Автономная некоммерческая профессиональная образовательная организация
«Хекслет колледж»

Допустить к защите:
Зам. директора по учебной и
производственной работе
_____А. К.Бесстрашнова

«___» _____2025 г.

ДИПЛОМНЫЙ ПРОЕКТ

Тема: Проектирование системы резервного копирования и аварийного восстановления данных на Linux

Студент группы 3102-д _____ А.А. Алавердян
№ группы подпись И.О. Фамилия

Специальность 09.02.06 Сетевое и системное администрирование

Руководитель _____ М.С. Караченцева
подпись И.О. Фамилия

Санкт-Петербург – 2025 год

С О Д Е Р Ж А Н И Е

В В Е Д Е Н И Е

Системы резервного копирования и аварийного восстановления данных представляют собой критически важный компонент информационной инфраструктуры любой организации. Как отмечают эксперты, резервное копирование - это последняя линия защиты данных от потери в случае аппаратных сбоев, человеческих ошибок или кибератак. Кроме того, грамотно спроектированная система аварийного восстановления позволяет минимизировать время простоя бизнес-процессов и обеспечить непрерывность работы организации. Проектирование таких систем в среде Linux требует комплексного подхода, учитывающего особенности файловых систем, сетевых протоколов и механизмов хранения данных.

Актуальность темы дипломного проекта обусловлена стремительным ростом объемов корпоративных данных и ужесточением требований к их сохранности. В условиях участившихся случаев ransomware-атак, аппаратных отказов и человеческих ошибок надежная система резервирования становится не просто рекомендацией, а обязательным элементом ИТ-инфраструктуры. Linux-серверы, благодаря своей стабильности, безопасности и гибкости, часто выступают платформой для построения таких систем, что делает изучение методов их проектирования и оптимизации особенно актуальным для современных ИТ-специалистов.

Целью данного дипломного проекта является разработка и реализация системы резервного копирования и аварийного восстановления данных на платформе Linux, включая анализ эффективности различных стратегий бэкапа и методов восстановления информации.

Для достижения поставленной цели были определены следующие задачи:

1. Дать понятие Linux.
2. Изучить системы резервного копирования и аварийного восстановления данных.

3. Разработать сеть, внутри которой будет проводиться настройка систем.
4. Реализовать системы резервного копирования и аварийного восстановления данных в смоделированной сети.

Объектом дипломного проекта является процесс проектирования и реализации систем резервного копирования в операционной системе Linux.

Предметом дипломного проекта являются методы организации бэкапов (полных, инкрементальных, дифференциальных), технологии хранения резервных копий (локальные, сетевые, облачные), а также механизмы аварийного восстановления данных с минимальным временем простоя.

Источниками информации для дипломного проекта являются материалы, полученные в ходе изучения темы дипломного проекта, а также материалы сайтов, посвященные:

– Linux;

– Система резервного копирования;

– Система аварийного восстановления данных;

– Системы резервного копирования и аварийного восстановления данных в ОС Linux.

Дипломный проект состоит из введения, двух глав, заключения, списка использованных источников.

В первой главе рассматриваются понятия по теме, такие как:

“ОС Linux”;

“Система резервного копирования”;

“Система аварийного восстановления данных”;

А также информация по проектированию систем резервного копирования и аварийного восстановления данных на ОС Linux.

Во второй главе была смоделирована локальная сеть, которая впоследствии была налажена следуя имеющейся топологии, а также настроены системы резервного копирования и аварийного восстановления данных.

1 СИСТЕМЫ РЕЗЕРВНОГО КОПИРОВАНИЯ И АВАРИЙНОГО ВОССТАНОВЛЕНИЯ ДАННЫХ НА LINUX

1.1 Понятие Linux

Linux — это свободная операционная система (точнее, семейство систем) с открытым исходным кодом. Само название Linux относится к ядру системы, которое является ее ключевым компонентом, управляющим центральным процессором (ЦП), памятью и периферийными устройствами компьютера. Технически Linux состоит из следующих компонентов:

– загрузчик — ПО, управляющее процессом загрузки компьютера;

– ядро — сердце системы, которое управляет всеми компонентами ОС и компьютера;

– демоны — служебные программы, которые работают в фоновом режиме, например управляют звуками;

– система инициализации — подсистема, которая управляет учетной записью пользователя и демонами;

– графический сервер — подсистема, которая управляет отображением графической информации на мониторе;

– среда рабочего стола — пользовательский интерфейс;

– приложения — пользовательские программы, выполняющие различные функции, например текстовые процессоры, интернет-браузеры, музыкальные проигрыватели.

В отличие от других ОС, Linux имеет довольно аскетичный интерфейс, лишенный всех функций более ориентированных на пользователя систем, таких как MacOS. При этом Linux — очень функциональная система, которая дает пользователю большую степень свободы в управлении программным и аппаратным обеспечением компьютера. Важнейшим преимуществом Linux является открытый исходный код, значительно расширяющий возможности персонализации и управления, а также многочисленное сообщество

пользователей, которые поддерживают работу этой операционной системы. Большинство ОС имеют одну основную версию, называемую дистрибутивом, которая выступает в качестве пользовательского интерфейса. Linux же имеет большое количество различных дистрибутивов. Наиболее популярными из них являются Ubuntu (для рядовых пользователей), Suse, Redhat (для корпоративных серверов) и Cent OS (для облачных платформ). Эти дистрибутивы специально разработаны под различные нужды пользователей. Кроме того, операционная система Android, на которой работают многие модели смартфонов, тоже построена на базе ядра Linux. ОС Linux подходит для использования в следующих целях.

– Для веб-серверов: может использоваться для высокопроизводительных серверов любого назначения.

– Для настольных компьютеров: подходит для рядовых пользователей, использующих традиционную среду рабочего стола.

– Для безголовых систем управления: идеальна для серверов удаленного доступа, которым не требуется графический интерфейс пользователя.

– Для встраиваемых систем: подходит для устройств с простейшими вычислительными функциями, например для домашнего оборудования.

– Для сетей: может использоваться для любых сетевых нужд.

– Для разработки ПО: лучшее решение для разработки корпоративного программного обеспечения.

– Для облачных сред: подходит для масштабных облачных вычислений.

Поскольку операционная система Linux устроена сложнее, чем Windows и MacOS, и предлагает гораздо больше возможностей для персонализации, она широко используется техническими специалистами. Кроме того, Linux часто используют для веб-серверов и центров управления сетью, где к программному обеспечению предъявляются особые требования. Для пользователей настольных компьютеров существует Linux Mint —

безопасный дистрибутив, оснащенный всеми необходимыми инструментами для офисной работы, графического дизайна, воспроизведения мультимедиа, поиска в интернете и игр.

Хотя Linux в основном используется технологически продвинутыми пользователями, она является такой же операционной системой, как более популярные Windows и MacOS, и обладает похожим набором функций. Например, Linux имеет графический интерфейс пользователя и собственные версии часто используемых программ, таких как фоторедакторы, электронные таблицы и почтовые клиенты. Linux можно установить на любое электронное устройство, от компьютера до смартфона. Однако у этой операционной системы есть и отличия. Именно поэтому Linux может быть использована в иных целях, чем Windows и MacOS. В то время как большинство операционных систем представляют собой замкнутую рабочую среду, Linux – это ПО с открытым исходным кодом. Это значит, что любой может видеть и редактировать исходный код, а следовательно, вносить изменения в систему. Разумеется, это оказывает определенное влияние на безопасность системы, о чем будет сказано ниже. Еще одно отличие заключается в том, что прочие операционные системы, как правило, имеют только одну комплектацию. Например, существует только одна разновидность MacOS или Windows (в рамках этой статьи мы не принимаем во внимание различные версии ОС, такие как MacOS Monterey или MacOS Sierra, поскольку это просто разные варианты одного и того же ПО). В то время как Linux имеет широкую линейку дистрибутивов и прикладных программ, что делает ее крайне гибкой. Пользователи могут не только выбирать, какое ПО должно быть установлено в их системе, но и персонализировать ключевые компоненты системы, такие как графический и пользовательский интерфейс.

Для защиты Linux все пользователи системы должны иметь уникальное имя и пароль. Кроме того, в этой ОС существуют различные уровни доступа. Например, суперпользователь (уровень root) обладает правами системного администратора. По умолчанию система назначает пользователям самый

низкий уровень, который ограничивает их доступ к файлам. Это делается для того, чтобы затруднить распространение вредоносных программ и тем самым обеспечить безопасность операционной системы. Если вредоносное ПО проникает в компьютер, работающий на Linux, оно не может получить доступ суперпользователя и нанести ущерб всей системе. Кроме того, все пользователи в системе изолированы, что ограничивает возможность перекрестного заражения вредоносным ПО.

Система Linux по умолчанию присваивает пользователям самый низкий уровень доступа, не позволяющий выполнять операции на системном уровне. Система Windows при установке создает учетную запись администратора, которая позволяет любому пользователю получить права администратора с помощью опции «Run as Administrator». Пользователи Windows могут изменить эту функцию в настройках системы, но часто не делают этого. Это дает Linux системное преимущество.

Пользователи Windows могут устанавливать любое ПО из интернета, просто загрузив и открыв файл с расширением .exe или .msi. Если нельзя убедиться в надежности источника, это может представлять серьезный риск для безопасности. Система защиты Linux сама управляет установкой ПО с помощью менеджера пакетов, который позволяет загружать программы только из репозиториев— доверенных источников, управляемых членами сообщества, которые проверяют ПО.

Сегодня сложно представить полноценную деятельность во многих современных сферах без использования Линукс. Его активно применяют в медицине, робототехнике, машиностроении, финансовой и платежной сфере и других важных направлениях. Операционка отлично работает как на домашних, так и на корпоративных компьютерах, включая устаревшие модели техники. Среди известных организаций, применяющих эту ОС в своей деятельности, можно выделить такие гиганты, как Google и NASA. Рассмотрим подробнее ключевые отрасли, где на основе Linux успешно решаются самые сложные задачи.

– Веб-серверы. Эта операционная система занимает лидирующие позиции и является общепризнанным стандартом в сфере серверного администрирования благодаря высокой стабильности, безопасности и широким функциональным возможностям. Согласно исследованиям аналитических агентств, под её управлением функционирует свыше 96% наиболее производительных веб-серверов в мире, а также размещено около 75,1% всех сайтов интернета.

– Мобильные устройства. Android основан на Линуксе, благодаря чему более 85% современных смартфонов функционируют на ее основе. Помимо мобильных телефонов, её используют в фитнес-браслетах, большинстве планшетов, электронных книгах, телевизорах с функцией Smart TV, в игровых консолях и множестве других цифровых решений, работающих под управлением Android. Такое широкое применение подчеркивает гибкость и универсальность данной технологии.

– Суперкомпьютеры. Ключевое отличие этих машин от обычных компьютеров — в колоссальной производительности. Сверхмощные комплексы способны за доли секунды выполнять миллионы ресурсоемких и высокоточных вычислений. Для управления такими процессами требуется специализированная операционная среда, адаптированная под конкретные задачи. Часто для этого используется операционная система Линукс, так как благодаря открытому исходному коду специалисты, работающие с суперкомпьютерами, могут настраивать ее под собственные нужды, модифицируя функциональность и архитектуру в соответствии с проектными требованиями.

– Игровые консоли. Проектов, изначально ориентированных на Linux, пока немного. Эту ситуацию пытается изменить компания Steam — известная цифровая площадка для распространения видеоигр. Она разрабатывает собственную ОС SteamOS на базе этой ОС, предназначенную для использования в игровых консолях Steam Machine. Американская компания Valve Corporation, специализирующаяся на разработке и издании

компьютерных игр, создала свою игровую приставку и уникальную программную платформу на базе открытого кода в сфере цифровых развлечений.

– Авиация и транспорт. В авто пилотируемых автомобилях Google и в бортовых системах машин Tesla используется ОС Linux. С 2006 года на её основе работают и большинство средств отслеживания воздушного трафика в США. Однако в 2023 году специалисты отметили, что в текущей реализации эта технология не соответствует строгим требованиям, предъявляемым к программному обеспечению авионики, необходимому для эксплуатации в вертолётах и самолётах.

А также:

– Для домашнего ежедневного использования (например, Ubuntu);

– Для реанимации старого железа (например, поставить Calculate для слабых компьютеров в школьных классах);

– Для отказоустойчивых станций для работы в бесперебойном режиме;

– Для систем безопасности и шифрования;

– Для создания сети из компьютеров для параллельных вычислений;

– Для обслуживания сигнализаций, умных домов и районов;

– Для роутеров и прочего компьютерного железа;

– Для роботов и робототехники.

Чтобы выбрать подходящую сборку, требуется сделать осознанный выбор, стоит опираться на несколько ключевых критериев:

– какие задачи предстоит выполнять;

– характеристики и возможности оборудования;

– насколько важны регулярные обновления и техническая поддержка;

– нужен ли интуитивно понятный и лёгкий в освоении интерфейс;

– поддерживает ли система необходимые для тебя приложения;

– есть ли активное сообщество, к которому можно обратиться за помощью.

Для пользователей с разным уровнем подготовки подходят разные варианты. Новичкам подойдут сборки, ориентированные на простоту установки и дружелюбный интерфейс. Более опытным стоит обратить внимание на решения с широкими возможностями настройки. А профессионалам — на мощные платформы, предназначенные для специализированных задач и тонкой конфигурации.

Также при выборе сборки, следует учитывать, с каким интерфейсом пользователь собирается работать. Можно выделить 4 интерфейса для работы на Linux:

– Оконный менеджер KDE (рисунок 1).

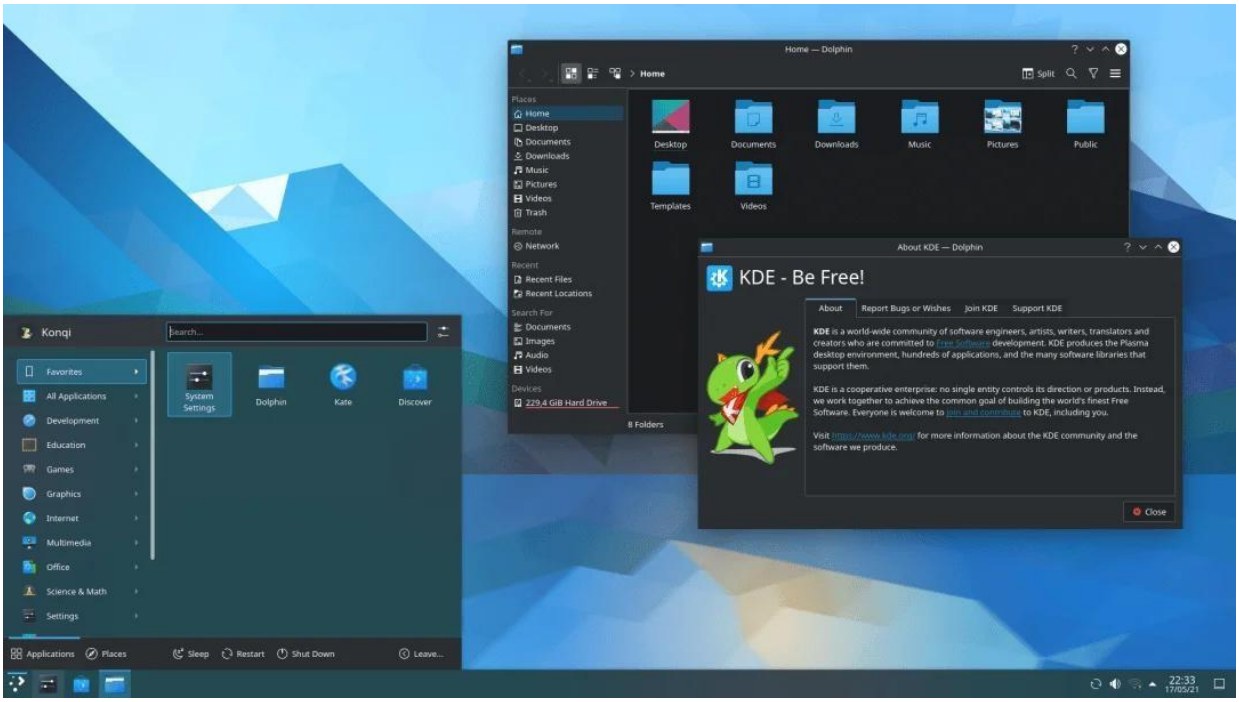

Рисунок 1. Оконный менеджер KDE.

– GNOME (рисунок 2).

Рисунок 2. GNOME.

– Менеджер Xfce (рисунок 3).

