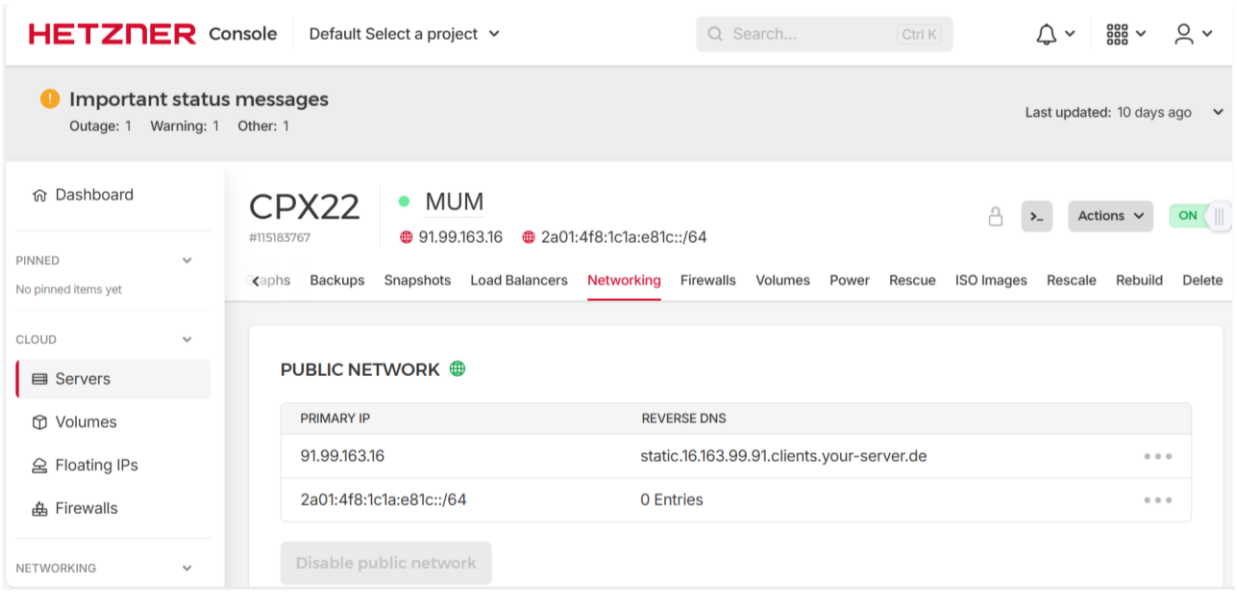
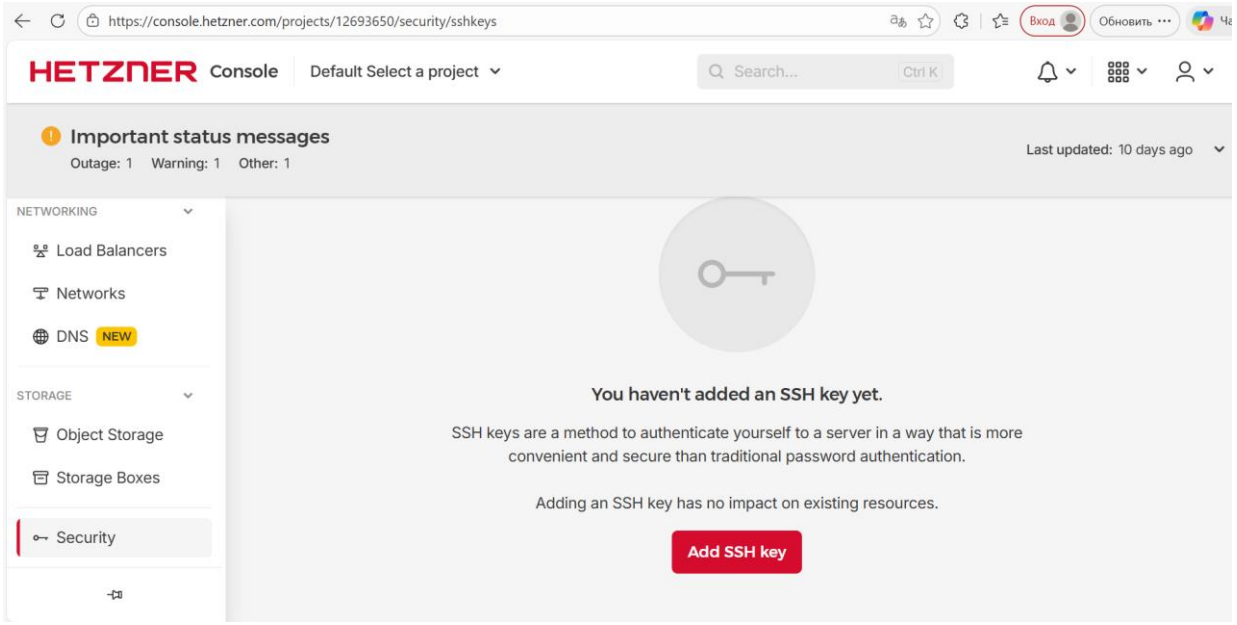
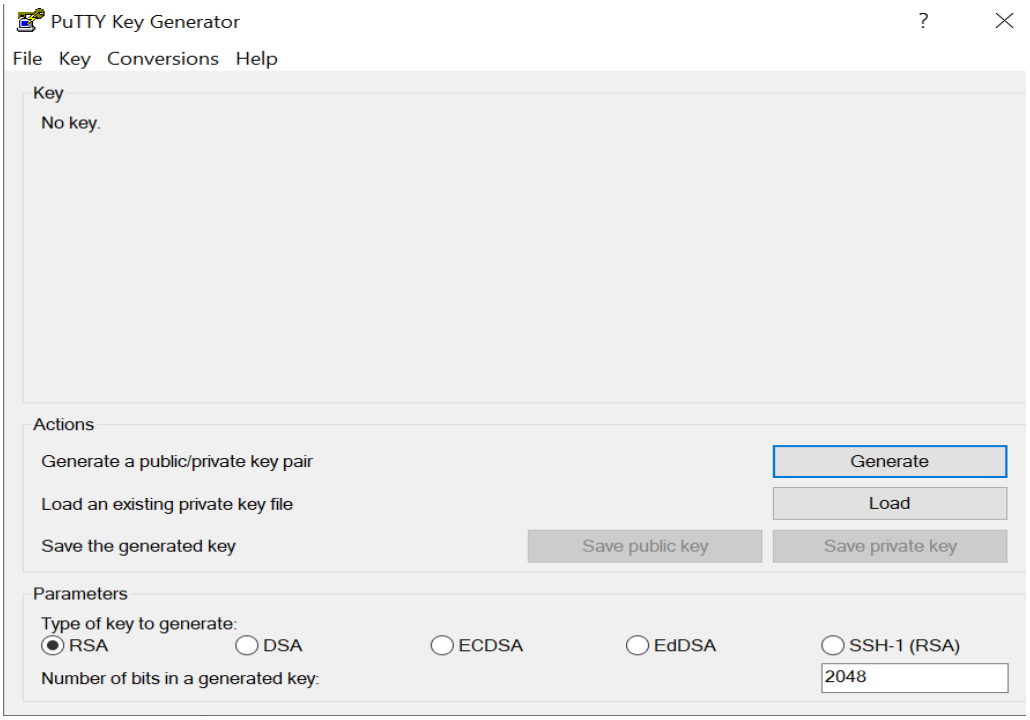
Linux Server Recovery and Port Validation



