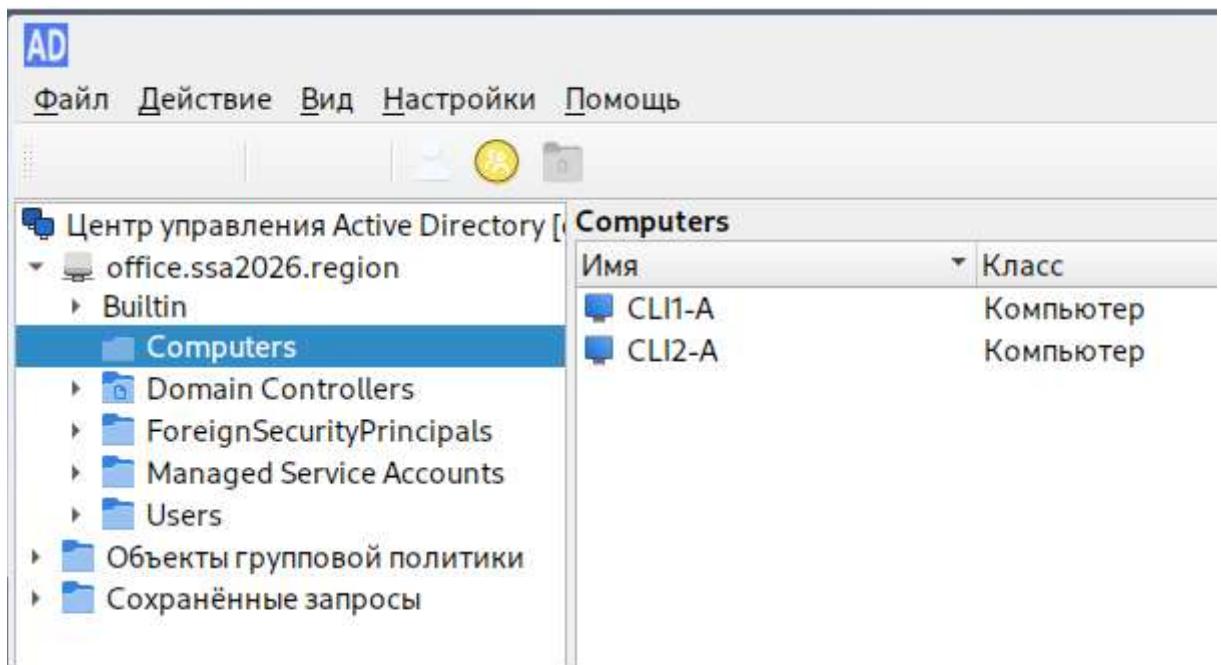
- Установить пакет **admc**:

```
apt-get install -y admc
```

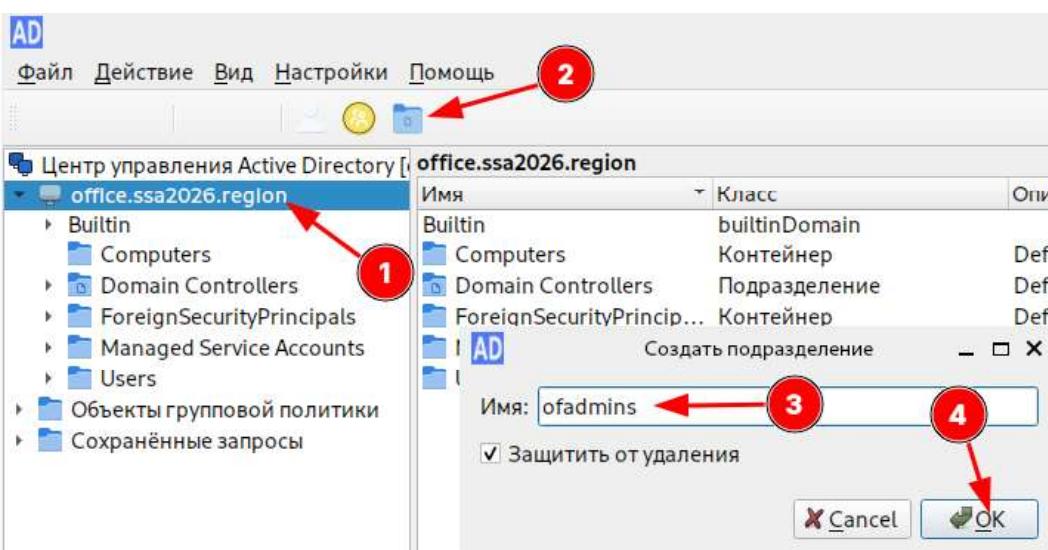
- Запуск **ADMC** осуществляется из меню запуска приложений:
пункт **Системные** → **ADMC** или из командной строки (команда **admc**)
- Для запуска **ADMC** необходимо предварительно получить ключ Kerberos для администратора домена:

```
[user@cli1-a ~]$ kinit administrator@OFFICE.SSA2026.REGION
Password for administrator@OFFICE.SSA2026.REGION:
Warning: Your password will expire in 41 days on Ср 31 дек 2025 09:22:12
[user@cli1-a ~]$
```

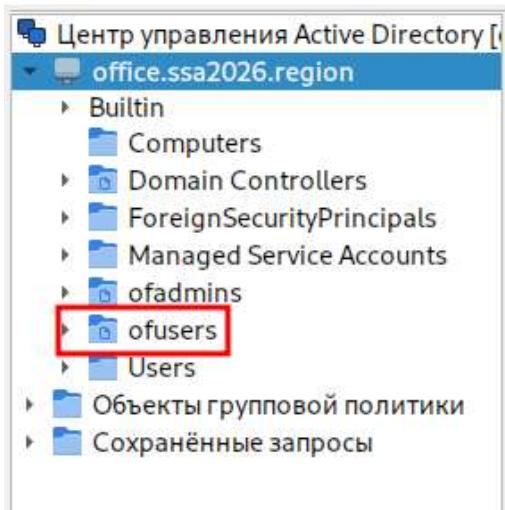
- Результат успешного запуска **admc**:



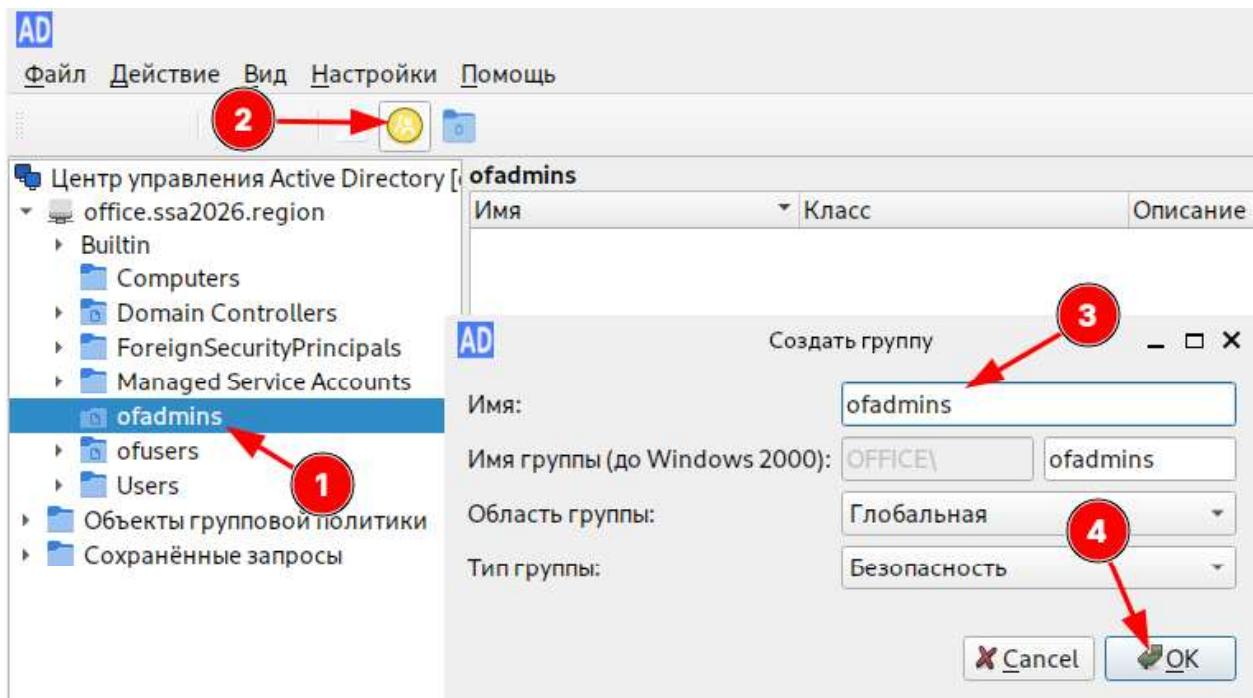
- Создать **ofadmins**:



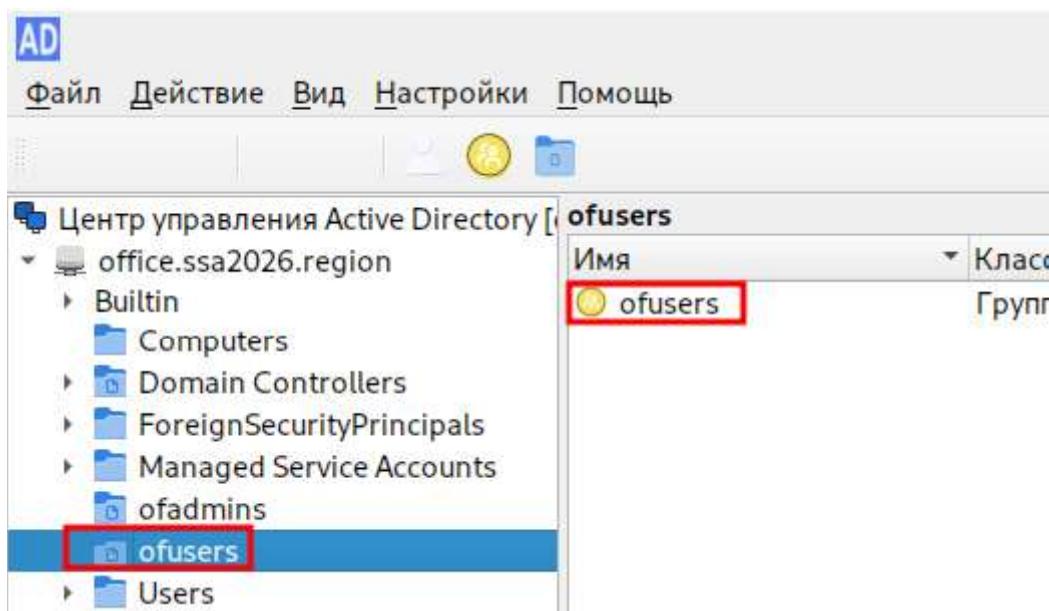
- Создать **ofusers**:



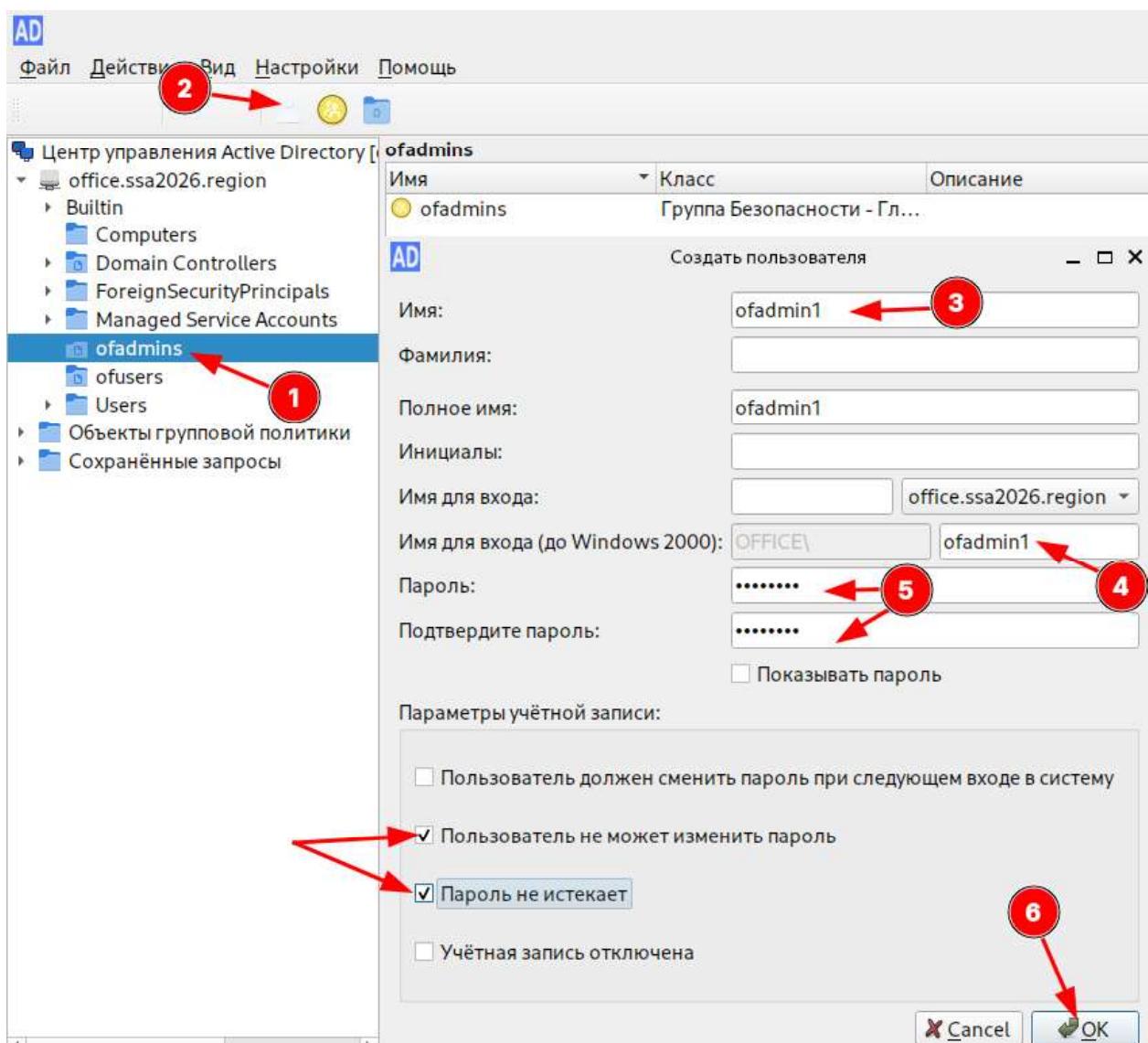
- Создать группу **ofadms** в подразделение **ofadms**:



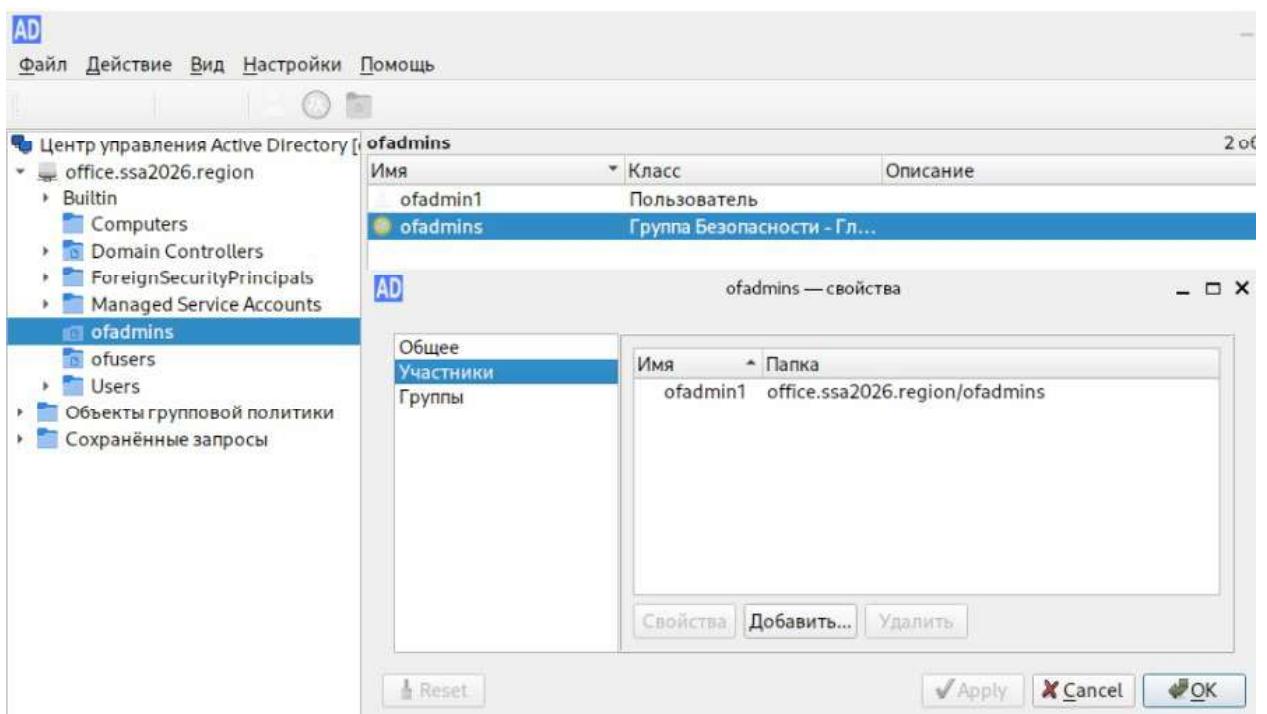
- Создать группу **ofusers** в подразделение **ofusers**:



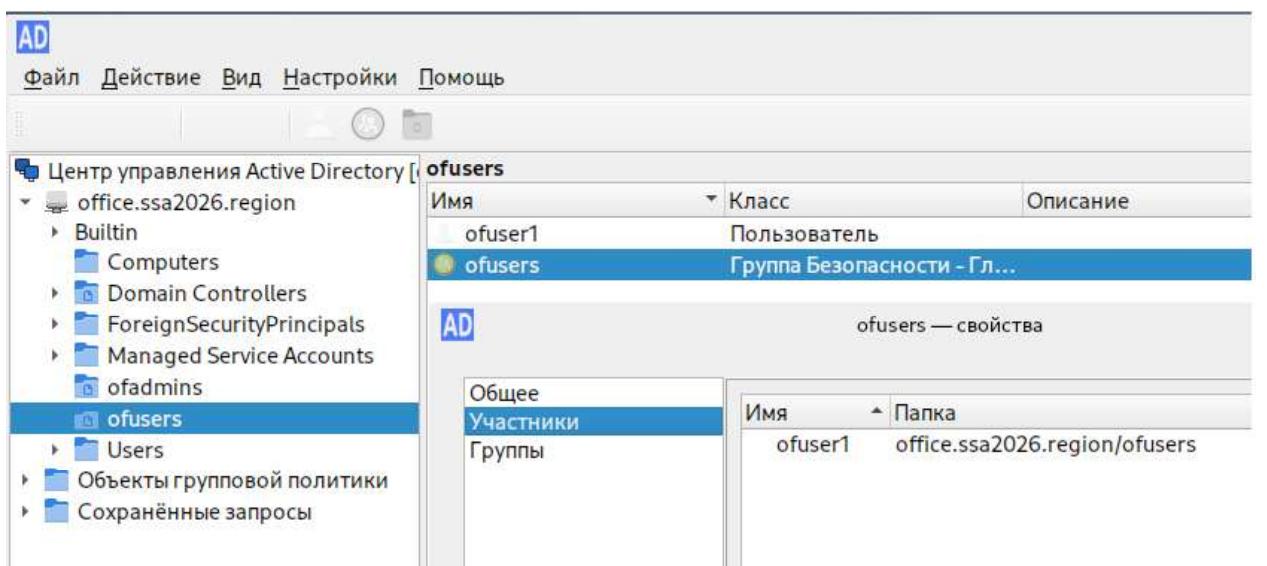
- Создать пользователя **ofadmin1** в подразделение **ofadmins**:



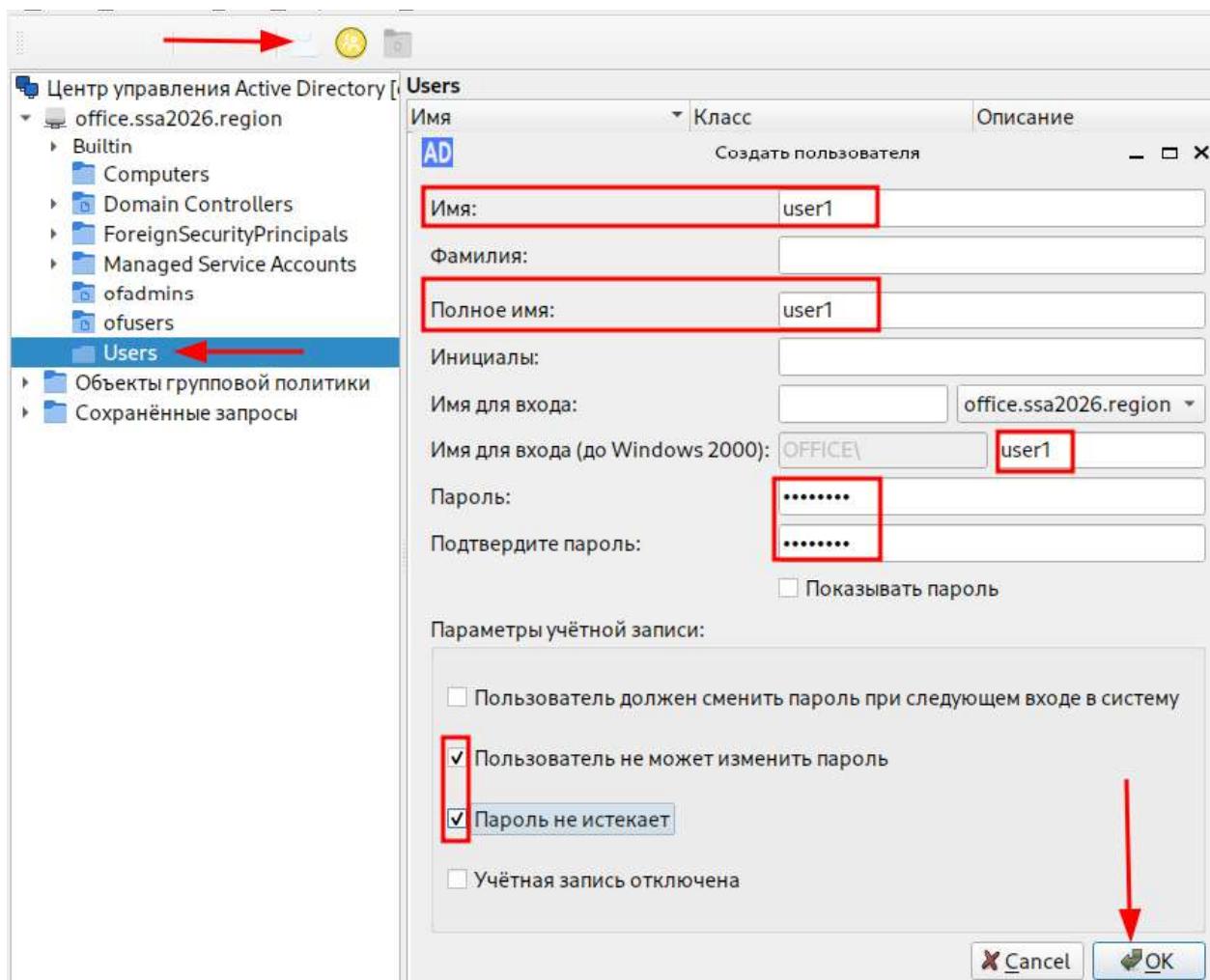
- Добавить созданного пользователя в группу **ofadmins**:



- Создать пользователя **ofuser1** в подразделение **ofusers**
 - Добавить созданного пользователя в группу **ofusers**



- Создать пользователя **user1**:



cli2-a или cli1-a (alt-workstation):

- Проверить возможность входа из-под созданных пользователей:
 - **ofadmin1:**

```

Q          ofadmin1@cli2-a: /home/OFFICE.SSA2026.REGION/ofadmin1 + ☰ ×
[ofadmin1@cli2-a ~]$ id
uid=29801108(ofadmin1) gid=29800513(domain users) группы=29800513(domain users),
14(uucp),19(proc),22(cdrom),71(floppy),80(cdwriter),81(audio),83(radio),100(user
s),951(usershares),959(camera),965(fuse),978(video),988(vboxsf),989(vboxadd),997
(xgrp),998(scanner),29801106(ofadmins)
[ofadmin1@cli2-a ~]$

```

- **ofuser1:**

```
Q          ofuser1@cli2-a: /home/OFFICE.SSA2026.REGION/ofuser1 + 三 x
[ofuser1@cli2-a ~]$ id
uid=29801109(ofuser1) gid=29800513(domain users) группы=29800513(domain users),1
4(uucp),19(proc),22(cdrom),71(floppy),80(cdwriter),81(audio),83(radio),100(users)
,951(usershares),959(camera),965(fuse),978(video),988(vboxsf),989(vboxadd),997(
xgrp),998(scanner),29801107(ofusers)
[ofuser1@cli2-a ~]$
```

- user1:

```
Q          user1@cli2-a /home/OFFICE.SSA2026.REGION/user1 + 三 x
[user1@cli2-a ~]$ id
uid=29801110(user1) gid=29800513(domain users) группы=29800513(domain users),14(
uucp),19(proc),22(cdrom),71(floppy),80(cdwriter),81(audio),83(radio),100(users),
951(usershares),959(camera),965(fuse),978(video),988(vboxsf),989(vboxadd),997(xg
rp),998(scanner)
[user1@cli2-a ~]$
```

cli1-a и cli2-a (alt-workstation):

- Необходимо установить пакет **gpupdate**:

```
apt-get install -y gpupdate
```

- Включить модуль групповых политик:

```
gpupdate-setup enable
```

cli1-a (alt-workstation):

- Установим пакет **grui**, для редактирования настроек клиентской конфигурации:

```
apt-get install -y grui
```