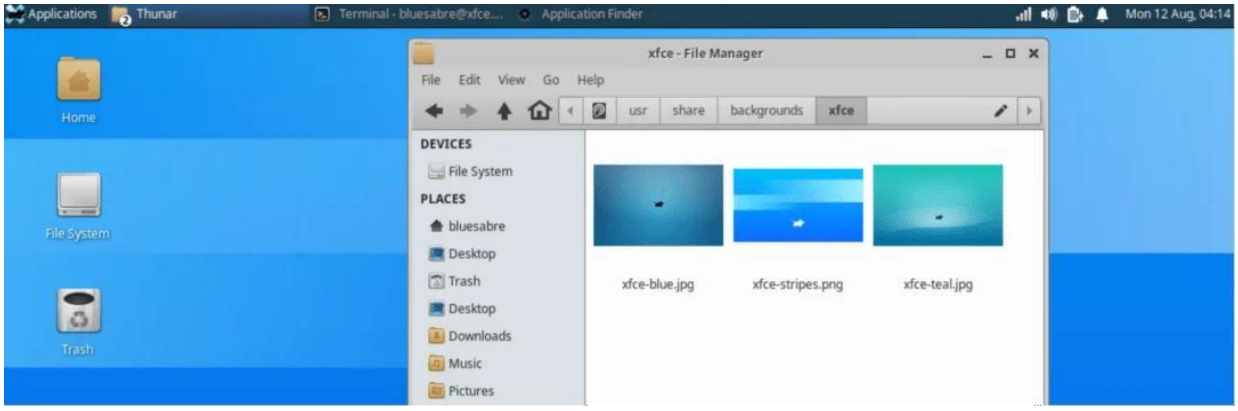

Рисунок 3. Менеджер Xfce.

– Командная строка (рисунок 4).

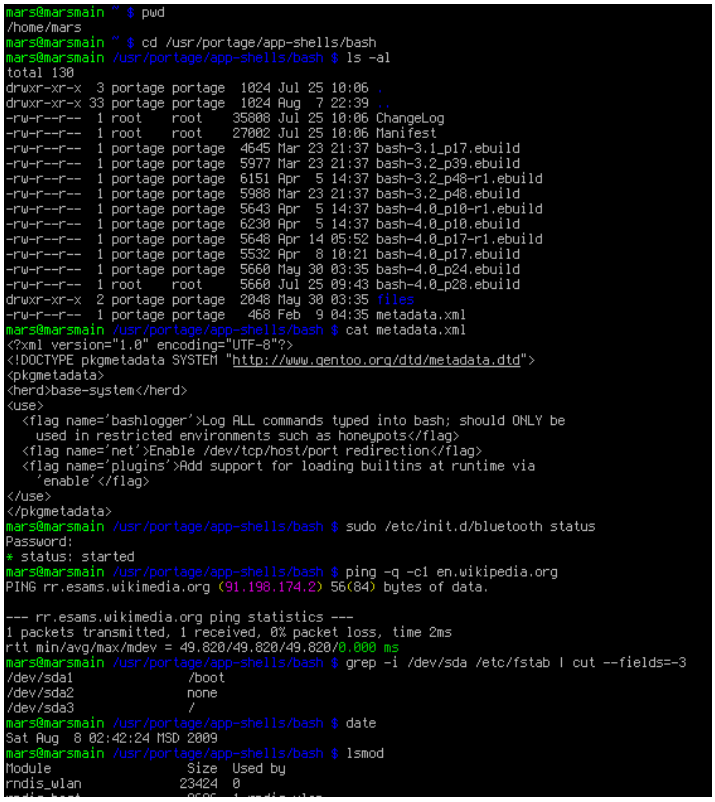

Рисунок 4. Командная строка.

1.2 Понятие системы резервного копирования

Система резервного копирования — совокупность программного и аппаратного обеспечения, выполняющее задачу создания копии данных на носителе, предназначенном для восстановления информации в оригинальном месте их расположения в случае их повреждения или разрушения. Системы резервного копирования обеспечивают непрерывность бизнес-процессов и защиту информации от природных и техногенных катастроф, действий злоумышленников. Эти технологии активно используются в ИТ-инфраструктурах организаций самых разных отраслей и масштабов.

Резервное копирование данных — процесс создания копии данных на носителе, предназначенном для восстановления данных в оригинальном месте их расположения в случае их повреждения или разрушения. Кроме того, система резервного копирования — это один из необходимых методов обеспечения непрерывности бизнеса. Построение централизованной системы

резервного копирования позволяет сократить совокупную стоимость владения ИТ-инфраструктурой благодаря оптимальному использованию устройств резервного копирования и сокращению расходов на администрирование (по сравнению с децентрализованной системой).

Архитектура системы резервного копирования, а также её работа:

Централизованная система резервного копирования имеет многоуровневую архитектуру, в которую входят:

– сервер управления резервным копированием, способный также совмещать функции сервера копирования данных;

– один или несколько серверов копирования данных, к которым подключены устройства резервного копирования;

– компьютеры-клиенты с установленными на них программами-агентами резервного копирования;

– консоль администратора системы резервного копирования.

Администратор системы ведет список компьютеров-клиентов резервного копирования, устройств записи и носителей хранения резервных данных, а также составляет расписание резервного копирования. Вся эта информация содержится в специальной базе, которая хранится на сервере управления резервным копированием. В соответствии с расписанием или по команде оператора сервер управления дает команду программе-агенту, установленной на компьютере-клиенте, начать резервное копирование данных в соответствии с выбранной политикой. Программа-агент собирает и передает данные, подлежащие резервированию, на сервер копирования, указанный ей сервером управления. Сервер копирования сохраняет полученные данные на подключенное к нему устройство хранения данных. Информация о процессе (какие файлы копировались, на какие носители осуществлялось копирование и т. п.) сохраняется в базе сервера управления. Эта информация позволяет найти местоположение сохраненных данных при необходимости их восстановления на компьютере-клиенте.

Чтобы система резервного копирования сохраняла непротиворечивые данные компьютера-клиента, они не должны подвергаться изменениям в процессе их сбора и копирования программой-агентом. Для этого приложения компьютера-клиента должны завершить все транзакции, сохранить содержимое кэш-памяти на диск и приостановить свою работу. Этот процесс инициируется по команде программы-агента, которая передается приложениям компьютера-клиента. Поскольку система резервного копирования предназначена для восстановления данных после сбоя или аварии, созданные резервные копии необходимо проверять на предмет целостности и работоспособности. Кроме того, при построении системы резервного копирования необходимо уложиться в сокращенное «окно» резервного копирования. Вообще говоря, требование круглосуточной работы информационных систем сокращает практически до нуля доступный временной интервал остановки приложений, необходимый для осуществления операции резервного копирования («окно» резервного копирования). Как правило, резервное копирование происходит автоматически. Для доступа к данным обычно требуются повышенные привилегии. Так что процесс, обеспечивающий резервное копирование, запускается из-под учетной записи с повышенными привилегиями — вот тут-то и закрадывается определенный риск.

Существует 2 классификации резервного копирования:

1. По полноте сохраняемой информации:

– Полное резервирование (Full backup) — создание резервного архива всех системных файлов, обычно включающего состояние системы, реестр и другую информацию, необходимую для полного восстановления рабочих станций. То есть резервируются не только файлы, но и вся информация, необходимая для работы системы.

– Добавочное резервирование (Incremental backup) — создание резервного архива из всех файлов, которые были модифицированы после предыдущего полного или добавочного резервирования.

– Разностное резервирование (Differential backup) — создание резервного архива из всех файлов, которые были изменены после предыдущего полного резервирования.

– Выборочное резервирование (Selective backup) — создание резервного архива только из отобранных файлов.

2. По способу доступа к носителю:

– Оперативное резервирование (Online backup) — создание резервного архива на постоянно подключенном (напрямую или через сеть) носителе.

– Автономное резервирование (Offline backup) — хранение резервной копии на съемном носителе, кассете или картридже, который перед использованием следует установить в привод.

Технологии, используемые системами резервного копирования:

От ошибок, в результате которых изменяются или удаляются данные и в которых виноваты операционная система или человек, не защищают ни RAID, ни кластер, ни любая другая технология обеспечения отказоустойчивости. Резервное копирование — одно из оптимальных решений для таких ситуаций, так как оно позволяет хранить копии разного срока давности, например за каждый день текущей недели, двух недельной, месячной, полугодовой и годовой давности. Возможность использовать внешние съемные носители существенно снижает затраты на хранение информации, однако для некоторых задач больше подходят альтернативные технологии.

– Резервное копирование с использованием SAN. Применение Сеть хранения данных SAN позволяет полностью перенести трафик резервного копирования с локальной сети на сеть хранения. Существует два варианта реализации: без загрузки локальной сети, или вне сетевое копирование (LAN-

free backup), и без участия сервера, или вне серверное копирование (Server-free backup).

– Вне Сетевое копирование. При несетевом копировании данные с диска на ленту и обратно передаются внутри SAN. Исключение сетевого сегмента из пути резервного копирования данных позволяет избежать излишних задержек на передачу трафика через сеть IP и платы ввода-вывода. Нагрузка локальной сети падает, и резервное копирование можно проводить практически в любое время суток. Однако пересылку данных выполняет сервер, подключенный к SAN, что увеличивает нагрузку на него. Благодаря протоколу Fibre Channel с помощью одного оптического кабеля может быть организовано несколько каналов передачи данных. При этом весь объем резервируемых данных с backup-серверов хранения направляется на ленточное устройство, минуя локальную сеть. В этом случае локальная сеть необходима лишь для контроля работы самих backup-серверов со стороны главных серверов. Таким образом, только небольшой объем метаданных, которые содержат информацию о резервируемых данных, передается по локальной сети. Главные серверы отвечают в целом за политику резервного копирования данных в своем сегменте или зоне ответственности. Все backup-серверы по отношению к главному серверу являются клиентами. Считается, что рассматриваемый метод резервного копирования может максимально задействовать пиковую полосу пропускания Fibre Channel. В качестве протокола, применяемого для передачи данных между серверами и библиотеками, могут использоваться как SCSI поверх Fibre Channel, так и IP поверх Fibre Channel, тем более что большинство FC-адаптеров и FC-концентраторов работают одновременно с обоими протоколами (IP и SCSI) на одном Fibre Channel-канале.

– Вне Серверное копирование. Данный тип резервного копирования представляет собой дальнейшее развитие метода вне сетевого копирования (LAN-free), поскольку уменьшает количество процессоров, памяти, устройств ввода-вывода, задействованных в этом процессе. Данный процесс архивирует

разделы целиком, в отличие от по файлового архивирования, но при этом позволяет восстанавливать отдельные файлы. По определению, при вне-серверном копировании данные копируются с диска на ленту и обратно без прямого участия сервера. Поскольку для резервного копирования требуется наличие некоторого дополнительного третьего узла, полностью отвечающего за процесс копирования, то отсюда происходит и другое название этого подхода — копирование с участием третьей стороны (Third_-Party Copy, 3PC). Так, в качестве подобного оборудования может использоваться маршрутизатор хранилищ данных, который берет на себя функции, ранее выполнявшиеся сервером. Одно из преимуществ архитектуры SAN — отсутствие жесткой привязки составляющих ее систем к каким-либо устройствам хранения данных. Это свойство и заложено в основу технологии резервного копирования без участия сервера. В данном случае к дисковому массиву может иметь прямой доступ как сервер данных, так и устройства, принимающие участие в копировании с дисковых массивов. Резервному копированию блоков данных, относящихся к какому-либо файлу, предшествует создание некоего индекса или списка номеров принадлежащих ему блоков. Это и позволяет в дальнейшем привлечь внешние устройства для резервного копирования. Таким образом, вне серверное копирование позволяет напрямую перемещать данные между подключенными к сети SAN дисковыми массивами и библиотеками. При этом данные перемещаются по сети SAN и не загружают ни локальную сеть, ни серверы. Такое копирование считается идеальным для корпоративных сетей, которые должны функционировать в непрерывном режиме 24 часа в сутки, 7 дней в неделю. Особенно для тех, для которых временной период, в течение которого можно выполнять резервное копирование без существенного влияния на работу пользователей и приложений, становится недопустимо малым.

– Репликация данных. Современные дисковые массивы обладают средствами создания копий данных внутри самого массива. Данные, созданные этими средствами, носят название Point-In-Time (PIT)-копий, т. е.

фиксированных на определенный момент времени. Существует две разновидности средств создания PIT-копий: клонирование и «моментальный снимок» (snapshot). Под клонированием обычно понимают полное копирование данных. Для него требуется столько же дискового пространства, как и для исходных данных, и некоторое время. При использовании такой копии нет нагрузки на дисковые тома, содержащие исходные данные. Иными словами, нет дополнительной нагрузки на дисковую подсистему продуктивного сервера. Механизм работы «моментальных снимков» иной и может быть реализован как программно на продуктивном сервере, так и аппаратно внутри массива. В момент, когда необходимо начать резервное копирование, программа-агент дает команду приложению завершить все транзакции и сохранить кэш-память на диск. Затем создается виртуальная структура — snapshot, представляющая собой карту расположения блоков данных, которую ОС и другое ПО воспринимает как логический том. Приложение прерывает стандартный режим работы на короткое время, необходимое для сохранения данных. После этого приложение продолжает работать в стандартном режиме и изменять блоки данных, при этом перед изменением старые данные блока с помощью драйвера snapshot копируются в область кэш-памяти snapshot и в карте расположения блоков данных указывается ссылка на новое местоположение блока. Таким образом, карта snapshot всегда указывает на блоки данных, полученные на момент завершения транзакций приложением. Блоки данных, которые не были изменены, хранятся на прежнем месте, а старые данные измененных блоков — в области кэш-памяти snapshot. Программа-агент копирует непротиворечивые данные, полученные на момент завершения транзакций приложением, осуществляя доступ к ним через драйвер snapshot, т. е. используя карту расположения блоков. Создание копий с помощью «моментальных снимков» экономит дисковое пространство, но создает дополнительную нагрузку на дисковую подсистему продуктивного сервера. Какой из методов создания PIT-

копий выбрать, решается на этапе проектирования системы резервного копирования, исходя из бизнес-требований, предъявляемых к системе.

1.3 Понятие системы аварийного восстановления данных

Disaster Recovery (аварийное восстановление) — это сервис восстановления ИТ-систем и данных после сбоя любого уровня. Как правило, предлагается облачными провайдерами как отдельная услуга или включается в состав крупного решения. Условно можно разбить на три составляющие: резервная площадка, программные решения и план восстановления.

Ключевые параметры аварийного восстановления данных:

У решений Disaster Recovery есть два основных параметра, которые влияют на стоимость катастрофоустойчивой системы и размер ущерба в случае инцидента: RTO и RPO.

– RTO (recovery time objective) — период времени, за который ИТ-система должна восстановиться. Если RTO составляет четыре часа, то инфраструктура заработает не позже, чем за этот срок. Если RTO несколько секунд, то пользователи могут даже не заметить, что система «падала». Часть решений аварийного восстановления поддерживают автоматическое переключение трафика на резервную инфраструктуру. Это позволяет нивелировать последствия аварии, сделав их незаметными для пользователей. Длительность RTO определяется потребностями бизнеса. Например, сайту с маленьким трафиком большой RTO не повредит, а для крупного онлайн-магазина 2-3 часа RTO — это серьёзные убытки.

– RPO (recovery point objective) — период времени, за который могут быть утеряны данные в результате аварии. Заявленные три часа RPO означают, что после восстановления системы могут быть утеряны данные не более чем за три часа до инцидента. А при RPO в несколько секунд сохранятся почти все данные, что особенно критично для банков, крупных девелоперов и других организаций, которым нельзя терять данные даже за минуту. Величина RPO влияет на частоту создания копий IT-инфраструктуры. Очевидно, что

стоимость решения Disaster Recovery будет тем дороже, чем меньше RTO/RPO. Подбирайте модель аварийного восстановления, стоимость которой не превышает размер убытков в случае простоя. Необходим баланс между затратами на катастрофоустойчивость и убытками из-за инцидента с учётом времени восстановления бизнес-процессов и объёма утерянных данных.

Понятие Disaster Recovery Plan:

Disaster Recovery Plan (DRP) — это план аварийного восстановления всех ИТ-систем после катастрофы (который в идеале никогда не должен понадобиться). Представляет собой документ с детальным описанием всех действий по устранению последствий аварии и восстановлению данных. В плане указаны роли и обязанности ответственных сотрудников, последовательность предпринимаемых ими действий. На каком этапе развития компании требуется DRP, сказать непросто. Можно сформулировать этот критерий следующим образом. Disaster Recovery Plan требуется компании, когда:

– Остановка сервера/приложения или потеря базы данных влечёт за собой значительные финансовые, репутационные или иные потери;

– В штате имеется полноценный IT-отдел со своим бюджетом;

– Есть реальная возможность выделить средства на полноценное или хотя бы частичное резервирование на случай возникновения аварии.