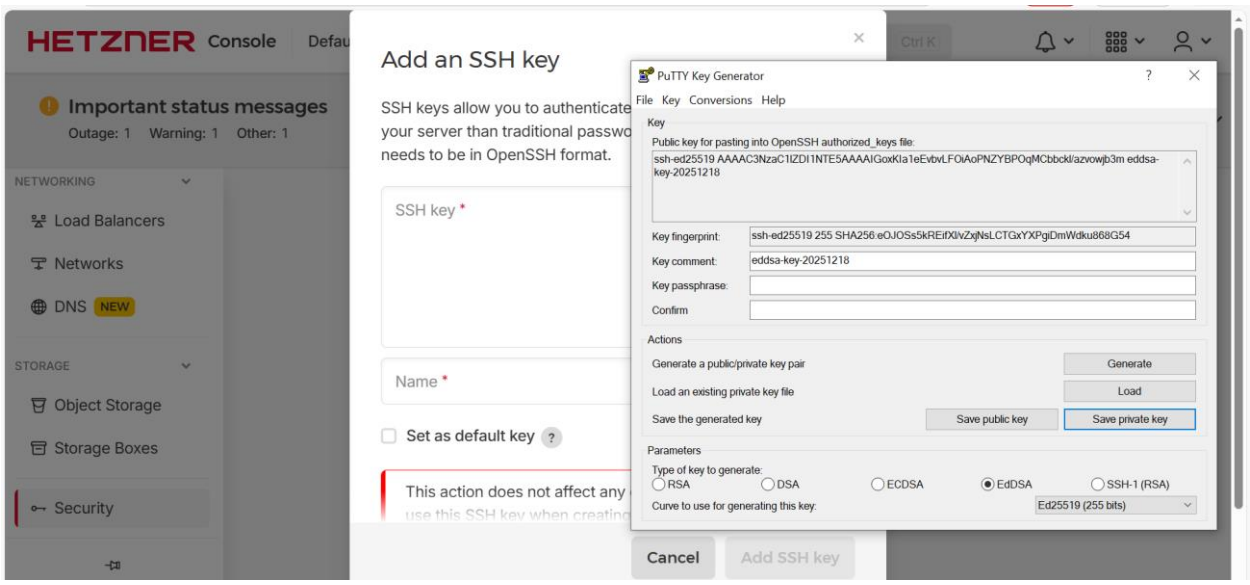
В панели Hetzner открываем вкладку Networking и смотрим сетевые параметры сервера (Public Network). Мы видим публичный IPv4 сервера **91.99.163.16**, а также IPv6 и Reverse DNS — это адрес, по которому будем подключаться по SSH и проверять доступность портов и сервисов.



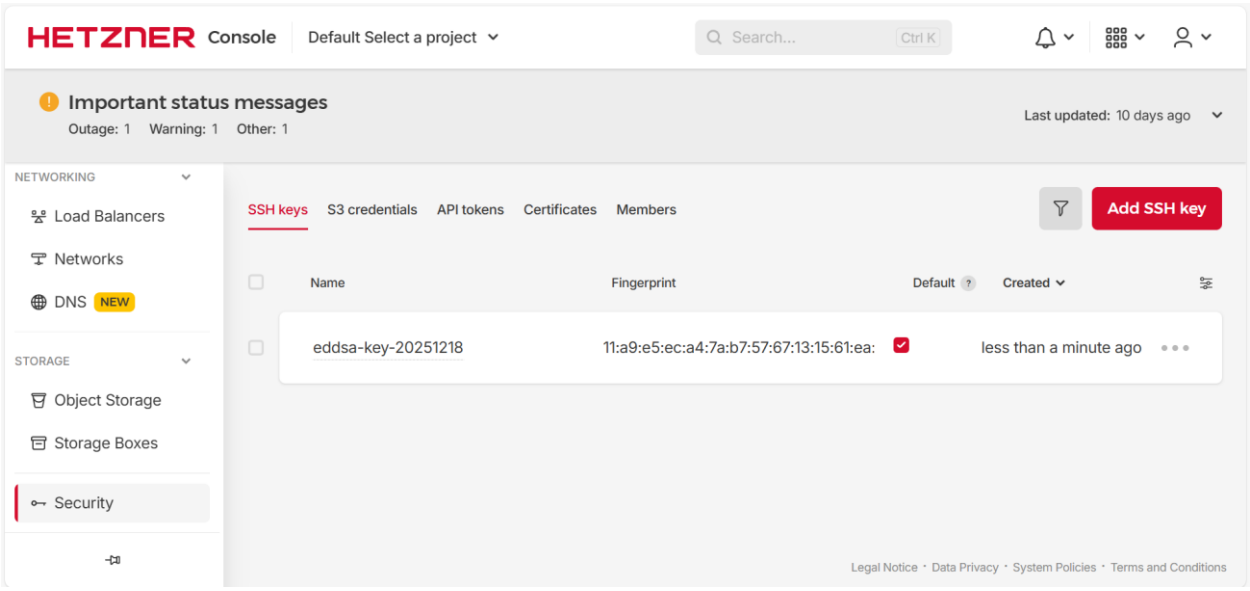
В панели Hetzner переходим в раздел **Security - SSH keys**. Мы видим, что SSH-ключей ещё нет, поэтому сервер пока не настроен для входа по ключу. Это означает, что нам необходимо создать и добавить SSH-ключ, чтобы обеспечить безопасную аутентификацию.



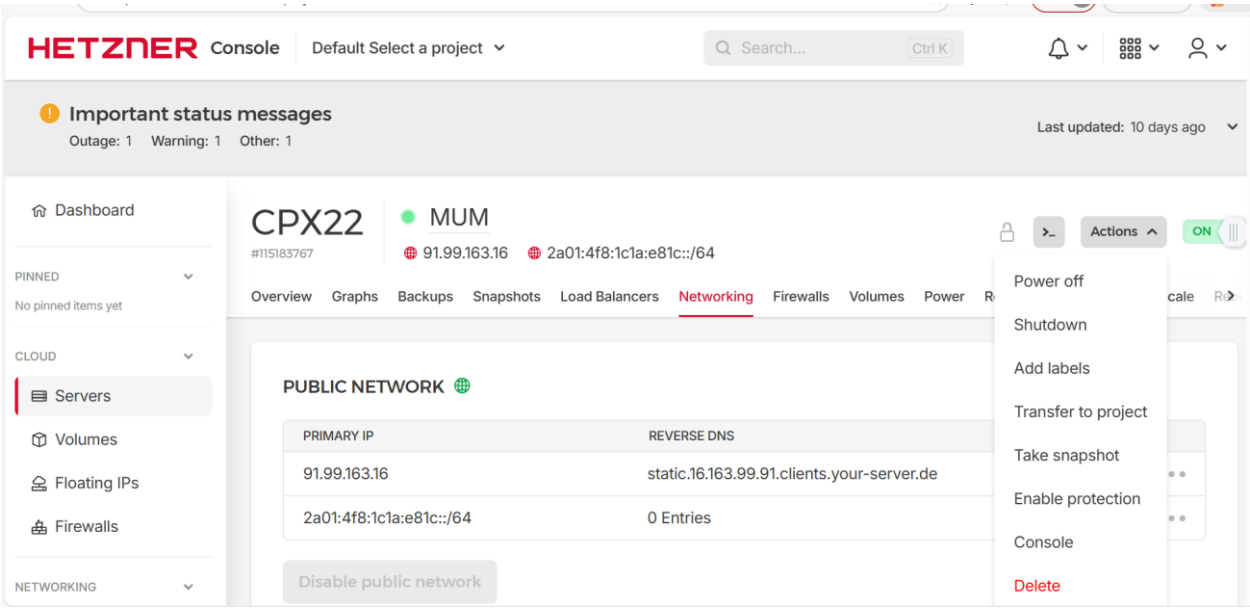
Открываем программу **PuTTY Key Generator (PuTTYgen)** для создания SSH-ключа. На этом этапе мы генерируем пару ключей — публичный и приватный. Публичный ключ будем добавлять в панель Hetzner, а приватный останется у нас для безопасного подключения к серверу по SSH.