- В оснастке ADMC - переходим в раздел **Объекты групповой политики -> ПКМ по office.ssa2026.region** и нажимаем **Создать политику и связать с этом подразделением**:

Centre d'administration Active Directory [office.ssa2026.region]

- office.ssa2026.region
 - Builtin
 - Computers
 - Domain Controllers
 - ForeignSecurityPrincipals
 - Managed Service Accounts
 - ofadmins
 - ofusers
 - Users
- Objets de la politique de groupe
 - office.ssa2026.region
 - Def
 - Dom
 - ofa
 - ofu
 - Все
- Сохранить

Создать подразделение

Создать политику и связать с этим подразделением

Связать существующую политику

Найти объект групповой политики

Блокировать наследование

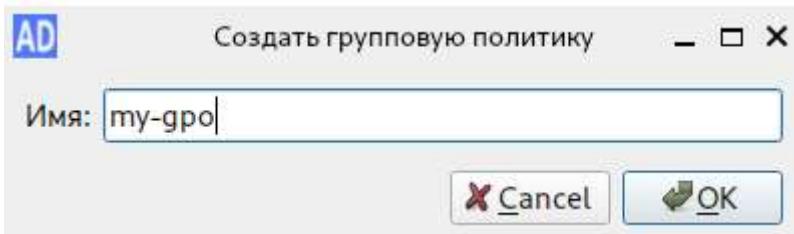
Обновить

Свойства

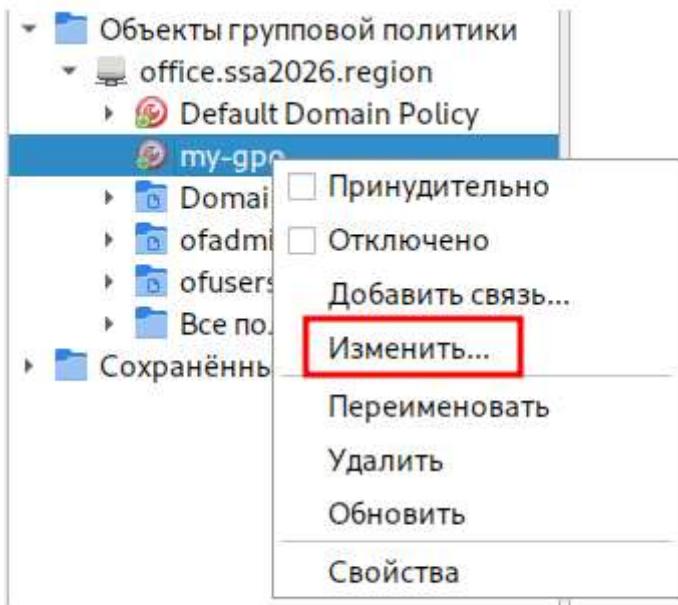
Привязанные политики

1 Default Domain Policy

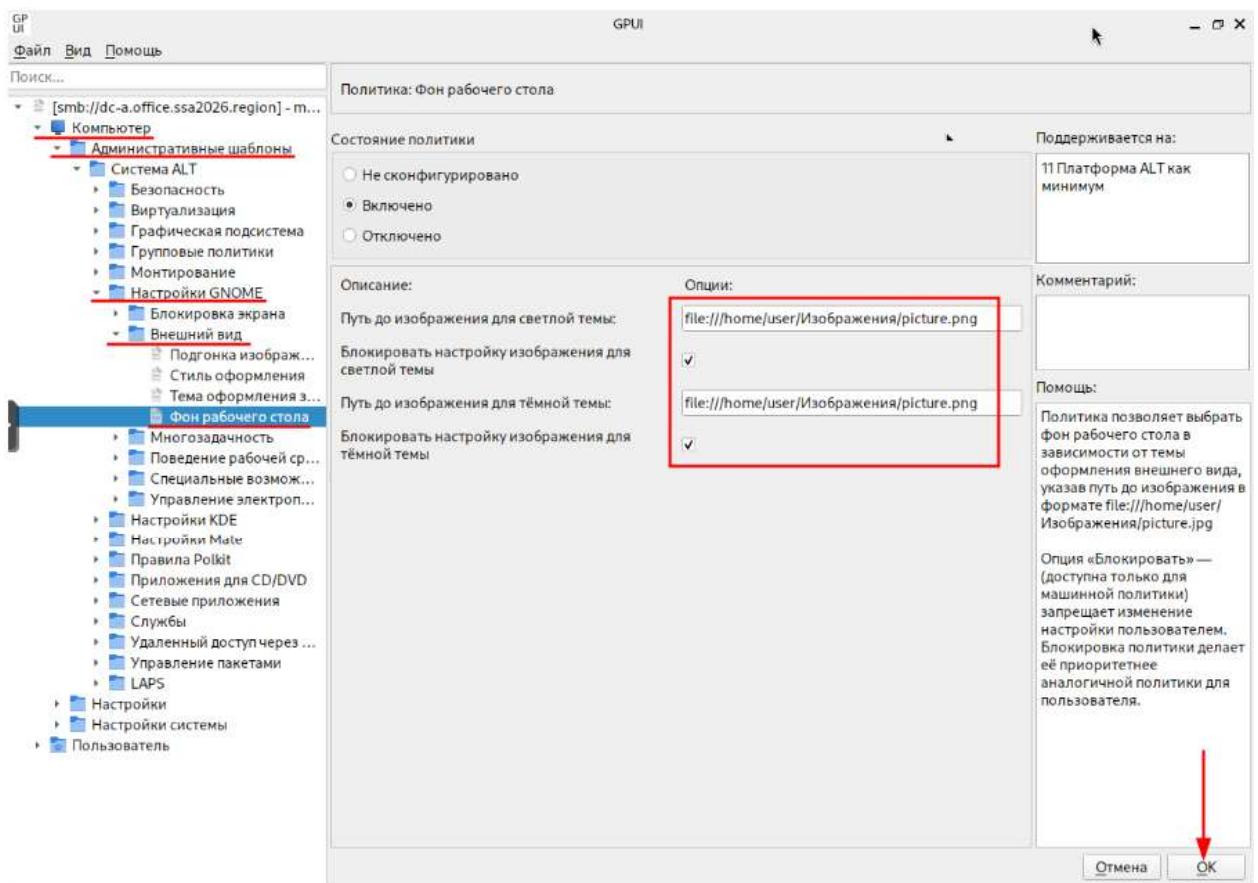
- Задаём имя и нажимаем OK:



- Выбираем созданную групповую политику ПКМ и нажимаем Изменить:

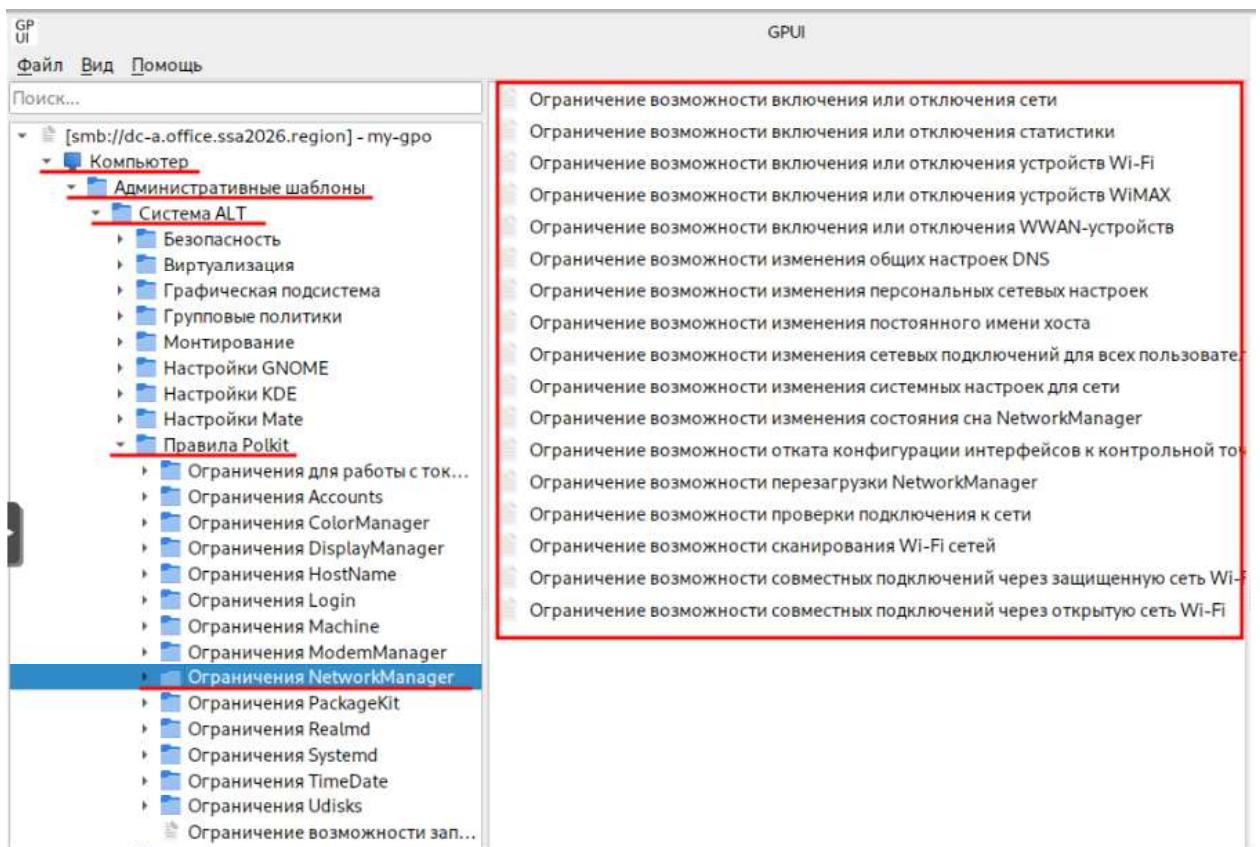


- В соответствие с требованиями задания - реализуем необходимый функционал:
 - Задаём картинку компании для рабочего стола и запрещаем её менять
 - но, чтобы на cli2-a (или же из под любого другого пользователя) картинка также была установлена, её необходимо разместить на общем ресурсе (например, создав общую папку)

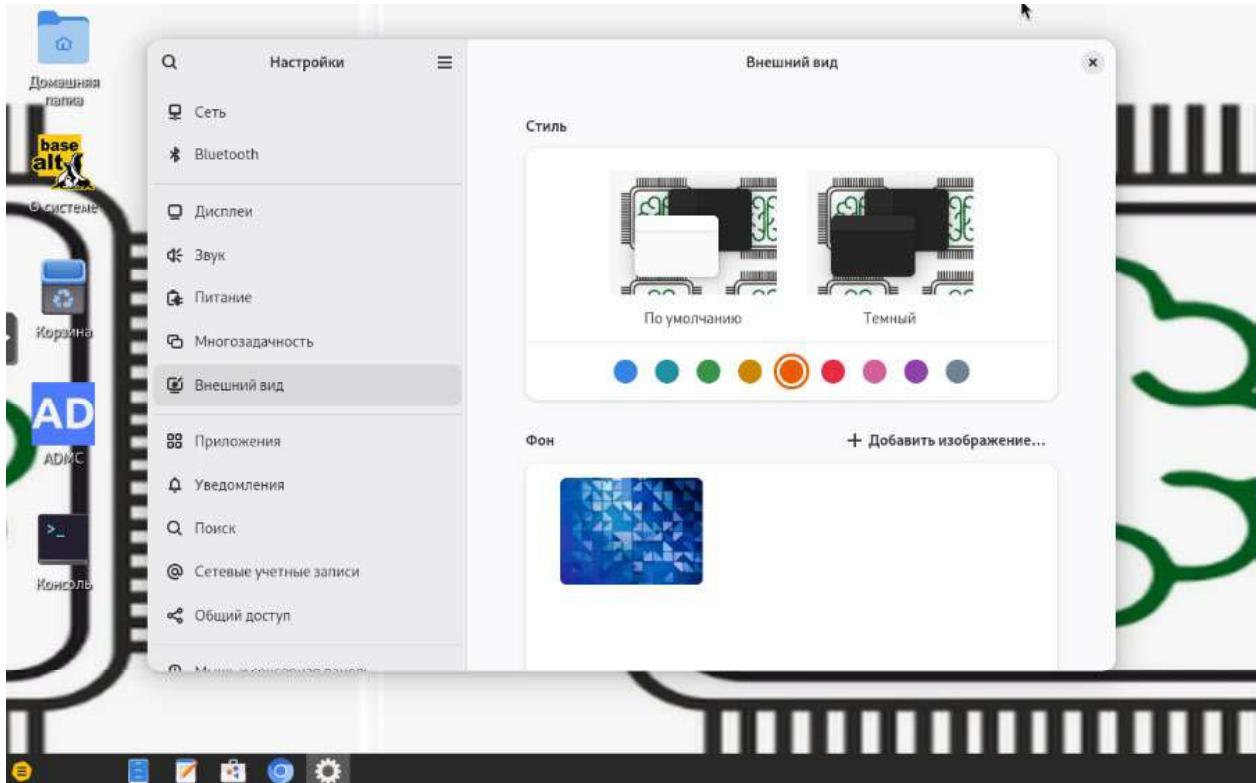


- Запрещаем изменение сетевых настроек:

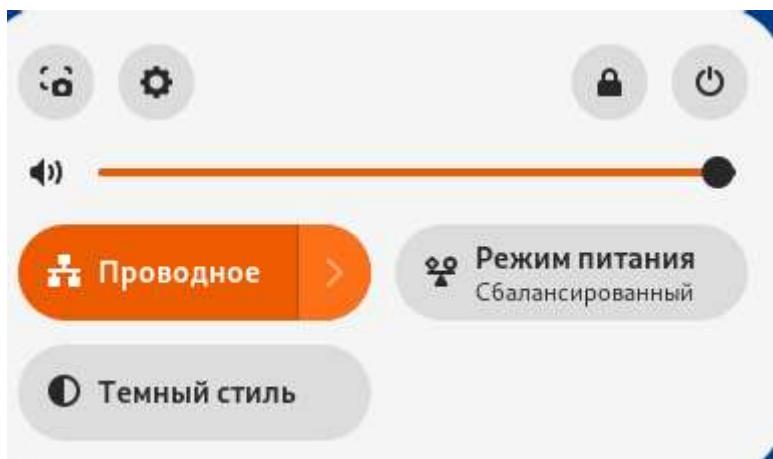
- пройтись по всему списку и выставить **Включено**, с вариантом ограничений **Но** (выставить чек-бокс **Блокировать**):



- Проверить, перезагружаем cli1-а и cli2-а:
 - картинка фона рабочего стола должна быть установлена:



- при попытке отключить сетевой интерфейс, данная возможность отсутствует:



Вариант реализации:

dc-a (alt-server):

- Средствами **samba-tool** можно посмотреть список зон DNS:

```
[root@dc-a ~]# samba-tool dns zonelist 127.0.0.1
Password for [administrator@OFFICE.SSA2026.REGION]:
2 zone(s) found

pszZoneName          : office.ssa2026.region
Flags                : DNS_RPC_ZONE_DSINTEGRATED DNS_RPC_ZONE_UPDATE_SECURE
ZoneType             : DNS_ZONE_TYPE_PRIMARY
Version              : 50
dwDpFlags            : DNS_DP_AUTOCREATED DNS_DP_DOMAIN_DEFAULT DNS_DP_ENLISTED
pszDpFqdn            : DomainDnsZones.office.ssa2026.region

pszZoneName          : _msdcs.office.ssa2026.region
Flags                : DNS_RPC_ZONE_DSINTEGRATED DNS_RPC_ZONE_UPDATE_SECURE
ZoneType             : DNS_ZONE_TYPE_PRIMARY
Version              : 50
dwDpFlags            : DNS_DP_AUTOCREATED DNS_DP_FOREST_DEFAULT DNS_DP_ENLISTED
pszDpFqdn            : ForestDnsZones.office.ssa2026.region
[root@dc-a ~]#
```

- Средствами **samba-tool** можно посмотреть список DNS-записей в определённой зоне:

```
[root@dc-a ~]# samba-tool dns query 127.0.0.1 office.ssa2026.region @ A
Password for [administrator@OFFICE.SSA2026.REGION]:
Name=, Records=1, Children=0
A: 172.20.10.10 (flags=600000f0, serial=38, ttl=900)
Name=_msdcs, Records=0, Children=0
Name=_sites, Records=0, Children=1
Name=_tcp, Records=0, Children=4
Name=_udp, Records=0, Children=2
Name=cli1-a, Records=1, Children=0
A: 172.20.20.1 (flags=f0, serial=39, ttl=3600)
Name=cli2-a, Records=1, Children=0
A: 172.20.20.2 (flags=f0, serial=39, ttl=3600)
Name=dc-a, Records=1, Children=0
A: 172.20.10.10 (flags=f0, serial=1, ttl=900)
Name=domaindnszones, Records=0, Children=2
Name=ForestDnsZones, Records=0, Children=2
[root@dc-a ~]#
```

- Средствами **samba-tool** создадим не достающие записи типа **A** в зоне прямого просмотра:

```
samba-tool dns add 127.0.0.1 office.ssa2026.region rtr-a A 172.20.10.254 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region rtr-a A 172.20.20.254 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region rtr-a A 172.20.30.254 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region sw1-a A 172.20.30.1 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region sw2-a A 172.20.30.2 -U administrator
```

- Проверить наличие записей:

```
[root@dc-a ~]# samba-tool dns query 127.0.0.1 office.ssa2026.region @ A
Password for [administrator@OFFICE.SSA2026.REGION]:
Name=, Records=1, Children=0
  A: 172.20.10.10 (flags=600000f0, serial=38, ttl=900)
Name=_msdcs, Records=0, Children=0
Name=_sites, Records=0, Children=1
Name=_tcp, Records=0, Children=4
Name=_udp, Records=0, Children=2
Name=cli1-a, Records=1, Children=0
  A: 172.20.20.1 (flags=f0, serial=39, ttl=3600)
Name=cli2-a, Records=1, Children=0
  A: 172.20.20.2 (flags=f0, serial=39, ttl=3600)
Name=dc-a, Records=1, Children=0
  A: 172.20.10.10 (flags=f0, serial=1, ttl=900)
Name=DomainDnsZones, Records=0, Children=2
Name=ForestDnsZones, Records=0, Children=2
Name=rtr-a, Records=3, Children=0
  A: 172.20.10.254 (flags=f0, serial=40, ttl=900)
  A: 172.20.20.254 (flags=f0, serial=41, ttl=900)
  A: 172.20.30.254 (flags=f0, serial=42, ttl=900)
Name=sw1-a, Records=1, Children=0
  A: 172.20.30.1 (flags=f0, serial=43, ttl=900)
Name=sw2-a, Records=1, Children=0
  A: 172.20.30.2 (flags=f0, serial=44, ttl=900)
[root@dc-a ~]#
```

- Проверить функционально:

```
[root@dc-a ~]# host rtr-a
rtr-a.office.ssa2026.region has address 172.20.20.254
[root@dc-a ~]# host sw1-a
sw1-a.office.ssa2026.region has address 172.20.30.1
[root@dc-a ~]# host sw2-a
sw2-a.office.ssa2026.region has address 172.20.30.2
[root@dc-a ~]# host dc-a
dc-a.office.ssa2026.region has address 172.20.10.10
[root@dc-a ~]# host cli1-a
cli1-a.office.ssa2026.region has address 172.20.20.1
[root@dc-a ~]# host cli2-a
cli2-a.office.ssa2026.region has address 172.20.20.2
[root@dc-a ~]# _
```