Если потеря БД за день ничего не меняет, а ИТ-отдел месяцами может ждать комплектующих к старому серверу, DRP вряд ли потребуется. Хотя этот документ способен выручить в трудной ситуации.

Основная цель Disaster Recovery Plan: создание пошаговой инструкции с указанием времени на выполнение отдельных процедур. С помощью плана компания:

– Сможет быстрее восстановить ИТ-инфраструктуру после сбоя;

– Сможет обеспечить работу критически важных процессов во время простоя основной площадки;

– Сможет сохранить важные данные компании.

План аварийного восстановления состоит из нескольких разделов. В первую очередь это цели разработки плана, факторы риска, список критически важных сервисов.

Целью DRP может являться:

– Подготовка сотрудников. Важно, чтобы в критической ситуации они не растерялись, а действовали чётко по инструкции.

– Сохранение работоспособности. Восстановление работы сервисов в короткий срок и сохранение данных.

– PR-контакты. Правильное взаимодействие со СМИ, клиентами партнёрами в момент аварии играет важную роль.

– Соблюдение стандартов. В ходе аварийного восстановления важно соблюдать корпоративные стандарты, чтобы избежать хаоса.

Факторы риска показывают, какие процессы требуют особого внимания в процессе аварийного восстановления. В документе прописываются действия по устранению этих рисков. Например, проверка корректности создания бэкапов, работы каналов резервной связи, тестирование резервной инфраструктуры, проверка наличия нужного оборудования. Список критически важных сервисов определяет очередность процессов восстановления. Чем критичнее процесс, тем быстрее нужно восстановить его работоспособность. Режим аварийного восстановления предполагает, что критические сервисы переносятся на резервную платформу. Поэтому даже при серьёзном инциденте их работоспособность должна сохраняться. Но если и с резервной площадкой что-то не так, работы по восстановлению начинаются с наиболее критичных систем.

1.4 Системы резервного копирований и аварийного восстановления данных на ОС Linux

Всегда полезно сохранять резервные копии данных с компьютеров, это можно делать вручную или настроить автоматическое выполнение. Многие

средства резервного копирования имеют разные функции, которые позволяют пользователям настраивать тип резервного копирования, время резервного копирования, протоколирование операций резервного копирования.

Имеется 10 инструментов для того, чтобы совершать резервное копирование на ОС Linux:

1. Rsync. Это средство резервного копирования командной строки, популярное среди пользователей Linux, особенно системных администраторов. Оно обладает богатыми возможностями, включая инкрементное резервное копирование, обновление всего дерева каталогов и файловой системы, как локальных, так и удаленных резервных копий, сохранение прав доступа к файлам, ссылок и многое другое. Также имеет графический пользовательский интерфейс Grsync, но главное преимущество с Rsync заключается в том, что резервные копии могут быть автоматизированы с использованием сценариев и заданий cron системными администраторами прямо в командной строке.
2. Fwbackups. Это бесплатное программное обеспечение с открытым исходным кодом, которое является кросс-платформенным и многофункциональным, и пользователи могут внести свой вклад в его разработку или просто участвовать в его тестировании. Оно имеет интуитивно понятный интерфейс, который позволяет пользователям легко делать резервные копии.
3. Bacula. Это программное обеспечение для резервного копирования, восстановления и проверки данных с открытым исходным кодом, предназначенное для того, чтобы быть готовым к работе с определенными сложностями, хотя эти сложности фактически и определяют его мощные функции, такие как резервные конфигурации, удаленное резервное копирование и многое другое.
4. Backupninja. Это мощный инструмент резервного копирования, который позволяет пользователям создавать файлы конфигурации резервной активности, которые можно поместить в каталог /etc/backup.d/. Он

помогает выполнять безопасные, удаленные и инкрементные резервные копии по сети.

5. Simple Backup Suite (sbackup). Это решение для рабочего стола Gnome, где пользователи могут получить доступ ко всей конфигурации через интерфейс Gnome. Пользователи могут использовать регулярное выражение для указания путей файлов и каталогов в процессе резервного копирования.
6. Kbackup. Это простой в использовании инструмент резервного копирования для операционной системы Unix и может использоваться в Linux. Он может создавать архивы и сжимать их, используя утилиты tar и gzip.
7. BackupPC. Это программное обеспечение для кросс-платформенной резервной копии, которое может работать в Unix/Linux, Windows и Mac OS X. Оно предназначено для использования на уровне предприятия с высокой производительностью. BackupPC может использоваться на серверах, настольных и портативных компьютерах.
8. Amanda. Amanda — это программное обеспечение с открытым исходным кодом, которое работает в Unix/GNU Linux и Windows. Оно поддерживает собственные утилиты резервного копирования и форматы, такие как GNU tar для резервного копирования в Unix/Linux. А для резервного копирования на Windows-машине он использует собственный клиент Windows. Пользователи могут настроить один резервный сервер для хранения резервных копий с нескольких компьютеров в сети.
9. Back Time. Это простой в использовании инструмент резервного копирования для операционной системы Linux.
10. Mondorescue. Это бесплатное программное обеспечение для резервного копирования и восстановления, которое является надежным и поддерживает множество функций. Оно может выполнять резервное копирование с персональных компьютеров, рабочих станций или

серверов на разделы жесткого диска, ленты, NFS, CD-[R|W], DVD-R[W], DVD +R[W] и т.д.

Иногда пользователю может понадобиться получить доступ к основным данным, сохраненным в хранилище компьютера под управлением Linux, но он может неоднократно потерпеть неудачу. Наиболее распространенными причинами такой проблемы являются вирусная атака, постоянное или случайное удаление файлов, сообщения об ошибках и так далее. Какой бы ни была причина, такая потеря может нанести пользователю непоправимый ущерб. Например непредвиденная ошибка, в случае которой он вряд ли сможет принять меры предосторожности. Но то, что пользователь определенно может сделать, это использовать инструменты Linux для восстановления потерянных данных. В качестве примера будут приведены 5 инструментов:

1. Ddrescue Data Recovery Tool. Ddrescue — это лицензионное программное обеспечение GNU, которое вы можете бесплатно использовать для восстановления потерянных данных. Этот дистрибутив восстановления данных Linux является «экспертом» в спасении данных при возникновении ошибок чтения. Для этого он просто копирует файл с компакт-диска или жесткого диска на другое внешнее или внутреннее устройство.
2. SafeCopy. SafeCopy — это инструмент восстановления файлов Linux, который предназначен для получения максимального объема данных с поврежденного диска. Этот инструмент написан с использованием языка программирования C.
3. TestDisk. TestDisk — это бесплатная утилита для восстановления данных с открытым исходным кодом. Она предназначена для восстановления потерянных разделов хранилища файлов. TestDisk также может превратить диск в загрузочный, если есть какая-либо ошибка вызвана вирусной атакой или неисправным программным обеспечением.

4. Redo Backup and Recovery. Redo Backup and Recovery считается самым простым в использовании системным программным обеспечением для восстановления компакт-дисков. Причина кроется в наличии улучшенного графического интерфейса и широко распространенных операциях. Утилита выпущена под GNU GPL3. Это один из инструментов восстановления дисков Linux, оснащенный удобными функциями многозадачности.
5. PhotoRec. PhotoRec относится к тем редким приложениям для восстановления дисков Linux, которые предназначены для восстановления отсутствующих файлов, содержащих документы, архивы и видео, с CD-ROM и жестких дисков. Эта мультиплатформенная программа с открытым исходным кодом совершенно бесплатна. Обычно она распределяется под «GNU General Public License».

2 ПРОЕКТИРОВАНИЕ СИСТЕМЫ РЕЗЕРВНОГО КОПИРОВАНИЯ И АВАРИЙНОГО ВОССТАНОВЛЕНИЯ ДАННЫХ НА LINUX

2.1 Описание локальной сети организации

Для выполнения практической части была смоделирована организация.

В соответствии с топологией локальной сети (рисунок 1), был составлен список имеющихся устройств:

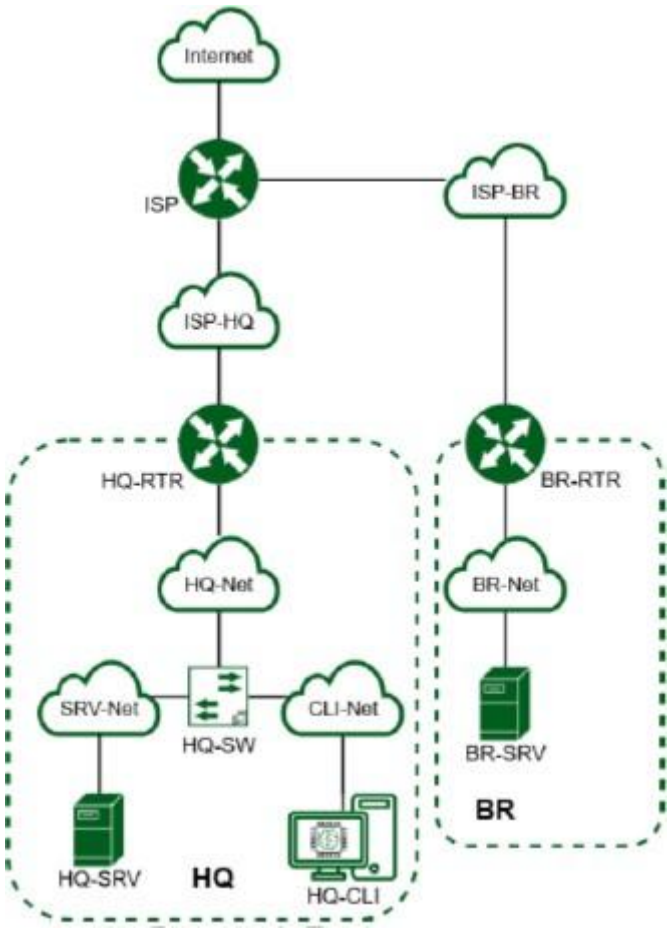

Рисунок 1. Топология локальной сети организации.

– ISP - маршрутизатор интернет-провайдера, позволяющий выходить всем устройствам в глобальную сеть Internet. Является шлюзом для двух других маршрутизаторов: BR-RTR и HQ-RTR.

– HQ-RTR - маршрутизатор, позволяющий устройствам в сети HQ выходить в сеть Internet, а также иметь связь с сетью BR через ip-tunnel с маршрутизатором BR-RTR. Является шлюзом для HQ-SRV, HQ-CLI, а также подсети для управления.

– HQ-SRV - сервер сети HQ, состоящий из 11 машин, обрабатывающих данные а также подключенные к сети HQ-SW.

– HQ-CLI - LAN сеть стационарных компьютеров, объединённых по топологии пассивная звезда. Состоит из 9 стационарных компьютеров, а также одного коммуникатора, принимающий все данные со стационарных компьютеров, после же пересылающий их в сеть HQ через подключение к сети HQ-SW.

– BR-RTR - маршрутизатор, позволяющий устройствам в сети BR выходить в сеть Internet, а также иметь связь с сетью HQ через ip -tunnel с маршрутизатором HQ-RTR Является шлюзом для BR-SRV.

– BR-SRV - сервер сети HQ, состоящий из 11 машин, хранящие данные из сети HQ, а также обрабатывающих их, в случае сбоя HQ-SRV.

Следуя списку устройств локальной сети, была составлена таблица (таблица 1) с раздачей IP каждому из устройств:

Таблица 1

Имя устройства	IP-адрес	Шлюз по умолчанию
ISP	172.16.222.130/16 172.16.4.1/28 172.16.5.1/28	172.16.222.1
HQ-RTR	172.16.4.2/28 192.168.99.1/29 192.168.100.1/28 192.168.200.1/28 10.5.5.1/30	172.16.4.1
HQ-SRV	192.168.100.10/28	192.168.100.1
HQ-CLI	192.168.200.10/28	192.168.200.1
BR-RTR	172.16.5.2/28 192.168.0.1/28 10.5.5.2/30	172.16.5.1
BR-SRV	192.168.0.2/28	192.168.0.1

Для дальнейшего поднятия DNS-сервера, была составлена таблица (таблица 2), где были расписаны устройства с их названиями и типами:

Таблица 2

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

2.2 Настройка IPS

В соответствии с таблицей всех IP устройств (таблица 1), была начата работа с ISP. Для начала требуется установить псевдографическую утилиту nmtui, дабы дальше работа проходила легче:

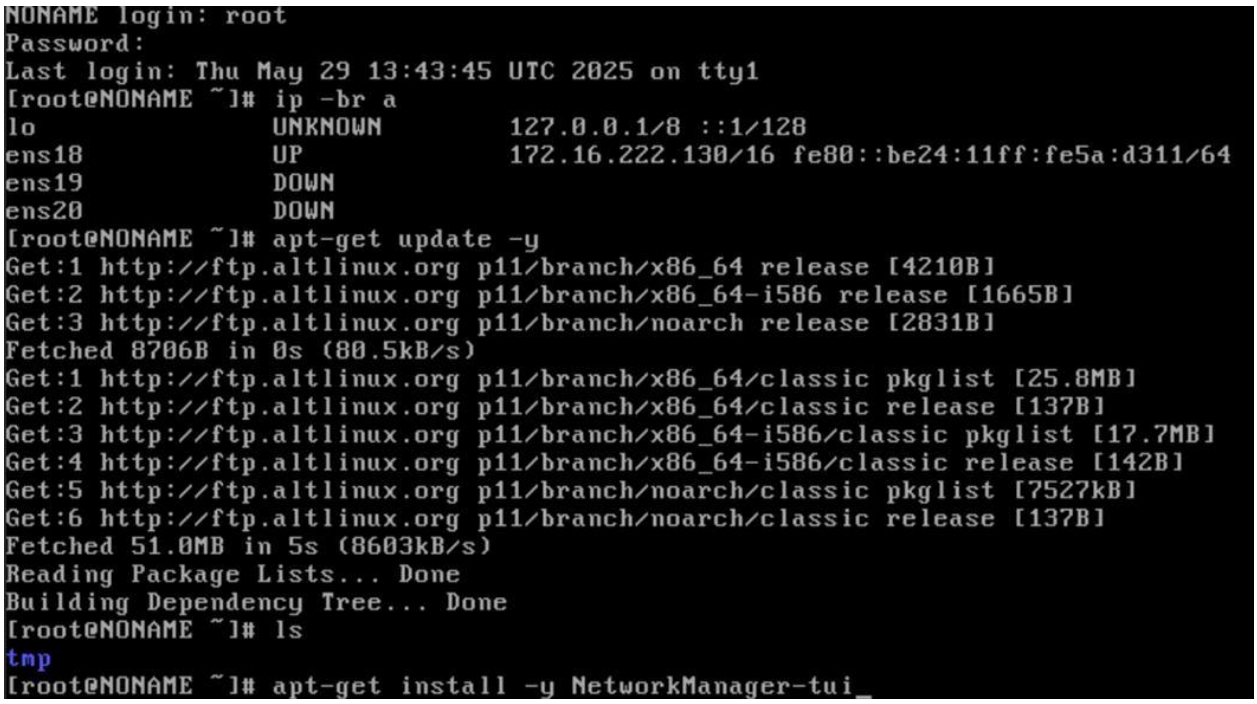

Рисунок 2. Установка NetworkManager-tui

Дальше, в файлах /options для ens18 и ens19 требуется изменить значение у “NM_CONTRILLED” на “yes”, это требуется чтобы утилита nmtui могла функционировать на устройстве:

Рисунок 3. Изменение значения NM_CONTROLLED

После настройки, производится включение систем nmtui, с последующим началом работы:

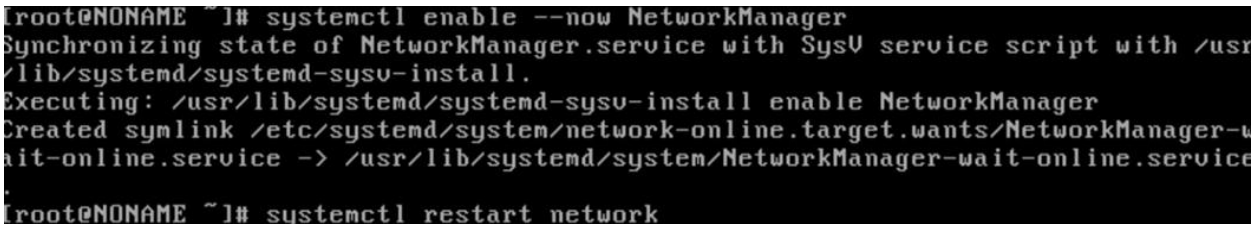

Рисунок 4. Включение систем nmtui

Раздача IP:

Сначала требуется скопировать /options ens19 в ens20, это нужно, чтобы настройки ens19 были теми же, что и у ens20, для последующей раздачи ip-адреса:

Рисунок 5. Копирование настроек ens19 в ens20

Далее даётся ip с временным доменом в ens19 и ens20 (впоследствии маски были изменены с 20 на 28). Задать IP можно несколькими способами, в данном примере показан способ через команду “echo”, где в файле ipv4address даётся ip-адрес устройства, а в файле resolv.conf выдаётся запись DNS адреса(для IPS даётся nameserver 8.8.8.8 для выхода в сеть. Сам DNS адрес в остальных устройствах будет настраиваться дальше, однако временно будет выдан адрес 8.8.8.8):

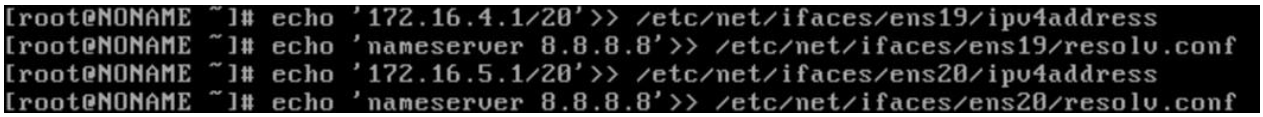

Рисунок 6. Установка значений IP

После раздачи IP, включается раздача пакетов по пути “/etc/net/sysctl.con”:

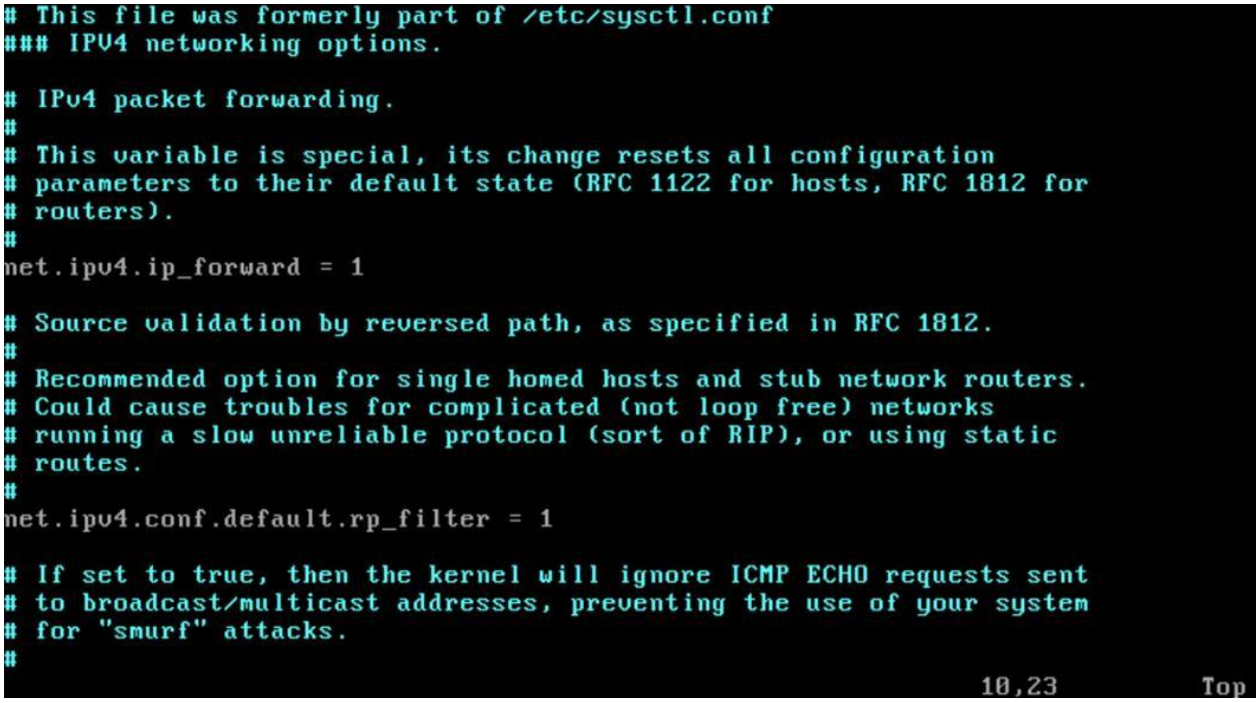

Рисунок 7. Включение раздачи пакетов

В строке net.ipv4.ip_forward значение меняется с “0” на “1”. После требуется установить iptables, для включения NAT:

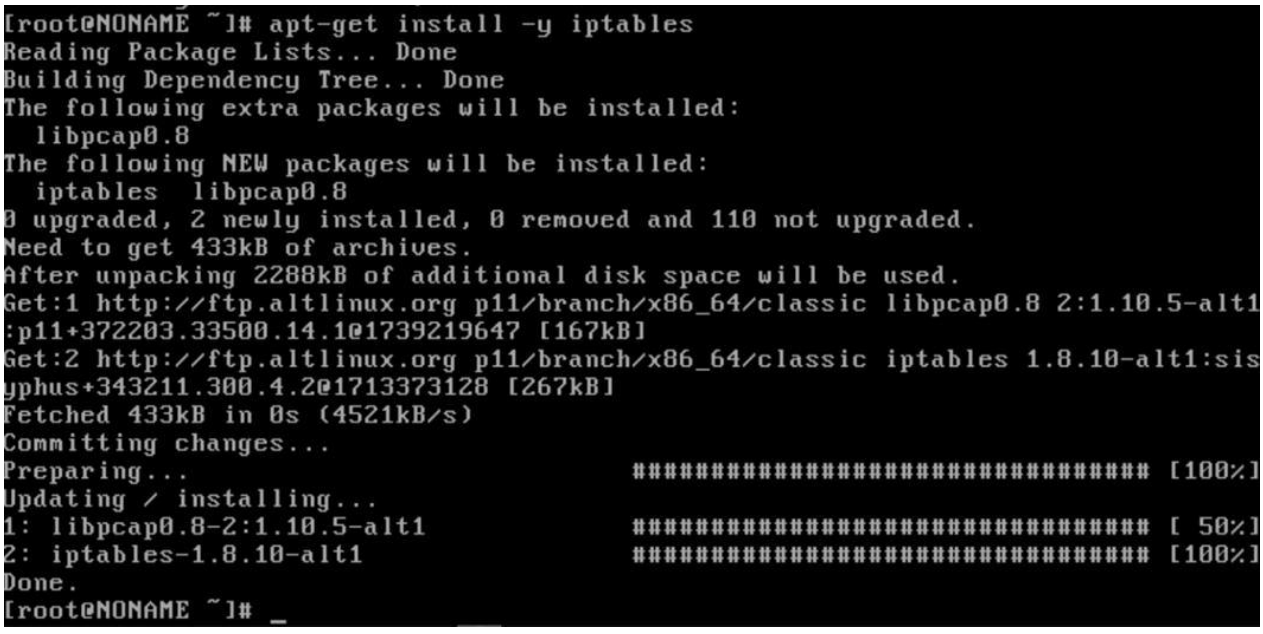

Рисунок 8. Установка iptables

```
[root@NONAME ~]# iptables -t nat -j MASQUERADE -A POSTROUTING -o ens18
[root@NONAME ~]# iptaples-save
-bash: iptaples-save: command not found
[root@NONAME ~]# iptables-save
# Generated by iptables-save v1.8.10 on Fri May 30 08:08:47 2025
*nat
:PREROUTING ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
-A POSTROUTING -o ens18 -j MASQUERADE
COMMIT
# Completed on Fri May 30 08:08:47 2025
[root@NONAME ~]#
```

Рисунок 9.

Далее происходит задача наименования устройства:

```
[root@NONAME ~]# hostnamectl set-hostname ISP
[root@NONAME ~]# exab bash
-bash: exab: command not found
[root@NONAME ~]# exec bash
[root@ISP ~]#
```

Рисунок 10. Установка имени устройства

Дальше будет проходить работа в BR-RTR:

Для начала требуется настроить /options в ens18, дабы затем скопировать его в ens19:

```
BOOTPROTO=static
TYPE=eth
CONFIG_WIRELESS=no
SYSTEMD_BOOTPROTO=static
CONFIG_IPV4=yes
DISABLED=no
NM_CONTROLLED=no
SYSTEMD_CONTROLLED=no
```

Рисунок 11. Настройка NM_CONTROLLED

```
"/etc/net/ifaces/ens18/options" 8L, 137B written
[root@noname ~]# mkdir /etc/net/ifaces/ens19
[root@noname ~]# cp /etc/net/ifaces/ens18/options /etc/net/ifaces/ens19/options
```

Рисунок 12. Копирование данных в ens19

После задаётся IP, временный DNS сервер и шлюз, где ens18 - ISP, а ens19 - BR-SRV:

```
[root@noname ~]# echo '172.16.5.2/20' >> /etc/net/ifaces/ens18/ipv4address
[root@noname ~]# echo 'default via 172.16.5.1' >> /etc/net/ifaces/ens18/ipv4route
[root@noname ~]# echo 'nameserver 8.8.8.8' >> /etc/net/ifaces/ens18/resolv.conf
```

Рисунок 13. Установка IP и DNS для ens18

```
[root@noname ~]# echo '196.168.0.1/28' >> /etc/net/ifaces/ens19/ipv4address
[root@noname ~]# echo 'nameserver 8.8.8.8' >> /etc/net/ifaces/ens19/resolv.conf
```

Рисунок 14. Установка IP и DNS для ens19

```
[root@noname ~]# systemctl restart network
[root@noname ~]# ip -br a
lo               UNKNOWN        127.0.0.1/8 ::1/128
ens18            UP             172.16.5.2/28 fe80::be24:11ff:fe37:9163/64
ens19            UP             196.168.0.1/28 fe80::be24:11ff:feda:c13b/64
```

Рисунок 15. Проверка

Включение пересылки пакетов дабы маршрутизатор мог пересылать пакеты в BR-SRV от ISP:

```
# This file was formerly part of /etc/sysctl.conf
### IPV4 networking options.

# IPv4 packet forwarding.
#
# This variable is special, its change resets all configuration
# parameters to their default state (RFC 1122 for hosts, RFC 1812 for
# routers).
#
net.ipv4.ip_forward = 1

# Source validation by reversed path, as specified in RFC 1812.
#
# Recommended option for single homed hosts and stub network routers.
# Could cause troubles for complicated (not loop free) networks
# running a slow unreliable protocol (sort of RIP), or using static
# routes.
#
net.ipv4.conf.default.rp_filter = 1

# If set to true, then the kernel will ignore ICMP ECHO requests sent
# to broadcast/multicast addresses, preventing the use of your system
# for "smurf" attacks.
#
net.ipv4.icmp_echo_ignore_broadcasts = 1

# TCP SYN cookies: http://cr.yp.to/syncookies.html
#
# If set to true and the kernel was compiled with CONFIG_SYN_COOKIES,
# it will send out SYN cookies when the SYN backlog queue of a socket
# overflows, defeating SYN flood attacks.  Note that SYN cookies make
# it possible (although hopefully impractical) to bypass certain
# packet filter setups which disallow incoming packets based on the
# SYN flag.  This is because with SYN cookies the attacker no longer
# strictly needs to send the initial SYN, but rather may guess a valid
# SYN cookie.
#
net.ipv4.tcp_syncookies = 1

# TCP timestamps, as specified in RFC 1323.
#
# The primary purpose of TCP timestamps is to allow for more accurate
# measurement of round-trip time, which in turn helps improve TCP
# performance over large bandwidth*delay product paths.  Other TCP
# extensions also aimed at improving transfer rate include scaled windows
# (also specified in RFC 1323) and selective acknowledgments (RFC 2018).
#
# Unfortunately, the sending of TCP timestamps as currently implemented
# in the Linux kernel leaks information which some may view as sensitive:
```

Рисунок 16. Включение пересылки пакетов

В конце задаётся имя у устройства:

```
[root@noname ~]# hostnamectl set-hostname BR-RTR.au-team.irpo
[root@noname ~]# exec bash
[root@BR-RTR ~]#
```

Рисунок 17. Установка имени

Для HQ-RTR проводятся аналогичные действия, как и для BR-RTR. Настройка BR-SRV будет проходить также, как и HQ-SRV, посему настройка будет показана BR-SRV: Для начала требуется настроить /options в ens18:

Рисунок 18. Установка NM_CONTROLLED

После же задаётся IP, временный DNS сервер и шлюз:

```
[root@noname ~]# echo '192.168.0.2/28' >> /etc/net/ifaces/ens18/ipv4address
[root@noname ~]# echo 'default via 192.168.0.1' >> /etc/net/ifaces/ens18/ipv4route
[root@noname ~]# echo 'nameserver 8.8.8.8' >> /etc/net/ifaces/ens18/resolv.conf
[root@noname ~]# systemctl restart network
[root@noname ~]# ip -br a
lo               UNKNOWN        127.0.0.1/8 ::1/128
ens18            UP             192.168.0.2/28 fe80::be24:11ff:fe27:b600/64
[root@noname ~]#
```

Рисунок 19. Установка IP, DNS и шлюза

В конце задаётся имя у устройства:

```
[root@noname ~]# hostnamectl set-hostname BR-SRV.au-team.irpo
[root@noname ~]# exec bash
[root@BR-SRV ~]#
```

Рисунок 20. Установка имени

2.3 Создание локальных учетных записей

Для HQ-SRV и BR-SRV требуется создать пользователя sshuser со следующими характеристиками:

– Пароль пользователя sshuser с паролем P@ssw0rd

– Идентификатор пользователя 1010

– Пользователь sshuser должен иметь возможность запускать sudo без дополнительной аутентификации.