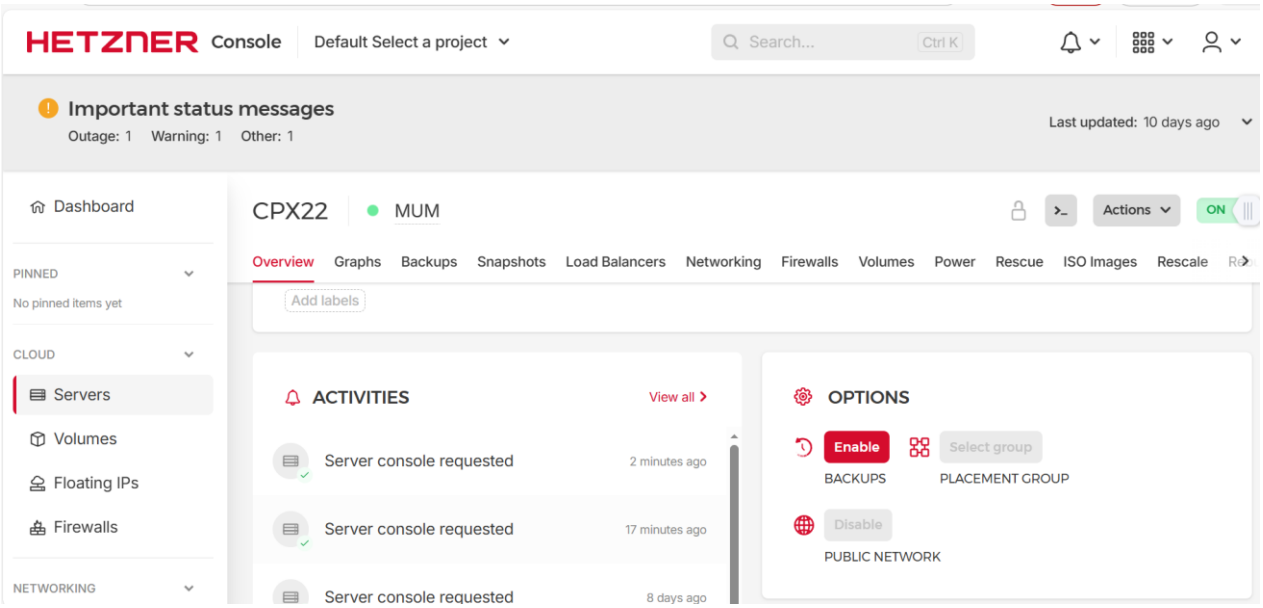
После генерации ключа в PuTTYgen копируем **публичную часть SSH-ключа** и вставляем её в форму добавления ключа в панели Hetzner (Add SSH key). Таким образом мы регистрируем наш публичный ключ в облачной панели, чтобы в дальнейшем использовать его для безопасного доступа к серверу.

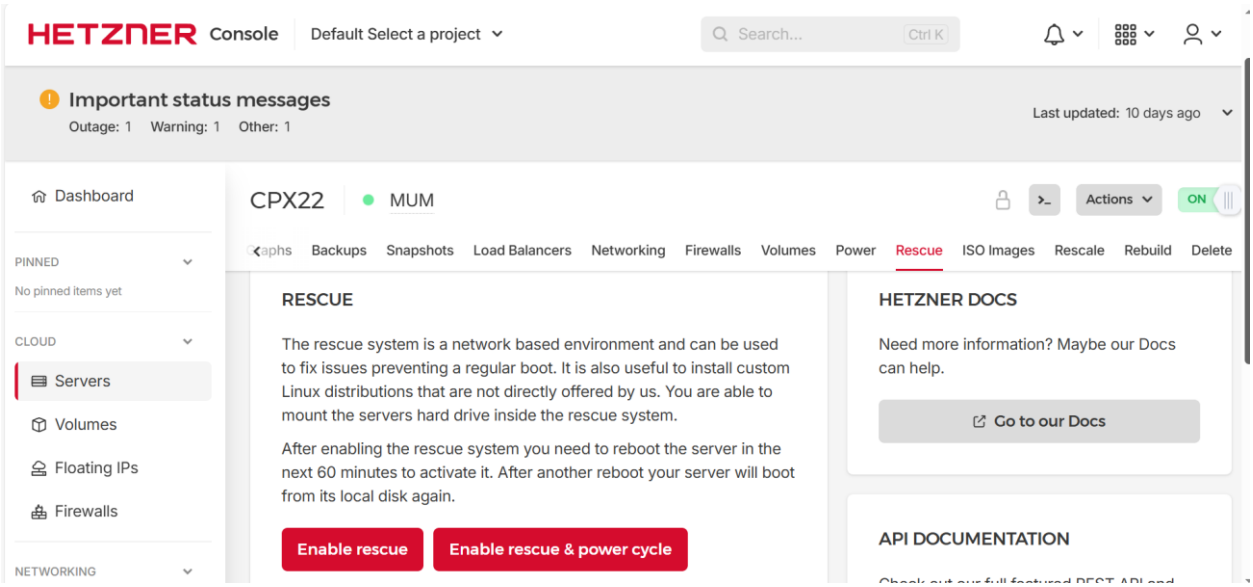


После добавления ключа возвращаемся в раздел **SSH keys** и видим, что новый ключ появился в списке. Это означает, что публичный SSH-ключ успешно сохранён в панели Hetzner и теперь может использоваться для доступа к серверу или для загрузки в режиме Rescue.