- Добавить в конфигурационный файл /etc/bind/local.conf информацию о файле зоны обратного просмотра и о зонах на srv1.cod:

```

//include "/etc/bind/rfc1912.conf";

// Consider adding the 1918 zones here,
// if they are not used in your organization.
//      include "/etc/bind/rfc1918.conf";

// Add other zones here
zone "20.172.in-addr.arpa" {
    type master;
    file "20.172.in-addr.arpa";
    allow-transfer { 192.168.10.1; };
};

zone "cod.ssa2026.region" {
    type forward;
    forward only;
    forwarders { 192.168.10.1; };
};

zone "168.192.in-addr.arpa" {
    type forward;
    forward only;
    forwarders { 192.168.10.1; };
};

```

- Скопировать файл шаблона для зоны обратного просмотра:

```
cp /etc/bind/zone/localhost /etc/bind/zone/20.172.in-addr.arpa
```

- Выдать права на файл зоны обратного просмотра:

```
chown root:named /etc/bind/zone/20.172.in-addr.arpa
```

- Привести файл **/etc/bind/zone/20.172.in-addr.arpa** зоны обратного просмотра к следующему виду:

\$TTL	TID				
0	IN	SOA	office.ssa2026.region.	root.office.ssa2026.region.	(
			2025110500	:	serial
			12H	:	refresh
			1H	:	retry
			1W	:	expire
			1H	:	ncache
)		
	IN	NS	office.ssa2026.region.		
254.10	IN	PTR	rtr-a.office.ssa2026.region.		
254.20	IN	PTR	rtr-a.office.ssa2026.region.		
254.30	IN	PTR	rtr-a.office.ssa2026.region.		
1.30	IN	PTR	sw1-a.office.ssa2026.region.		
2.30	IN	PTR	sw2-a.office.ssa2026.region.		
10.10	IN	PTR	dc-a.office.ssa2026.region.		
1.20	IN	PTR	cli1-a.office.ssa2026.region.		
2.20	IN	PTR	cli2-a.office.ssa2026.region.		

- Перезапустить службу bind:

```
systemctl restart bind
```

- Проверить записи типа PTR:

```
[root@dc-a ~]# host 172.20.10.254
254.10.20.172.in-addr.arpa domain name pointer rtr-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.20.254
254.20.20.172.in-addr.arpa domain name pointer rtr-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.30.254
254.30.20.172.in-addr.arpa domain name pointer rtr-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.30.1
1.30.20.172.in-addr.arpa domain name pointer sw1-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.30.2
2.30.20.172.in-addr.arpa domain name pointer sw2-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.20.1
1.20.20.172.in-addr.arpa domain name pointer cli1-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.20.2
2.20.20.172.in-addr.arpa domain name pointer cli2-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.10.10
10.10.20.172.in-addr.arpa domain name pointer dc-a.office.ssa2026.region.
[root@dc-a ~]# -
```

- Проверить несколько записей для forward зон:

```
[root@dc-a ~]# host 192.168.10.1
1.10.168.192.in-addr.arpa domain name pointer srv1-cod.cod.ssa2026.region.
[root@dc-a ~]# host 192.168.10.2
2.10.168.192.in-addr.arpa domain name pointer srv2-cod.cod.ssa2026.region.
[root@dc-a ~]# -
```

sw1-a и sw2-a (alt-server):

- Задаём в качестве DNS-сервера dc-a:

```
cat <<EOF > /etc/net/ifaces/mgmt/resolv.conf
search office.ssa2026.region
nameserver 172.20.10.10
EOF
```

- Перезагружаем службу network:

```
systemctl restart network
```

- Проверить:

```
[root@sw1-a ~]# cat /etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search office.ssa2026.region
nameserver 172.20.10.10
[root@sw1-a ~]# ping -c3 office.ssa2026.region
PING office.ssa2026.region (172.20.10.10) 56(84) bytes of data.
64 bytes from dc-a.office.ssa2026.region (172.20.10.10): icmp_seq=1 ttl=63 time=77.2 ms
64 bytes from dc-a.office.ssa2026.region (172.20.10.10): icmp_seq=2 ttl=63 time=78.0 ms
64 bytes from dc-a.office.ssa2026.region (172.20.10.10): icmp_seq=3 ttl=63 time=73.3 ms
--- office.ssa2026.region ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 73.257/76.143/77.987/2.066 ms
[root@sw1-a ~]# ping -c3 ya.ru
PING ya.ru (77.88.55.242) 56(84) bytes of data.
64 bytes from ya.ru (77.88.55.242): icmp_seq=1 ttl=48 time=97.3 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=2 ttl=48 time=95.8 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=3 ttl=48 time=92.7 ms
--- ya.ru ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 92.658/95.265/97.336/1.947 ms
[root@sw1-a ~]# -
```

VM	Network Device	Network	VLAN	IP	Gateway
rtr-cod	net1	178.207.179.0/29	-	178.207.179.4	178.207.179.1
	net2	172.16.1.0/30		172.16.1.1	
fw-cod	net0	172.16.1.0/30	-	172.16.1.2	172.16.1.1
	net1	192.168.10.0/24		192.168.10.254	
		192.168.20.0/24		192.168.20.254	
		192.168.30.0/24		192.168.30.254	
		192.168.40.0/24		192.168.40.254	
		192.168.50.0/24		192.168.50.254	
sw1-cod	SVI	192.168.30.0/24	300	192.168.30.1	192.168.30.254
sw2-cod	SVI	192.168.30.0/24	300	192.168.30.2	192.168.30.254
cli-cod	net0	192.168.40.0/24	400	192.168.40.40	192.168.40.254
srv1-cod	net0	192.168.10.0/24	100	192.168.10.1	192.168.10.254
	net1	192.168.20.0/24	200	192.168.20.1	
srv2-cod	net0	192.168.10.0/24	100	192.168.10.2	192.168.10.254
	net1	192.168.20.0/24	200	192.168.20.2	
sip-cod	net0	192.168.50.0/24	500	192.168.50.50	192.168.50.254
admin-cod	net0	192.168.30.0/24	300	192.168.30.30	192.168.30.254
rtr-a	net1	178.207.179.24/29	-	178.207.179.28	178.207.179.25
	net2	172.20.10.0/24	100	172.20.10.254	
		172.20.20.0/24	200	172.20.20.254	
		172.20.30.0/24	300	172.20.30.254	
sw1-a	SVI	172.20.30.0/24	300	172.20.30.1	172.20.30.254
sw2-a	SVI	172.20.30.0/24	300	172.20.30.2	172.20.30.254
dc-a	net0	172.20.10.0/24	100	172.20.10.10	172.20.10.254
cli1-a	net0	172.20.20.0/24	200	172.20.20.1	172.20.20.254
cli2-a	net0	172.20.20.0/24	200	172.20.20.2	172.20.20.254