Так как пользователи на обеих устройствах будет один и тот же, настройка будет показана на HQ-SRV:

Создание юзера с последующей задачей пароля и идентификатора. Создание юзера производится командой “adduser”, задача пароля юзеру производится командой “passwd”. Для того, чтобы дать пользователю идентификатор, требуется использовать (индентификатор) (имя пользователя)”: команду “usermod -u

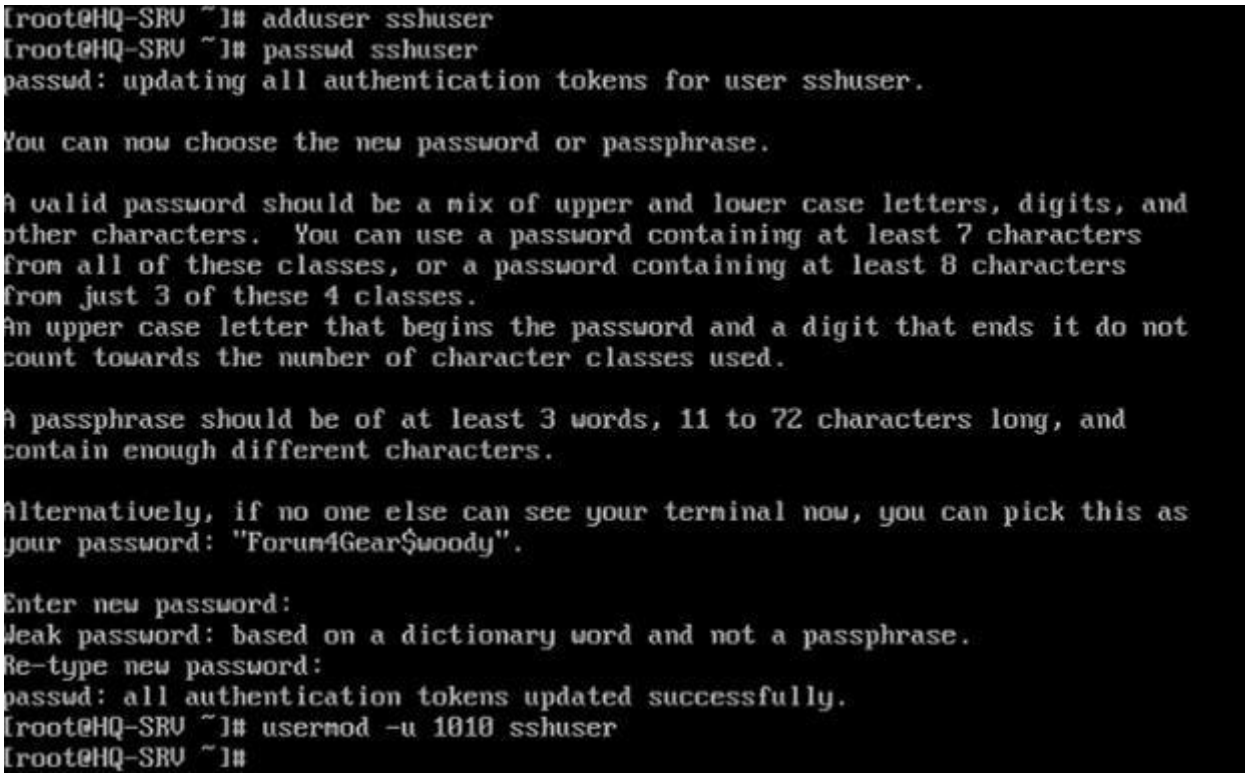

Рисунок 21. Создание нового пользователя

После создания и настройки, дается пользователю запускать sudo без дополнительной аутентификации. Для этого потребуется поработать в файлах /etc/sudoers, а также /etc/group. В файле “sudoers” выдаётся возможность всем юзерам из группы “wheel” запускать sudo без аутентификации. А в файле “group”, добавляется пользователь sshuser в группу “wheel”:

Рисунок 22. Открытие конфигурационного файла

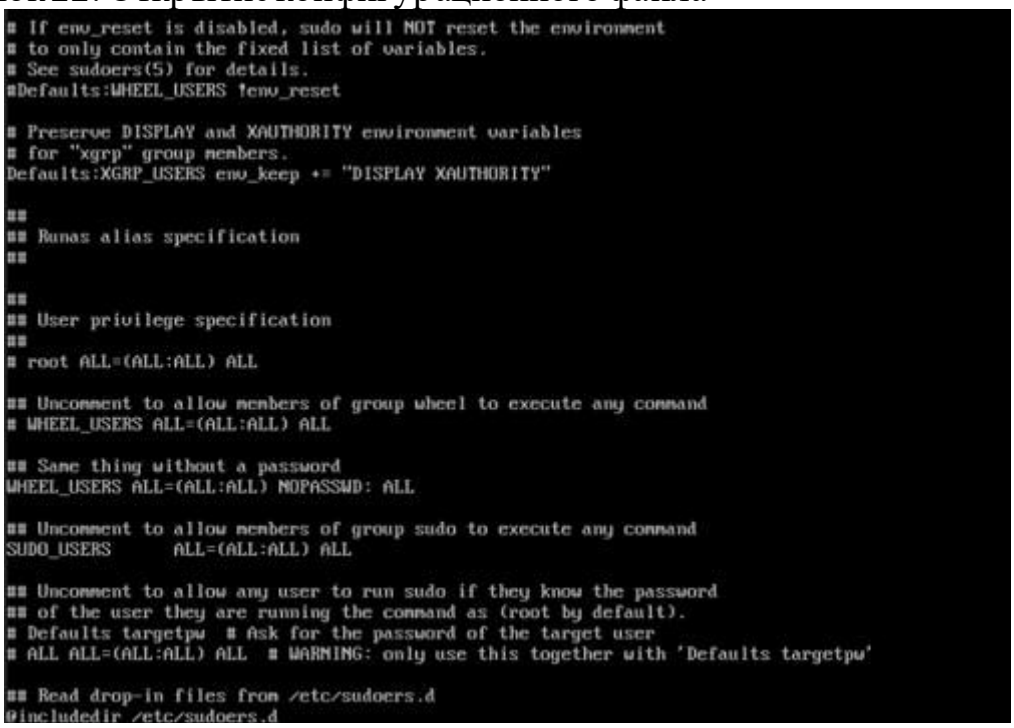

Рисунок 23. Установка конфигурационного файла

Рисунок 24. Открытие конфигурационного файла

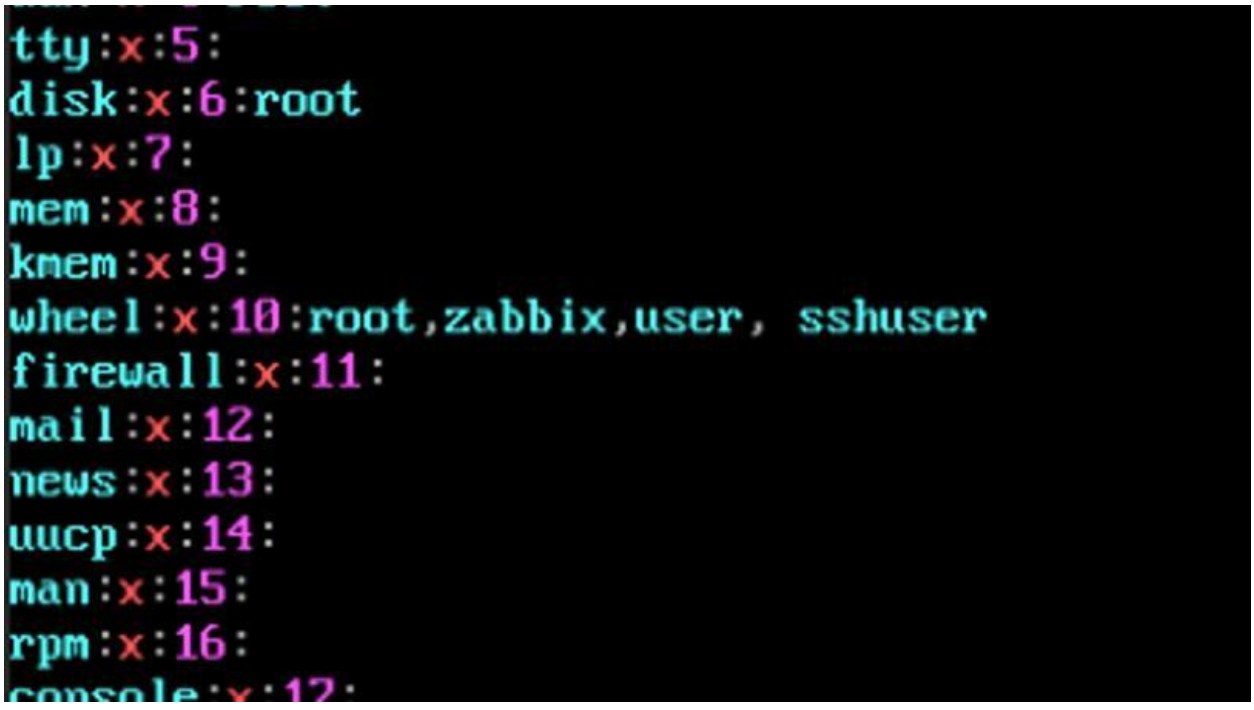

Рисунок 25. Настройка конфигурационного файла

Для HQ-RTR и BR-RTR требуется создать пользователя net_admin со следующими характеристиками:

– Пароль пользователя net_admin с паролем P@$$word

– При настройке ОС на базе Linux, запускать sudo без дополнительной аутентификации

– Так как пользователи на обеих устройствах будет один и тот же, настройка будет показана на HQ-RTR:

Для начала создается пользователь и дается ему пароль. Эти действия производится аналогично тем, что производились при создании юзера на HQ-SRV и BR-SRV, за исключением того, что для пользователя на HQ-RTR и BR-RTR не требуется задача идентификатора:

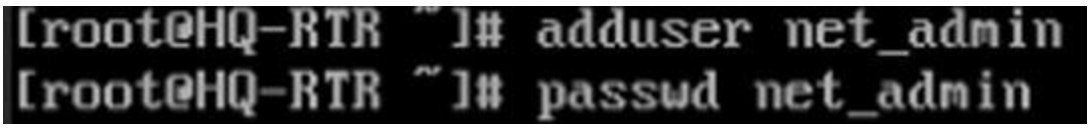

Рисунок 26. Добавление нового пользователя

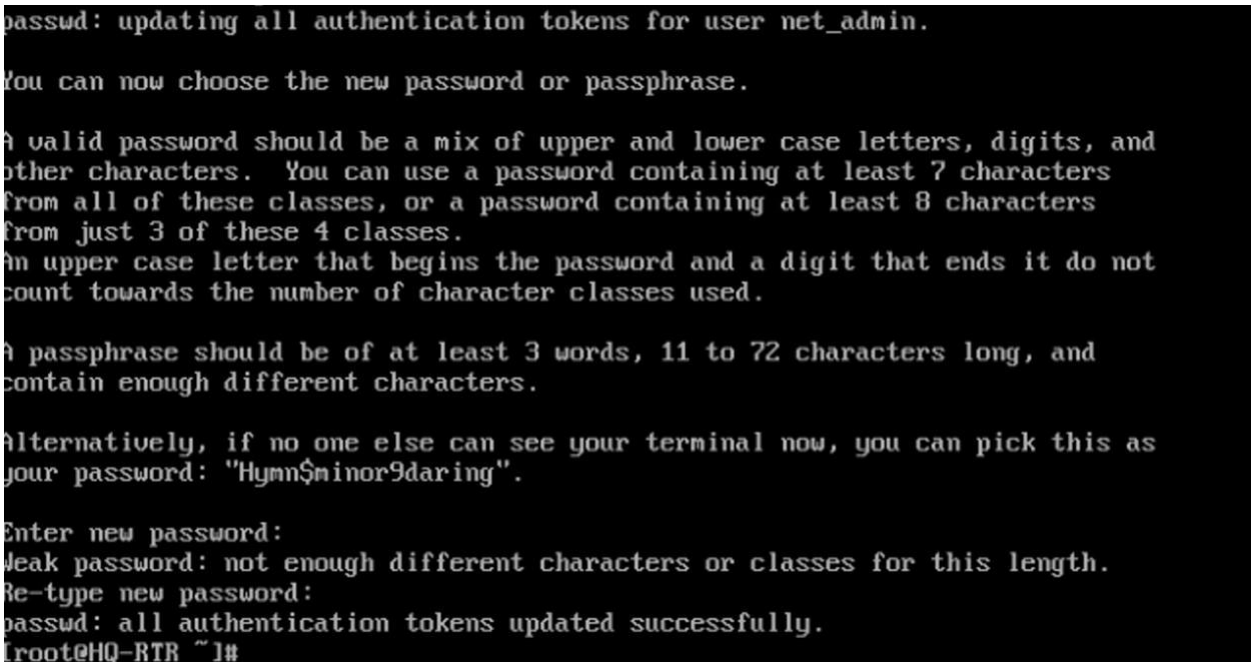

Рисунок 27. Подтверждение создания нового пользователя

После создания и настройки, дается пользователю запускать sudo без дополнительной аутентификации. Для этого потребуется поработать в файлах /etc/sudoers, а также /etc/group. Все действия аналогичны тем, что и при настройке пользователя для HQ-SRV и BR-SRV:

Рисунок 28. Открытие конфигурационного файла

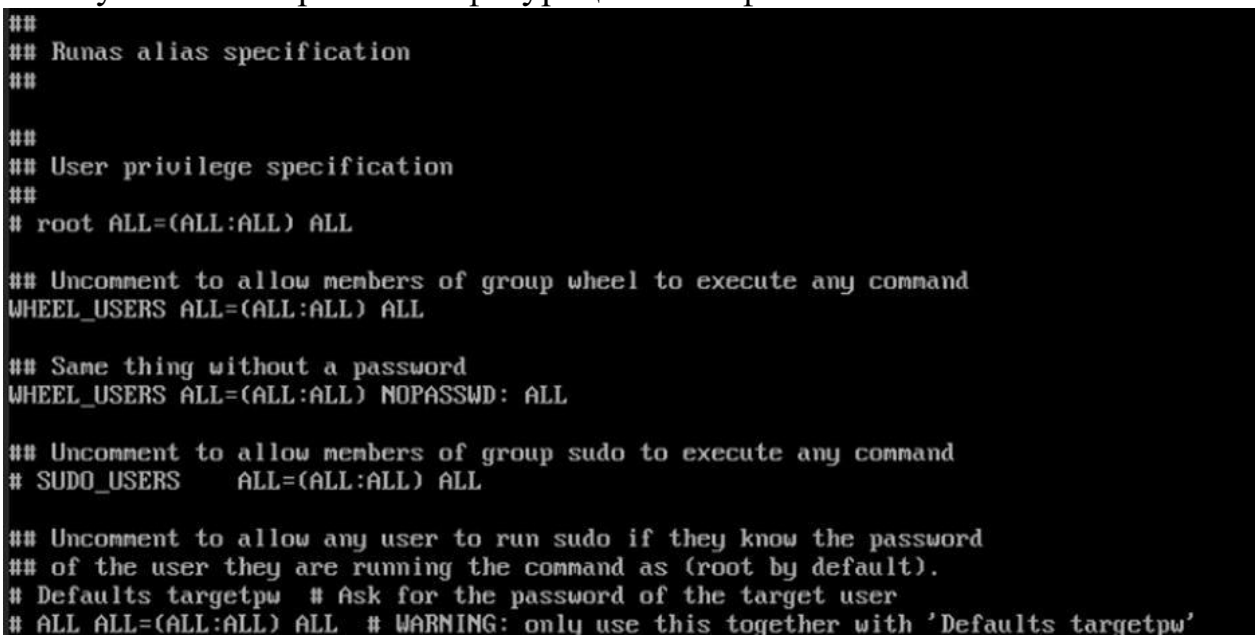

Рисунок 29. Настройка конфигурационного файла

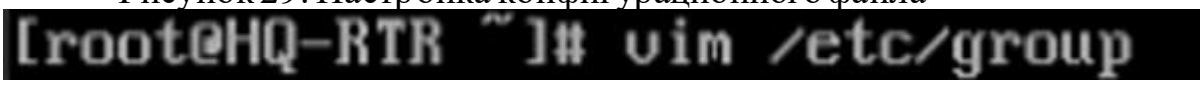

Рисунок 30. Открытие конфигурационного файла групп

Рисунок 31. Настройка конфигурационного файла групп

2.4 Настройка на интерфейсе HQ-RTR в сторону офиса HQ виртуального коммутатора

Перед началом настройки интерфейса, требуется ознакомиться с требованиями:

– Сервер HQ-SRV должен находиться в ID VLAN 100;

– Клиент HQ-CLI в ID VLAN 200;
– Подсеть управления с ID VLAN 999;

Для работы требуется установить и включить службу openswitch:

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

Рисунок 32. Установка openvswitch

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

Рисунок 33. Включение openvswitch

Далее создается бридж, который затем закрепляется на интерфейс в сторону CLI и SRV с последующим созданием вланов и их привязки к бриджу. Для начала создается мост, который закрепляется на ens19. После же, задается настройка для всех трех сетей, дабы сети могли нормально функционировать. После данной настройки, задаются ip-адреса для каждой созданной сети:

```
[root@HQ-RTR ~]# ovs-vsctl add-br HQ-SW
[root@HQ-RTR ~]# ovs-vsctl add-port HQ-SW ens19
[root@HQ-RTR ~]#
```

Рисунок 34. Создание моста

```
[root@HQ-RTR ~]# ovs-vsctl add-port HQ-SW vlan100 tag=100 -- set interface vlan100 type=internal
[root@HQ-RTR ~]# ovs-vsctl add-port HQ-SW vlan200 tag=200 -- set interface vlan200 type=internal
[root@HQ-RTR ~]# ovs-vsctl add-port HQ-SW vlan999 tag=999 -- set interface vlan999 type=internal
[root@HQ-RTR ~]#
```

Рисунок 35. Создание моста

```
[root@HQ-RTR ~]# mkdir /etc/net/ifaces/vlan100
[root@HQ-RTR ~]# mkdir /etc/net/ifaces/vlan200
[root@HQ-RTR ~]# mkdir /etc/net/ifaces/vlan999
[root@HQ-RTR ~]# cp /etc/net/ifaces/ens18/options /etc/net/ifaces/vlan100
[root@HQ-RTR ~]# cp /etc/net/ifaces/ens18/options /etc/net/ifaces/vlan200
[root@HQ-RTR ~]# cp /etc/net/ifaces/ens18/options /etc/net/ifaces/vlan999
[root@HQ-RTR ~]# echo '192.168.16.1/26' >> /etc/net/ifaces/vlan100
bash: /etc/net/ifaces/vlan100: Is a directory
[root@HQ-RTR ~]# echo '192.168.16.1/26' >> /etc/net/ifaces/vlan100/ipv4address
[root@HQ-RTR ~]# echo '192.168.16.65/86' >> /etc/net/ifaces/vlan200/ipv4address
[root@HQ-RTR ~]# echo '192.168.16.65/28' >> /etc/net/ifaces/vlan200/ipv4address
[root@HQ-RTR ~]# echo '192.168.16.81/29' >> /etc/net/ifaces/vlan999/ipv4address
```