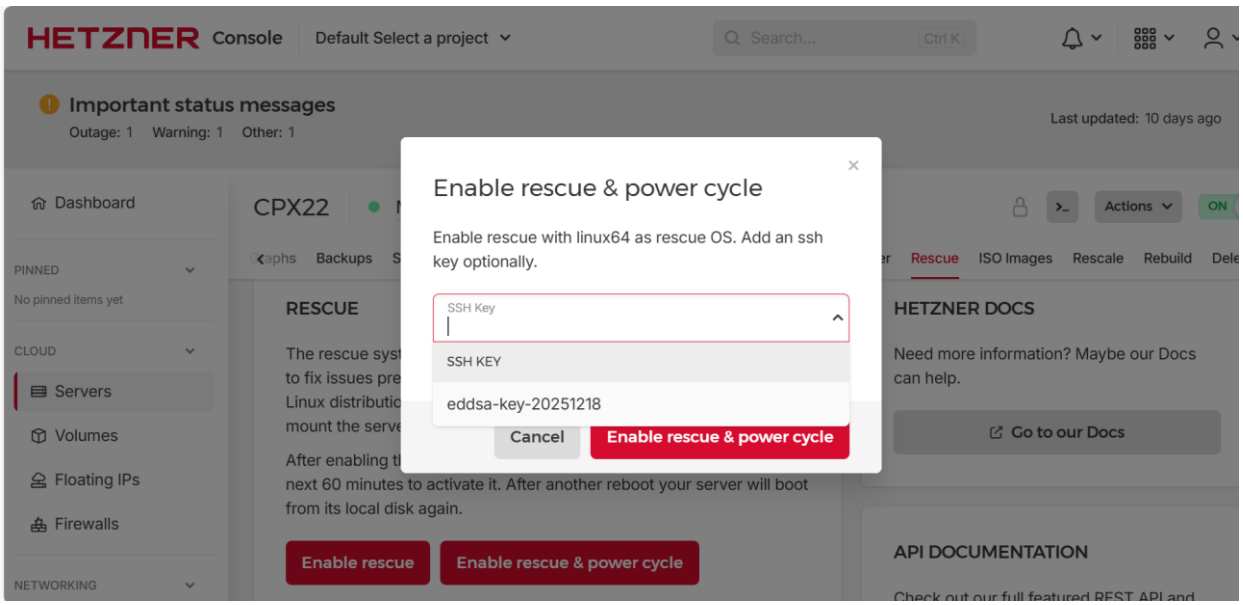


В панели управления сервером открываем раздел **Actions**, где доступны операции управления сервером (перезагрузка, выключение, консоль, Rescue и другие действия). Здесь мы готовимся перейти в режим Rescue, чтобы получить доступ к системе для восстановления или настройки SSH-доступа.

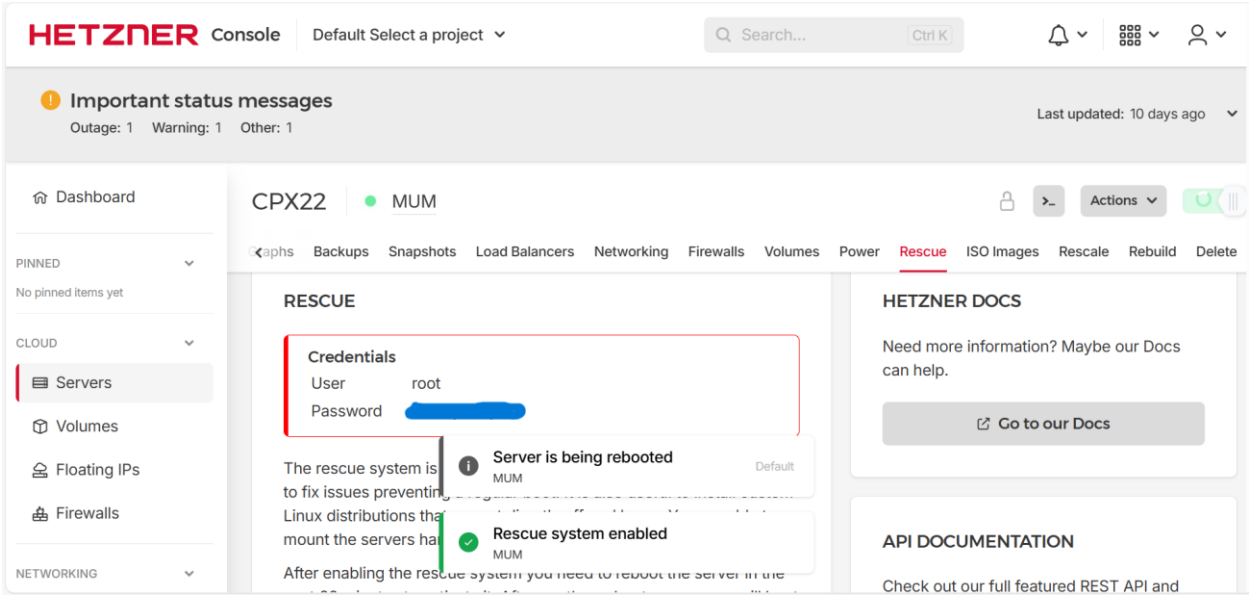




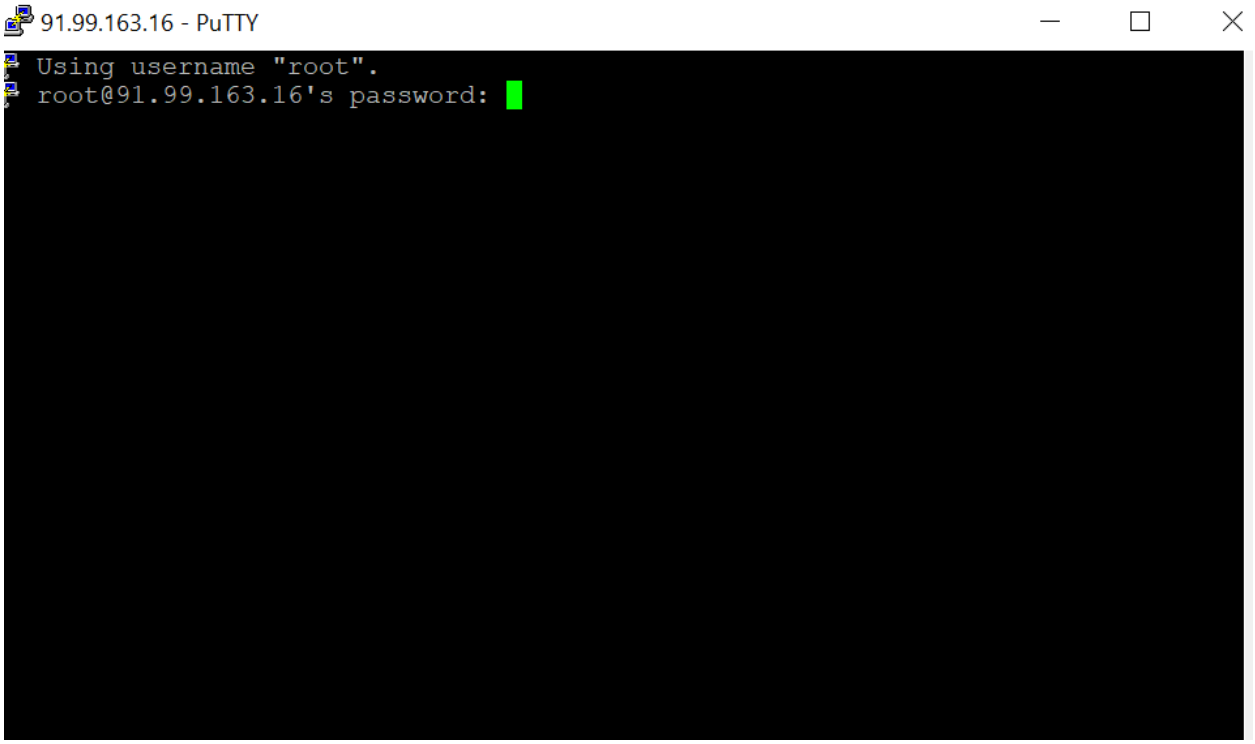
Переходим во вкладку **Rescue** в панели Hetzner. Здесь можно включить режим аварийной загрузки сервера. Rescue-система — это отдельная временная операционная система, которая загружается вместо основной и позволяет получить доступ к дискам сервера для восстановления системы или изменения конфигурации.