Рисунок 36. Создание моста

После перезапуска сети будут видны вланы:

```
ens19            UP              fe80::9ad:daic:4e6c:fd5e/64
ovs-system       DOWN
vlan100          UNKNOWN         192.168.100.1/28 fe80::5402:97ff:fef2:5995/64
HQ-SW            DOWN
vlan200          UNKNOWN         192.168.200.1/28 fe80::5c0e:eeff:fed0:dcdd/64
vlan999          UNKNOWN         192.168.99.1/29 fe80::7882:7eff:fe01:6255/64
```

Рисунок 37. Проверка вланов

2.5 Настройка безопасного удаленного доступа на серверах HQ-SRV и BR-SRV

Так как результатом работы будет один и тот же исход, показана работа будет на HQ-SRV: Настройка безопасного удаленного доступа будет проводиться согласно следующим параметрам:

– Для подключения используется порт 2024
– Разрешить подключение только пользователю sshuser
– Ограничение количества попыток входа до двух
– Настройка баннера «Authorized access only

Сперва проводится настройка SSH в файле конфигурации. В данном файле требуется видоизменить 4 строки: Port, AddressFamily, MaxAuthTries и Banner. В каждой строке задаются те значения, которые даны в условии, за исключением Banner. В данной строке задается путь к баннеру:

Рисунок 38. Открытие конфигурационного файла

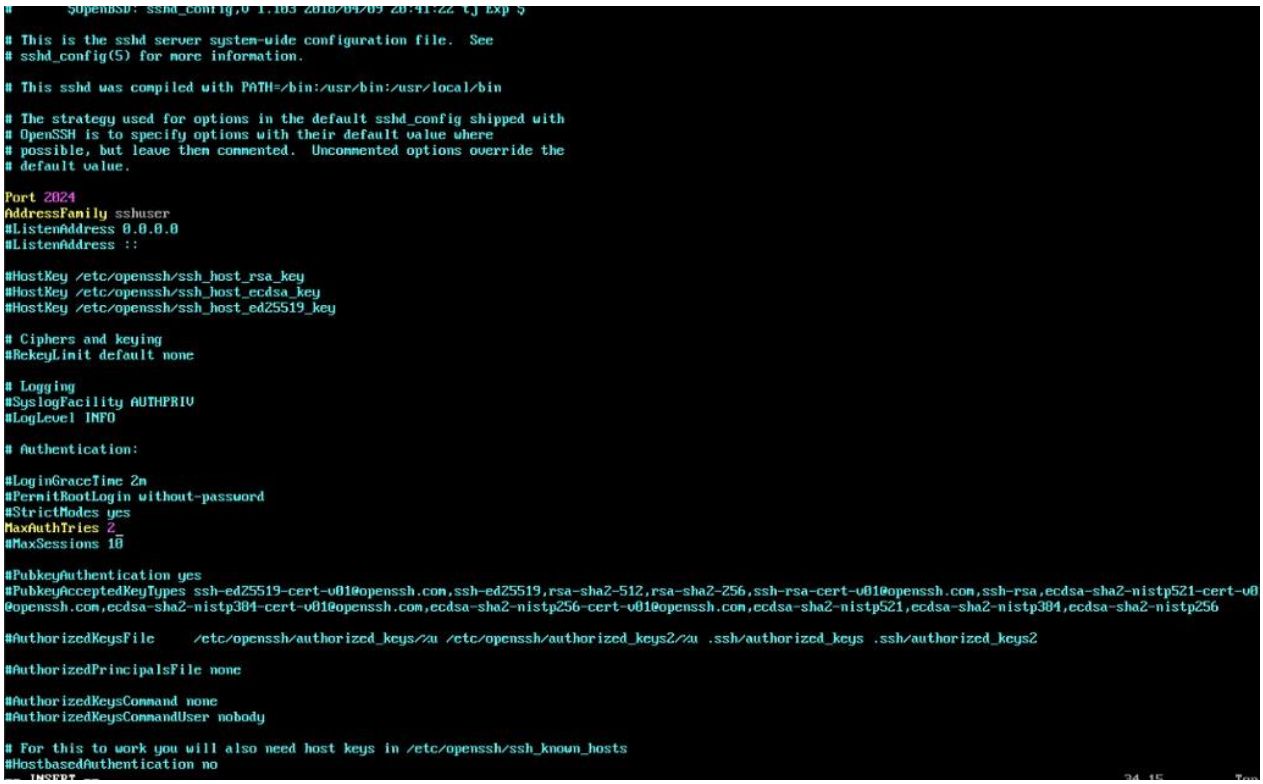

Рисунок 39. Настройка конфигурационного файла

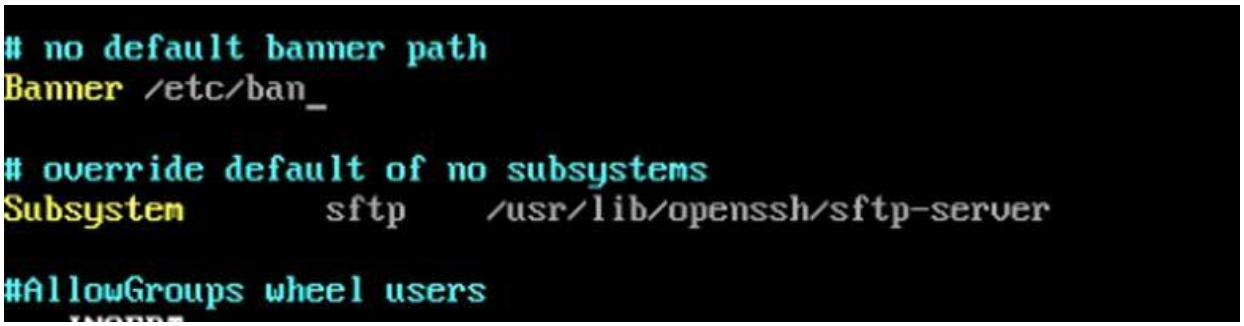

Рисунок 40. Указание на путь до баннера

После этого создается сам баннер. Это делается по тому пути, который указан в строке конфигурационного файла Banner:

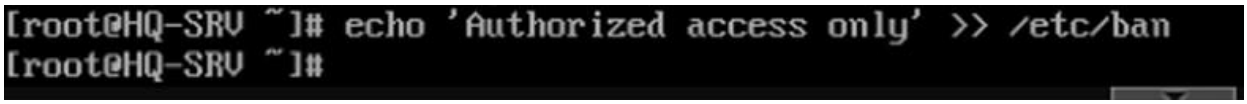

Рисунок 41. Создание баннера

2.6 Между офисами HQ и BR необходимо сконфигурировать ip туннель

Для того, чтобы сконфигурировать ip туннель, будет использоваться псевдографическая утилита nmtui. Так как работа будет вестись на маршрутизаторах HQ-RTR и BR-RTR, показана работа будет на HQ-RTR:

Для начала требуется установить и включить утилиту на устройстве. Это делается аналогично с тем же, как настраивалась и включалась в задании с настройкой ISP. Далее открывается утилита nmtui и происходит создание ip туннеля. Edit a connection -> Add -> IP Tunnel -> Дальше происходит настройка создания самого ip-туннеля, для которого будет выделена сеть 10.5.5.0/30:

Рисунок 42. Открытие утилиты nmtui

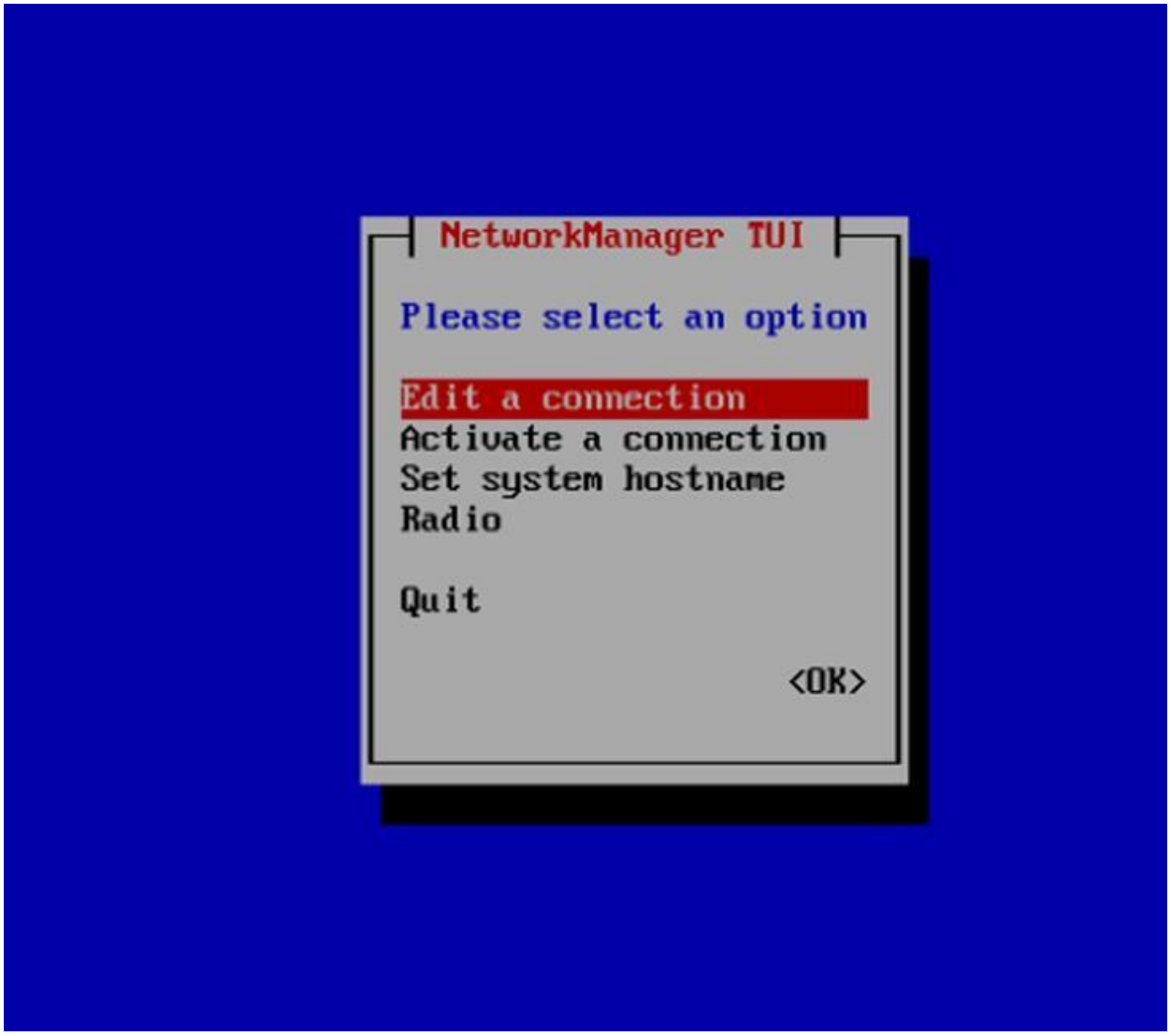

Рисунок 43. Выбор вкладки Edit a connection

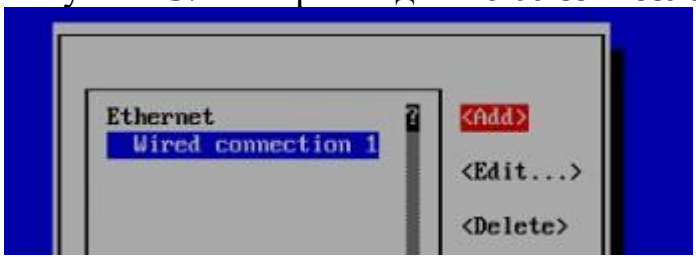

Рисунок 44. Добавление нового туннеля

Рисунок 45. Добавление IP tunnel

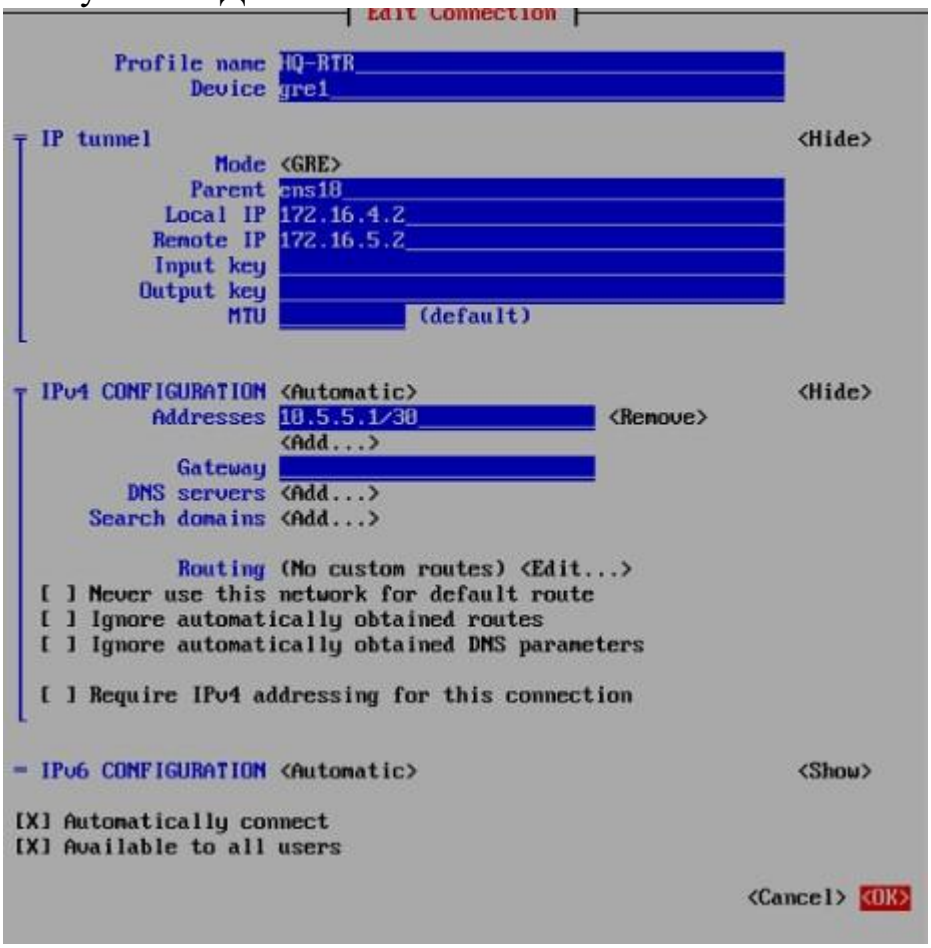

Рисунок 46. Настройка IP tunnel

После же активируем сам IP туннель и проверяем его в видимость устройством:

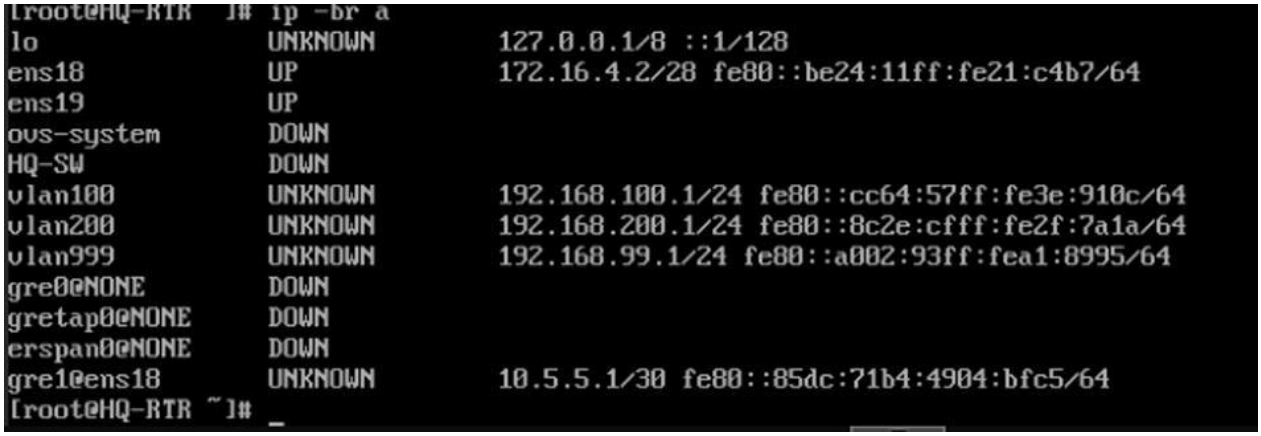

Рисунок 47. Проверка видимости туннеля

2.7 Обеспечить динамическую маршрутизацию: ресурсы одного офиса должны быть доступны из другого офиса