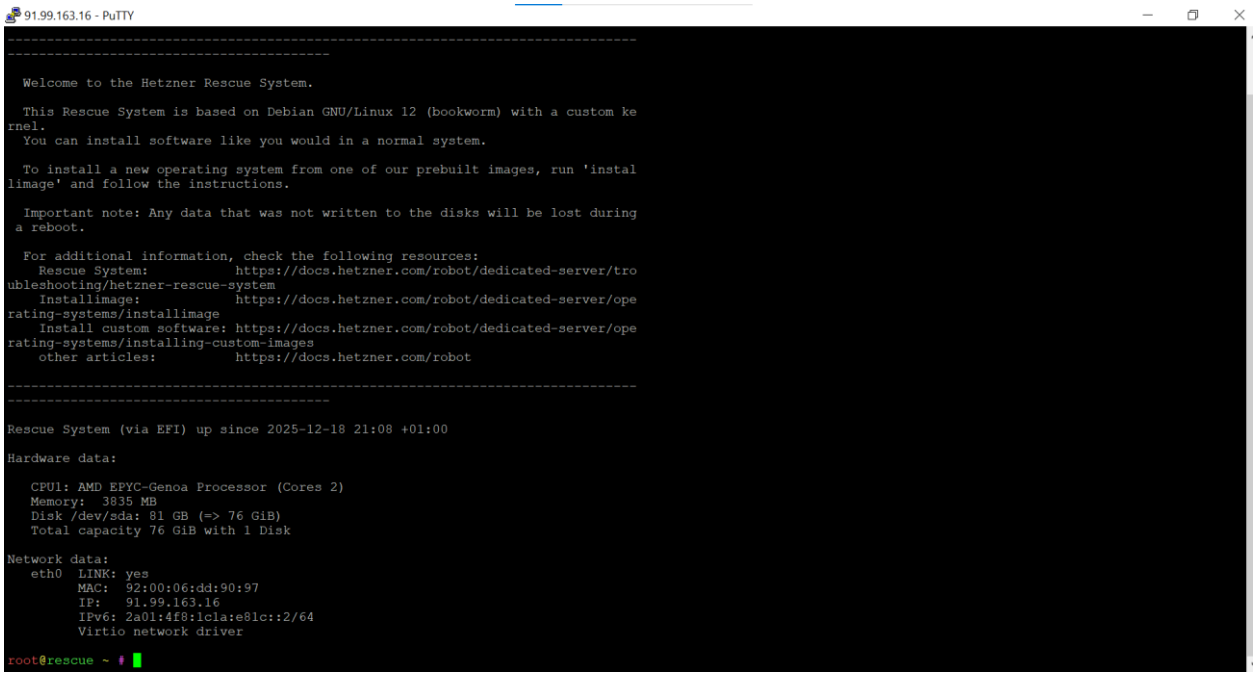
Во вкладке **Rescue** включаем режим аварийной загрузки и выбираем ранее добавленный SSH-ключ. После этого выполняем **Power cycle**, чтобы сервер перезагрузился и загрузился уже в Rescue-системе. Таким образом мы получаем возможность подключиться к серверу в специальном режиме для восстановления доступа.



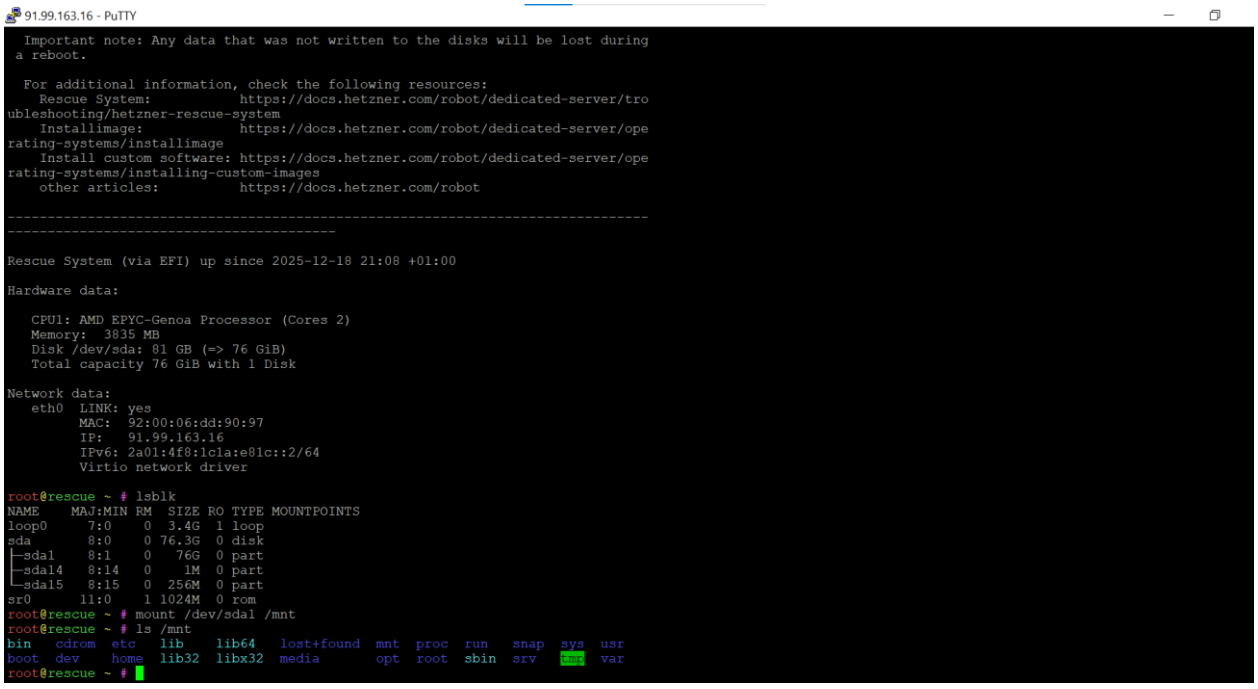
После активации Rescue-системы панель Hetzner показывает подтверждение включения режима и временные данные для входа (root и пароль). Это означает, что сервер теперь загружен в Rescue-окружение, и мы можем подключиться к нему по SSH для выполнения восстановительных действий.



Подключаемся к серверу по SSH через PuTTY, указывая публичный IP-адрес сервера и пользователя root. Система запрашивает пароль, выданный в режиме Rescue. Это шаг аутентификации, после которого мы получим доступ к серверу в аварийном режиме.