Обеспечение динамической маршрутизации будет происходить согласно требуемым параметрам:

– Разрешить выбранный протокол только на интерфейсах в ip туннеле

– Маршрутизаторы должны делиться маршрутами только друг с другом

– Обеспечить защиту выбранного протокола посредством парольной защиты

Работа будет проходить на двух маршрутизаторах: HQ-RTR и BR-RTR. Поэтому работа будет показана на HQ-RTR:

Установка пакета frr с последующей её настройкой:

Рисунок 48. Установка утилиты frr

Рисунок 49. Открытие конфигурационного файла

Рисунок 50. Настройка конфигурационного файла

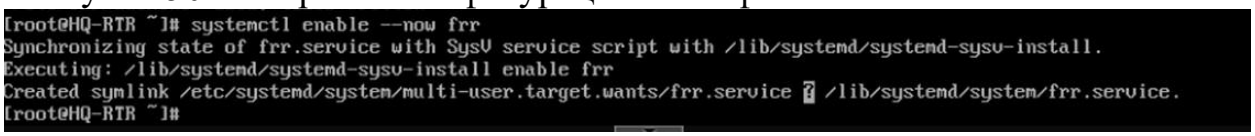

Рисунок 51. Включение утилиты frr

После же происходит сама работа:

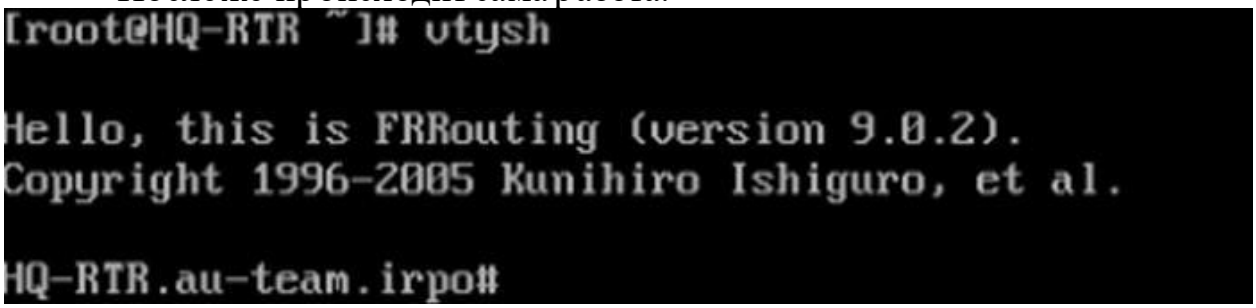

Рисунок 52.

```
HQ-RTR.au-team.irpo# conf t
HQ-RTR.au-team.irpo(config)# ip forwarding
HQ-RTR.au-team.irpo(config)# router ospf
HQ-RTR.au-team.irpo(config-router)# network 10.5.5.0/30 area 0
HQ-RTR.au-team.irpo(config-router)# network 196.168.100.0/28 area 0
HQ-RTR.au-team.irpo(config-router)# network 196.168.200.0/28 area 0
HQ-RTR.au-team.irpo(config-router)# network 196.168.99.0/29 area 0
HQ-RTR.au-team.irpo(config-router)# ex
HQ-RTR.au-team.irpo(config)# int gre1
HQ-RTR.au-team.irpo(config-if)# no ip ospf passive
HQ-RTR.au-team.irpo(config-if)# ex
HQ-RTR.au-team.irpo(config)# ex
HQ-RTR.au-team.irpo# wr
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Integrated configuration saved to /etc/frr/frr.conf
[OK]
HQ-RTR.au-team.irpo# ex
[root@HQ-RTR ~]#
```

Рисунок 53. Настройка с помощью frr

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

Рисунок 54. Перезапуск frr

Работа на BR-RTR:

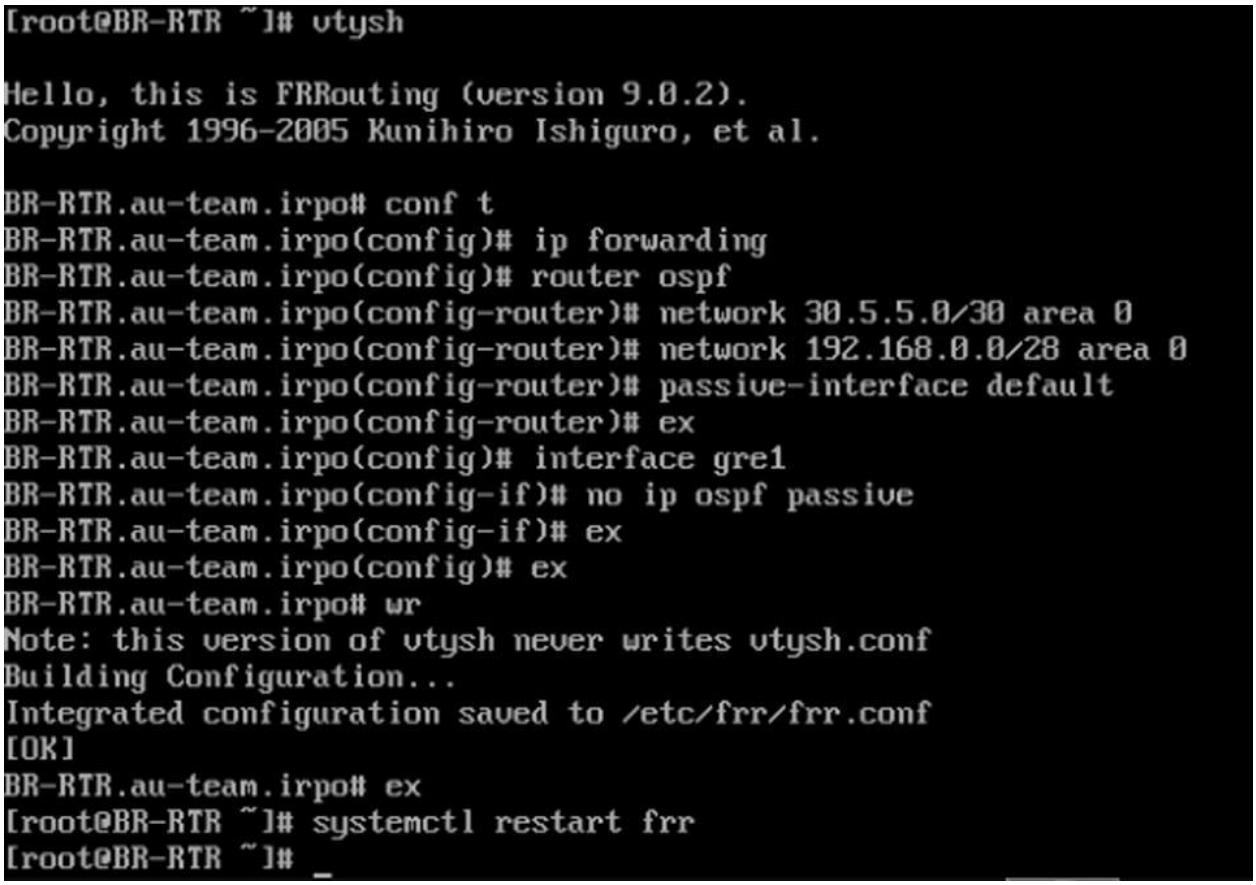

Рисунок 55. Настройка с помощью frr

2.8 Настройка динамической трансляции адресов

Работа будет вестись на маршрутизаторах HQ-RTR и BR-RTR, из-за чего работа будет показана на HQ-RTR:

Настройка iptables на устройстве HQ-RTR:

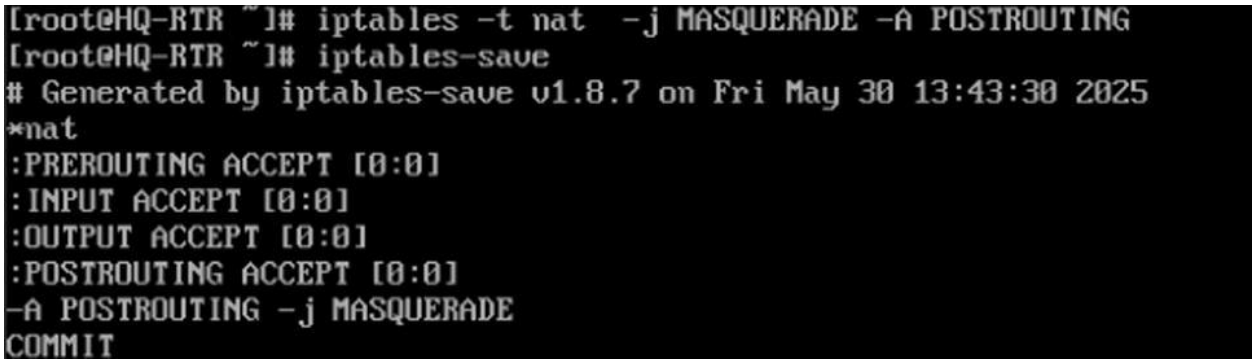

Рисунок 56. Настройка iptables

2.9 Настройка протокола динамической конфигурации хостов.

Работа будет вестись на маршрутизаторе HQ-RTR, с учетом всех требуемых данных:

– Настройка нужной подсети

– Для офиса HQ в качестве сервера DHCP выступает маршрутизатор HQ-RTR.

– Клиентом является машина HQ-CLI.

– Исключить из выдачи адрес маршрутизатора

– Адрес шлюза по умолчанию – адрес маршрутизатора HQ-RTR.

– Адрес DNS-сервера для машины HQ-CLI – адрес сервера HQ-SRV.

– DNS-суффикс для офисов HQ – au-team.irpo

Перед началом работы требуется установить утилиту dhcp -server, с последующей её настройкой:

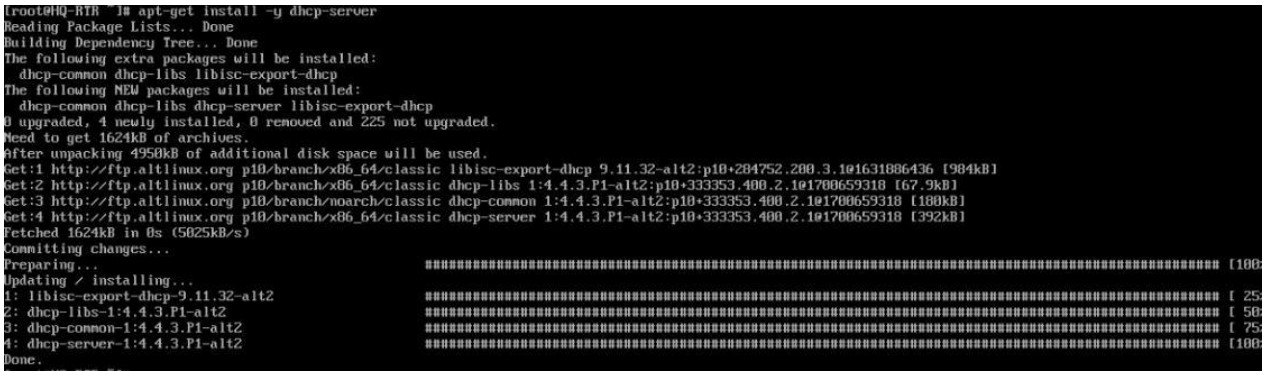

Рисунок 57. Установка утилиты dhcp-server

Рисунок 58. Открытие конфигурационного файла

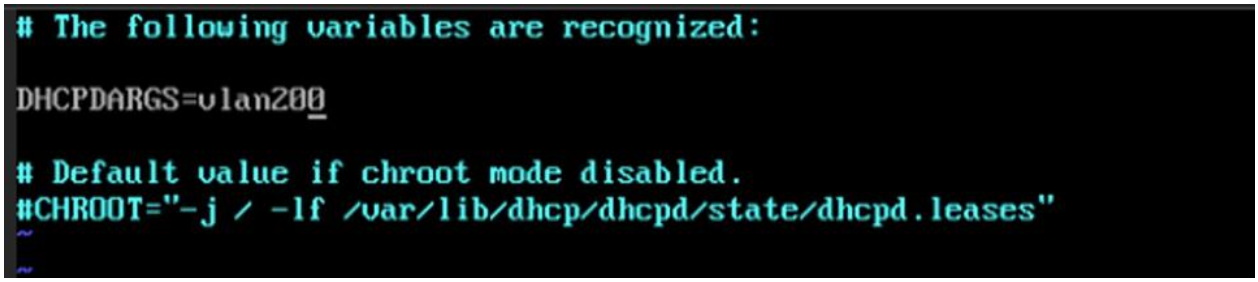

Рисунок 59. Настройка конфигурационного файла

После же происходит сама работа с конфигурацией, однако работа будет проходить в шаблоне:

Рисунок 60. Копирование шаблона

```
# option definitions common to all supported networks...
option domain-name "au-team.irpo";
option domain-name-servers 192.168.100.2, 192.168.0.2;
```

Рисунок 61. Настройка конфигурационного файла

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

zone au-team.irpo {
        primary 192.168.100.2;
}
zone 100.168.192.in-addr.arpa {
        primary 192.168.100.2;
}
zone 200.168.192.in-addr-arpa {
        primary 192.168.100.2;
}
```

Рисунок 62. Настройка dhcp-server

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

host hq-cli {
        hardware ethernet bc:24:11:58:f7:ab;
        fixed-address 192.168.200.2;
}
```

Рисунок 63.

После работы производится перезапуск службы:

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

Рисунок 64. Перезапуск служб

2.10 Настройка DNS для офисов HQ и BR

Задание будет выполняться на устройстве HQ-SRV в соответствии с требуемыми данными:

– Основной DNS-сервер реализован на HQ-SRV.

– Сервер должен обеспечивать разрешение имен в сетевые адреса устройств и обратно в соответствии имеющимися данными (таблица 2)

– В качестве DNS сервера пересылки используйте любой общедоступный DNS сервер

Перед началом работы требуется установить утилиту bind:

```
[root@HQ-SRV ~]# apt-get update
Get:1 http://ftp.altlinux.org p10/branch/x86_64 release [4215B]
Get:2 http://ftp.altlinux.org p10/branch/x86_64-i586 release [1665B]
Get:3 http://ftp.altlinux.org p10/branch/noarch release [2836B]
Fetched 8716B in 0s (24.3kB/s)
Get:1 http://ftp.altlinux.org p10/branch/x86_64/classic pkglist [24.5MB]
Get:2 http://ftp.altlinux.org p10/branch/x86_64/classic release [137B]
Get:3 http://ftp.altlinux.org p10/branch/x86_64-i586/classic pkglist [18.0MB]
Get:4 http://ftp.altlinux.org p10/branch/x86_64-i586/classic release [142B]
Get:5 http://ftp.altlinux.org p10/branch/noarch/classic pkglist [7304kB]
Get:6 http://ftp.altlinux.org p10/branch/noarch/classic release [137B]
Fetched 49.7MB in 34s (1448kB/s)
Reading Package Lists... Done
Building Dependency Tree... Done
[root@HQ-SRV ~]# apt-get install -y bind
```

Рисунок 65. Установка утилиты bind

После установки, происходит настройка конфига по пути “/var/lib/bind/etc/options.conf”:

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

Рисунок 66. Настройка конфига

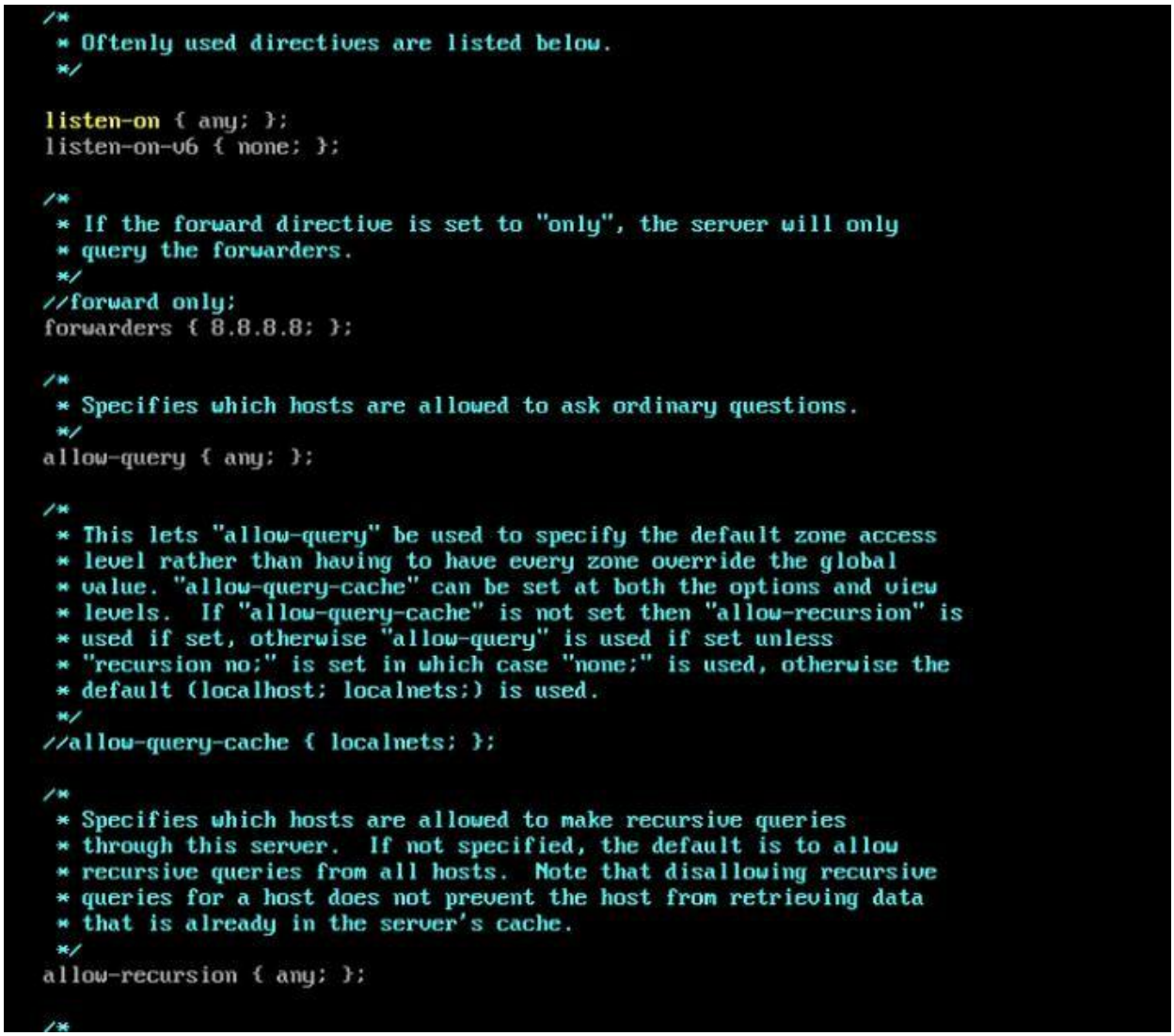
/*
* Oftenly used directives are listed below.
*/
listen-on { any; };
listen-on-v6 { none; };
/*
* If the forward directive is set to "only", the server will only
* query the forwarders.
*/
//forward only;
forwarders { 8.8.8.8; };
/*
* Specifies which hosts are allowed to ask ordinary questions.
*/
allow-query { any; };
/*
* This lets "allow-query" be used to specify the default zone access
* level rather than having to have every zone override the global
* value. "allow-query-cache" can be set at both the options and view
* levels. If "allow-query-cache" is not set then "allow-recursion" is
* used if set, otherwise "allow-query" is used if set unless
* "recursion no;" is set in which case "none;" is used, otherwise the
* default (localhost; localnets;) is used.
*/
//allow-query-cache { localnets; };
/*
* Specifies which hosts are allowed to make recursive queries
* through this server. If not specified, the default is to allow
* recursive queries from all hosts. Note that disallowing recursive
* queries for a host does not prevent the host from retrieving data
* that is already in the server's cache.
*/
allow-recursion { any; };
/*

Рисунок 67

Затем осуществляется работа в другом конфиге:

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

Рисунок 68. Открытие конфигурационного файла

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

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

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

Рисунок 69. Настройка конфигурационного файла

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

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

Рисунок 70. Копирование шаблона

После же происходит работа в шаблонах, пути к которым были указаны в конфиге:

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

Рисунок 71. Открытие конфигурационного файла

```
; BIND reverse data file for empty rfc1918 zone
;
; DO NOT EDIT THIS FILE - it is used for multiple zones.
; Instead, copy it and use that copy.
;
$TTL    1D
@       IN      SOA     hq-srv.au-team.irpo. root.hq-srv.au-team.irpo. (
                                2025020600      ; serial
                                12H             ; refresh
                                1H              ; retry
                                1W              ; expire
                                1H              ; ncache
                        )
        IN      NS      hq-srv.au-team.irpo.
        IN      A       192.168.100.2
hq-rtr  IN      A       192.168.100.1
br-rtr  IN      A       192.168.0.1
hq-srv  IN      A       192.168.100.2
hq-cli  IN      A       192.168.200.2
br-srv  IN      A       192.168.0.2

moodle  IN      CNAME   hq-rtr.
wiki    IN      CNAME   hq-rtr.
~
```

Рисунок 72. Настройка конфигурационного файла

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

Рисунок 73. Открытие конфигурационного файла

```
; BIND reverse data file for empty rfc1918 zone
;
; DO NOT EDIT THIS FILE - it is used for multiple zones.
; Instead, copy it and use that copy.
;
$TTL    1D
@       IN      SOA     100.168.192.in-addr.arpa. root.100.168.192.in-addr.arpa. (
                                2025020600      ; serial
                                12H             ; refresh
                                1H              ; retry
                                1W              ; expire
                                1H              ; ncache
                        )
        IN      NS      100.168.192.in-addr.arpa.
        IN      A       192.168.100.2
10      IN      PTR     hq-srv.au-team.irpo.
1       IN      PTR     hq-rtr.au-team.irpo.
~
~
```

Рисунок 74. Настройка конфигурационного файла

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

Рисунок 75. Открытие конфигурационного файла

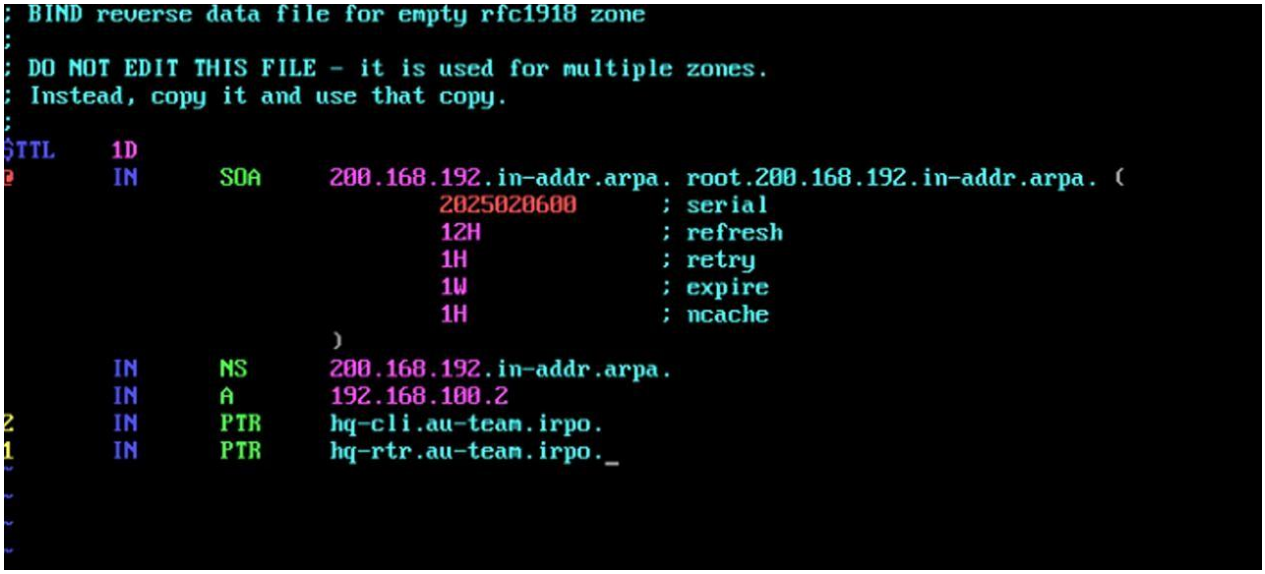

Рисунок 76. Настройка конфигурационного файла

Далее происходит настройка утилиты rndc, дабы bind корректно запускался, с последующим перезапуском службы:

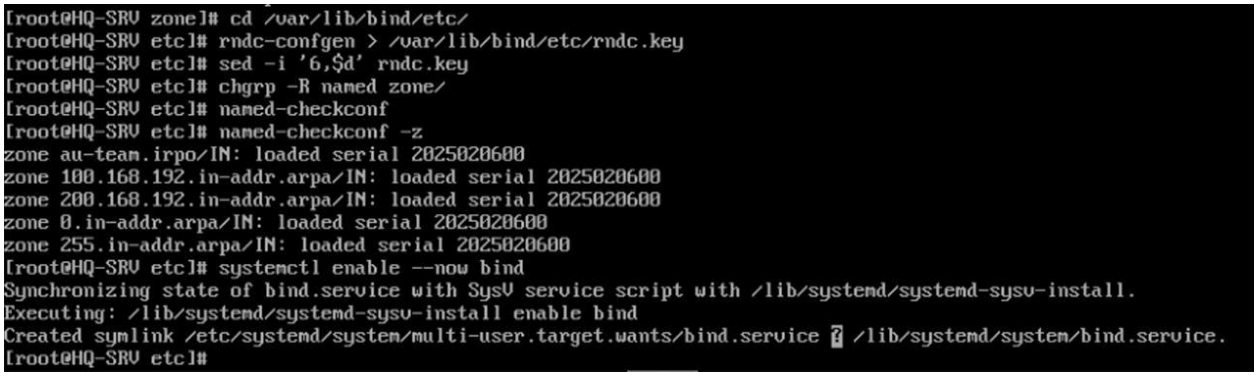

Рисунок 77. Настройка утилиты rndc

После всей настройки, на всех устройствах (кроме ISP) требуется изменить DNS сервер с “8.8.8.8” на “au-team.irpo”.

2.11 Настройка часового пояса на всех устройствах

Настройка часового пояса на всех устройствах производится одной командой. Так как на всех устройствах будет произведена одна и та же настройка, будет продемонстрирована работа этой команды на HQ-RTR и HQ-CLI:

HQ-RTR:

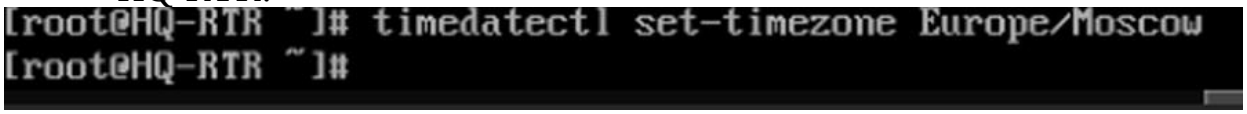

Рисунок 78. Настройка времени
HQ-CLI:

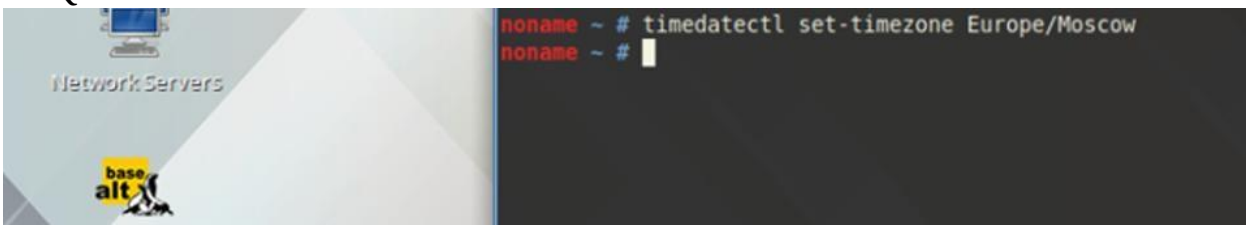

Рисунок 79. Настройка времени

По итогу всей проделанной работы, в корпоративной сети каждое устройство имеет выход в сеть интернет, а также сети HQ и BR способны пересылать между собой пакеты.

З А К Л Ю Ч Е Н И Е

В теоретической части описывается, что система резервного копирования и аварийного восстановления имеется почти в каждой организации, где ведется работа с вычислительными машинами. Хорошо налаженная система резервного копирования и аварийного восстановления данных, залог успеха и процветания организации.

Так же описана работа системы резервного копирования и аварийного восстановления данных на Linux. Помимо этого, было прописано создание такой системы.

В практической части была реализована смоделированная корпоративная сеть согласно заданной топологии:

1. Описание всех устройств в сети, а также последующая раздача IP адресов.

2. Описание сети, а также используемых утилит.

3. Настройка ISP: раздача ip адресов ISP, HQ-RTR, BR-RTR, HQ-SRV, BR-SRV, с последующей настройкой динамической трансляцией при помощи iptables.

4. Создание локальных учетных данных. Были созданы пользователи sshuser в HQ-SRV и BR-SRV, и net_admin в HQ-RTR и BR-RTR.

5. Настройка на интерфейсе HQ-RTR в сторону офиса HQ виртуального коммутатора, с использованием утилиты openvsvitch.

6. Настройка безопасносного удаленного доступа на серверах в сети HQ и BR.

7. Конфигурация IP туннеля между офисами HQ и BR: было выполнено при помощи псевдографической утилиты nmtui.

8. Настройка динамической маршрутизации между офисами: было выполнено при помощи утилиты frr.

9. Настройка динамический адресов маршрутизаторов HQ и BR: было выполнено при помощи iptables.

10. Настройка протокола динамической конфигурации хостов: было выполнено при помощи утилиты dhcp-сервер.

11. Настройка DNS для офисов HQ и BR: было выполнено при помощи утилиты bind.

12. Проведена настройка часового пояса у всех устройств.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Безопасна ли операционная система Linux? [Электронный ресурс] URL: https://www.kaspersky.ru/resource-center/definitions/linux?ysclid=mb9i7orquu203520650 (Дата обращения: 16.05.2025).
2. Что такое OS Linux: чья эта ОС и для чего нужна – преимущества операционной системы Линукс [Электр
3. онный ресурс] – URL: https://www.cleverence.ru/articles/auto-busines/os-linux-chto-eto-za-operatsionnaya-sistema-i-eye-klyuchevye-preimushchestva-i-nedostatki/ (Дата обращения: 16.05.2025).
4. Что такое Linux: полный гид по выбору дистрибутива и установке — Журнал «Код» [Электронный ресурс] – URL: https://thecode.media/linux-2/?ysclid=mb9idb7k7362203314 (Дата обращения: 16.05.2025).
5. Система резервного копирования [Электронный ресурс] – URL: https://www.tadviser.ru/index.php/Статья:Система_резервного_копирования?ysclid=mb9ifjhvm2943831020 (Дата обращения: 16.05.2025).
6. Что представляет собой система резервного копирования данных [Электронный ресурс] – URL: https://kitsvc.ru/blog/chto-takoe-sistema-rezervnogo-kopirovaniya (Дата обращения: 17.05.2025).
7. Что такое Disaster Recovery и DRP | Технология работы | Cloud4Y [Электронный ресурс] – URL: https://www.cloud4y.ru/blog/what-is-disaster-recovery/ (Дата обращения: 17.05.2025).
8. Резервное копирование и аварийное восстановление как технологии защиты данных [Электронный ресурс] – URL: https://blog.ishosting.com/ru/backup-disaster-recovery?utm_source=yandex&utm_medium=organic&utm_campaign=seo (Дата обращения: 17.05.2025).
9. Лучшие 15 инструментов для восстановления данных в Linux [Электронный ресурс] – URL: https://blog.sedicomm.com/2019/11/07/luchshie-15-instrumentov-dlya-vosstanovleniya-dannyh-v-linux/ (Дата обращения: 17.05.2025).
10. Программы резервного копирования Linux - Losst [Электронный ресурс] – URL: https://losst.pro/programmy-rezervnogo-kopirovaniya-linux (Дата обращения: 18.05.2025).
11. 14 утилит резервного копирования для Linux-систем [Электронный ресурс] – URL: https://blog.sedicomm.com/2018/07/22/14-utilit-rezervnogo-kopirovaniya-dlya-linux-sistem/?ysclid=mb9ioqr25t978318708 (Дата обращения: 18.05.2025).