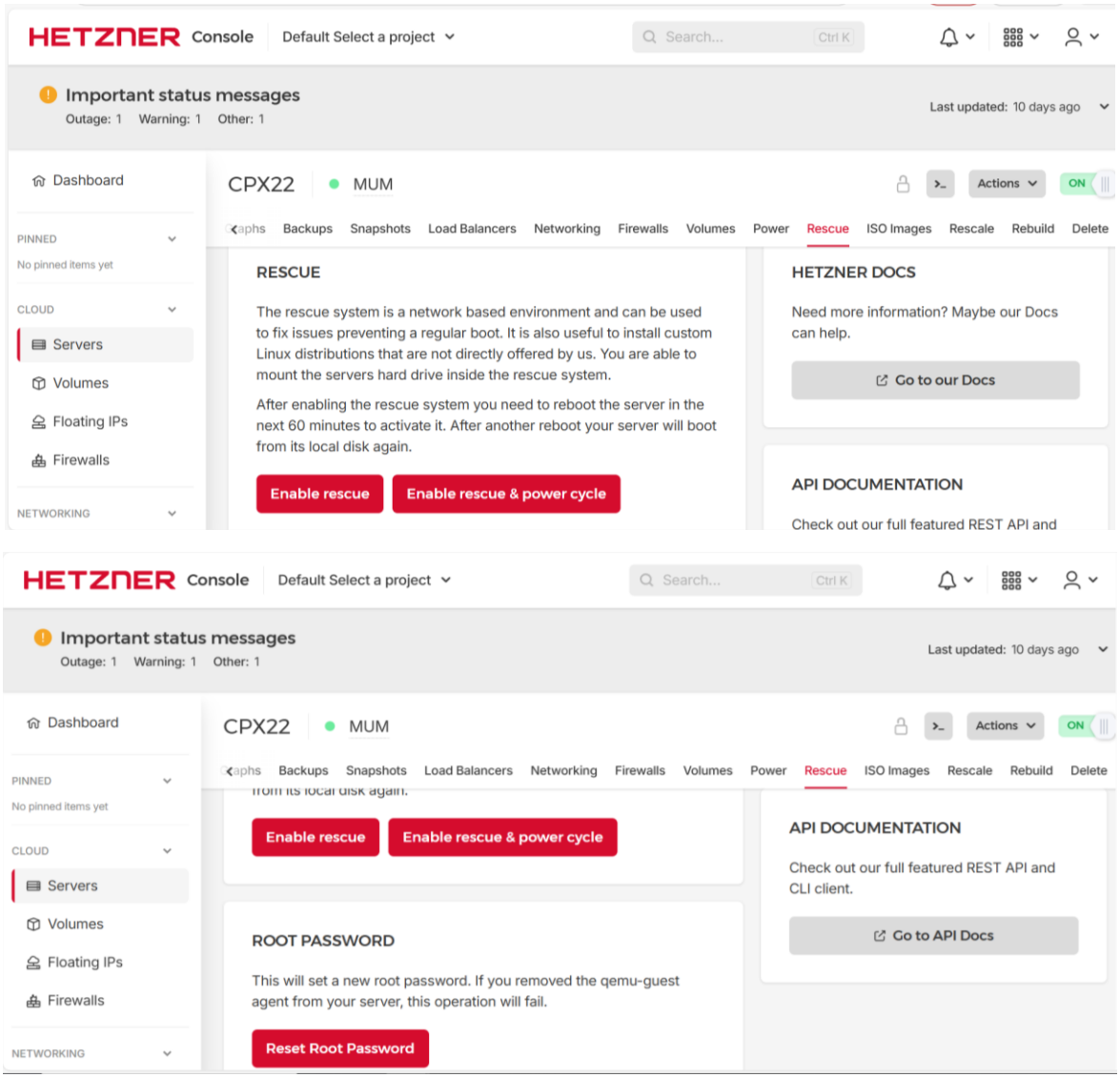
После успешной аутентификации мы видим приветственное сообщение **Hetzner Rescue System**. Это подтверждает, что сервер загружен не в основную операционную систему, а в аварийное Rescue-окружение.



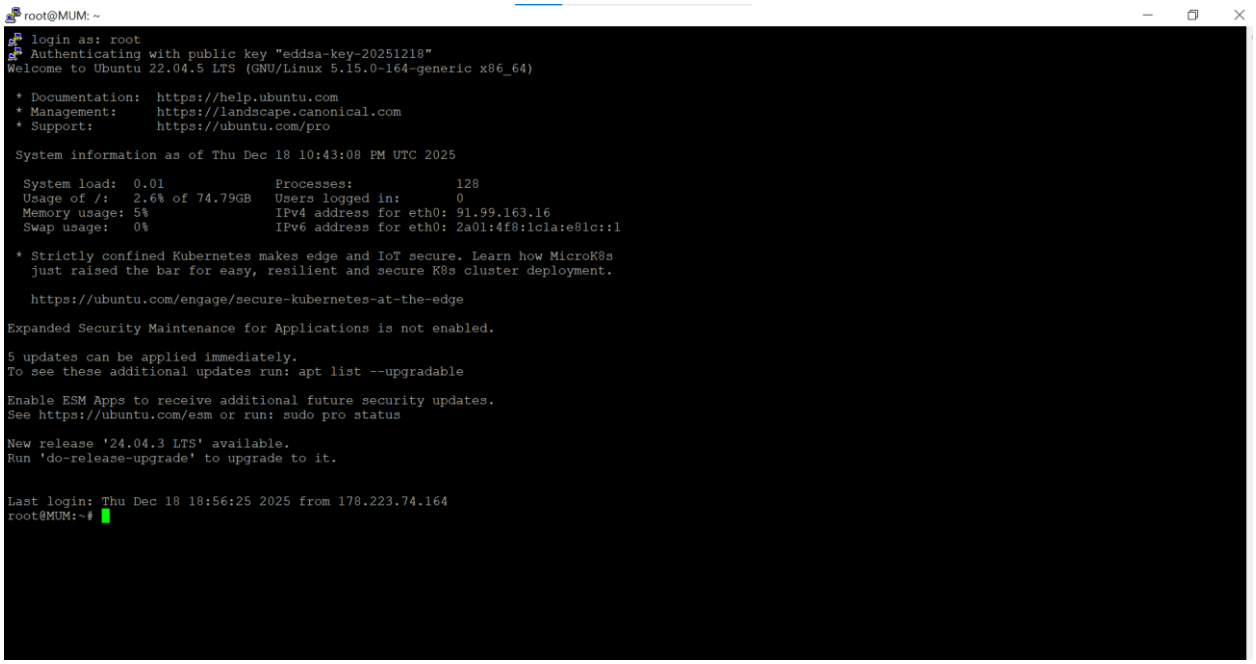
Rescue-системе выполняем команду lsblk, чтобы посмотреть список дисков и разделов сервера. Затем монтируем основной раздел системы, /dev/sda1, в каталог /mnt с помощью команды mount. После этого проверяем содержимое каталога /mnt и убеждаемся, что видим стандартную структуру Linux (bin, etc, home и другие директории). Это означает, что мы успешно подключили файловую систему основной операционной системы и можем вносить в неё изменения.

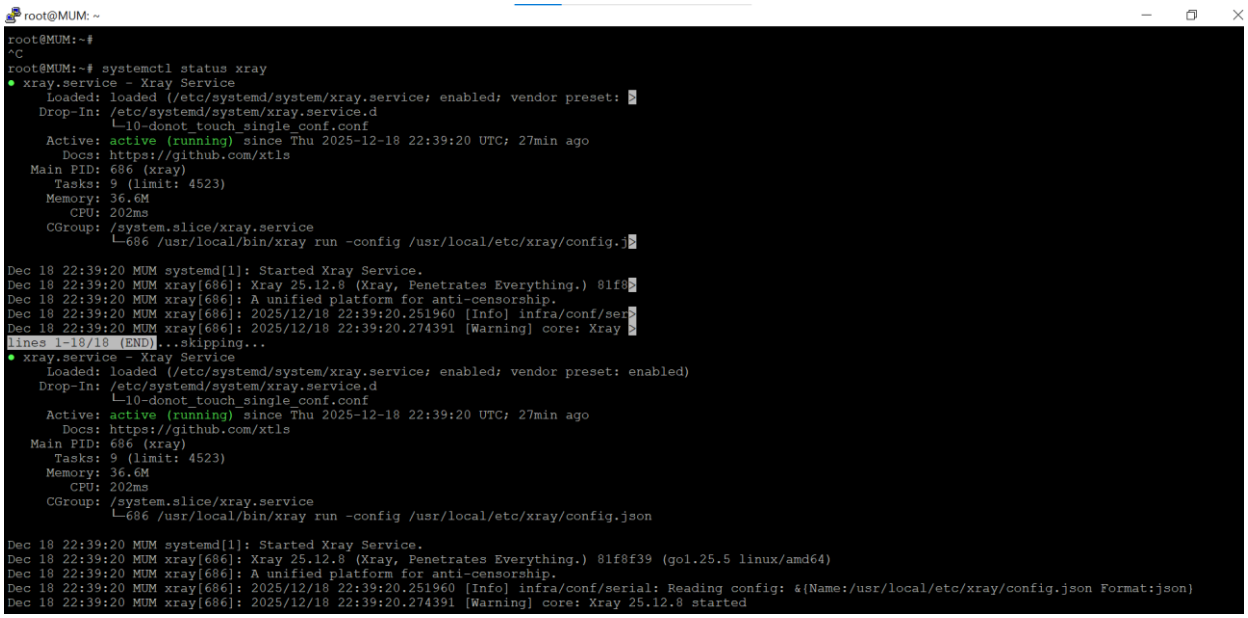
В Rescue-системе создаём каталог /mnt/root/.ssh, если он отсутствует, затем создаём файл authorized_keys с помощью команды touch. После этого открываем файл и вставляем в него наш публичный SSH-ключ. Далее устанавливаем корректные права доступа: chmod 600 для файла authorized_keys, чтобы обеспечить безопасность SSH-аутентификации. В конце выполняем reboot, чтобы сервер перезагрузился и применил изменения.

Таким образом, мы вручную добавляем SSH-ключ в основную систему, чтобы в дальнейшем входить на сервер без использования временного rescue-пароля.

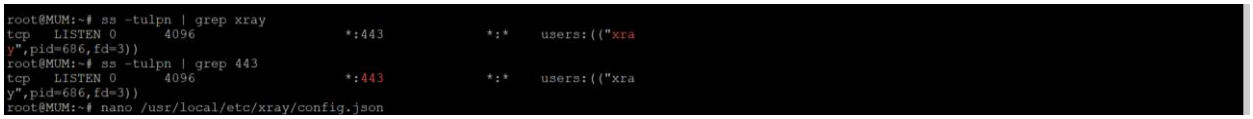


После перезагрузки сервер выходит из Rescue-режима и загружается в основную операционную систему. Мы снова подключаемся по SSH к серверу, и теперь вход выполняется с использованием ранее добавленного публичного ключа. Это подтверждает, что SSH-доступ успешно восстановлен и сервер работает в штатном режиме.

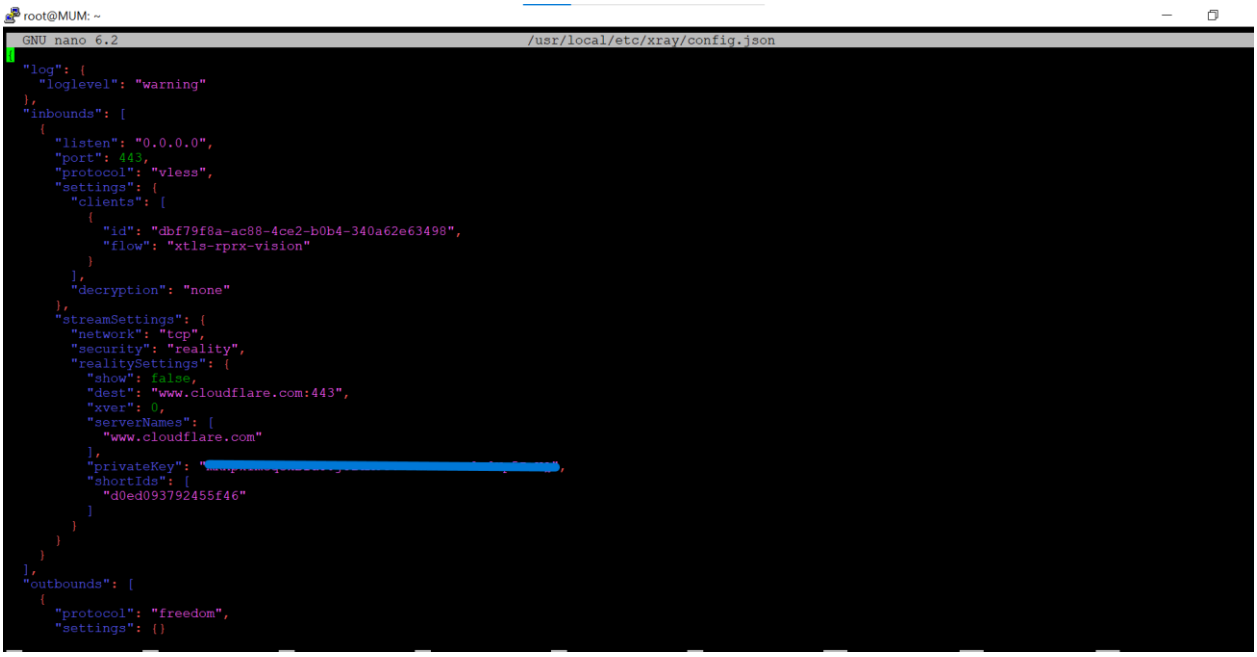




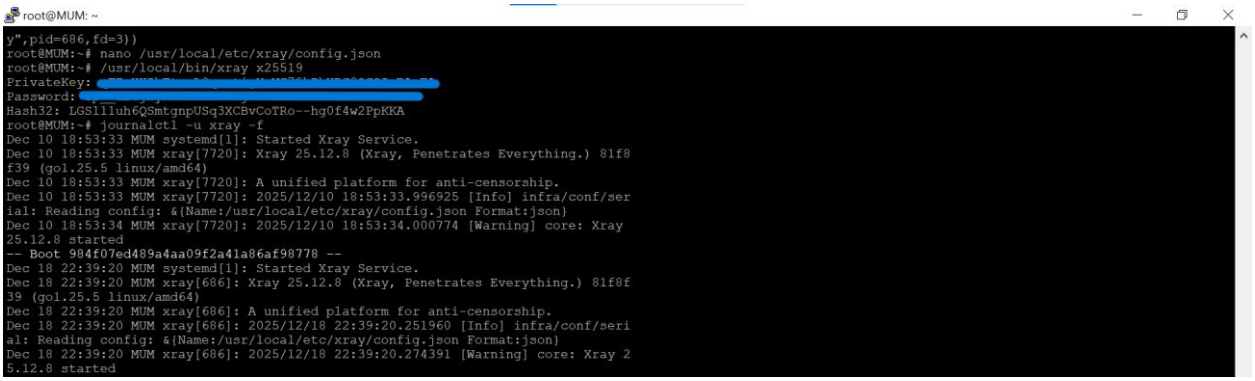
После успешного входа в основную систему Ubuntu мы проверяем состояние сервиса **xray** с помощью команды systemctl status xray. В выводе видно, что сервис загружен (loaded), включён для автозапуска (enabled) и находится в состоянии active (running). Это подтверждает, что служба запущена корректно и работает без критических ошибок.



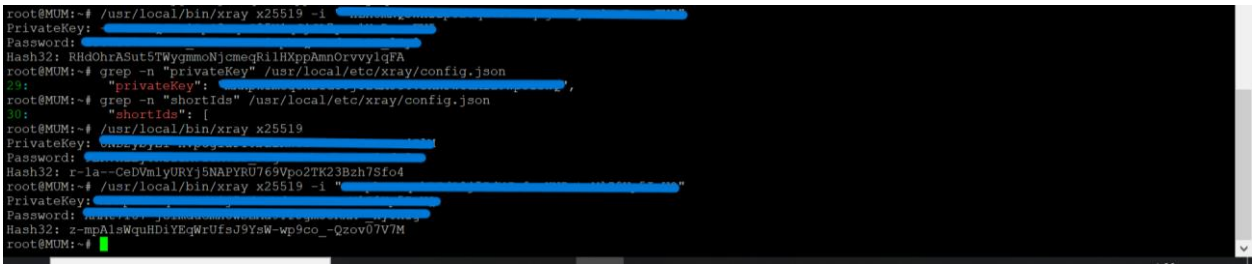
Проверяем, прослушивает ли сервис нужный порт, с помощью команды ss -tulpn | grep 443. В выводе видно, что процесс **xray** слушает порт **443** на всех интерфейсах. Это означает, что сервис корректно привязан к HTTPS-порту и готов принимать входящие подключения из сети.



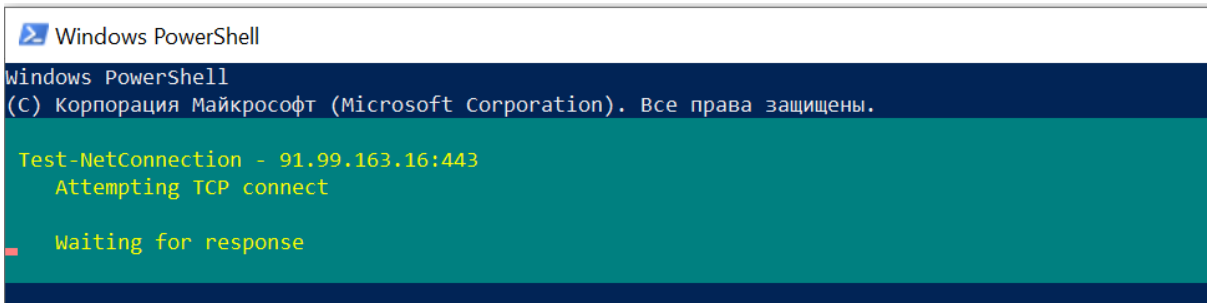
Открываем конфигурационный файл сервиса Xray с помощью команды nano /usr/local/etc/xray/config.json. В файле проверяем параметры входящего подключения: порт 443, сетевые настройки и параметры безопасности. Это позволяет убедиться, что сервис настроен корректно и использует нужные значения для обработки входящих соединений.

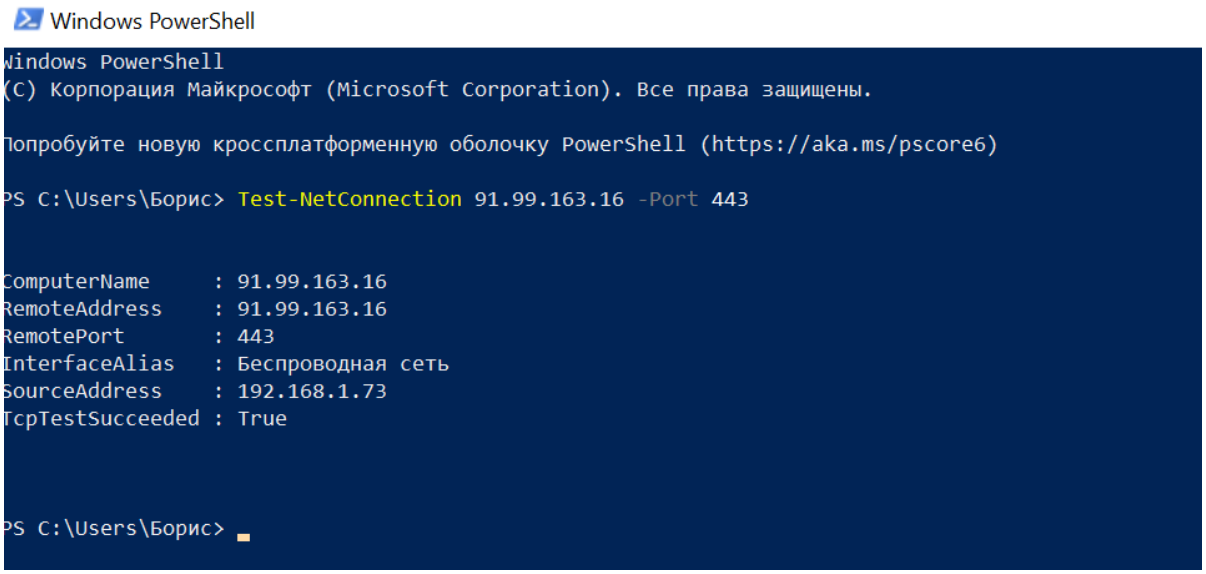


Проверяем логи сервиса с помощью команды journalctl -u xray -f. В режиме реального времени наблюдаем сообщения о запуске и работе службы. Отсутствие критических ошибок в логах подтверждает, что сервис функционирует корректно и обрабатывает подключения без сбоев.



Проверяем работу сервиса и криптографические параметры, используя встроенные команды Xray для генерации ключей и просмотра конфигурации. Это позволяет убедиться, что необходимые параметры безопасности (ключи и идентификаторы) корректно заданы в конфигурационном файле.





С локального компьютера выполняем проверку доступности сервера с помощью команды PowerShell Test-NetConnection 91.99.163.16 -Port 443. В выводе отображается TcpTestSucceeded : True, что означает успешное установление TCP-соединения с сервером на порту 443. Это подтверждает, что порт открыт, сервер доступен из интернета и сервис корректно принимает входящие подключения.

Таким образом, мы подтверждаем, что сервис запущен, слушает порт 443 и настроен с корректными криптографическими параметрами для безопасной работы.