СИСТЕМА ХРАНЕНИЯ ДАННЫХ

РУКОВОДСТВО ПО НАСТРОЙКЕ TROK

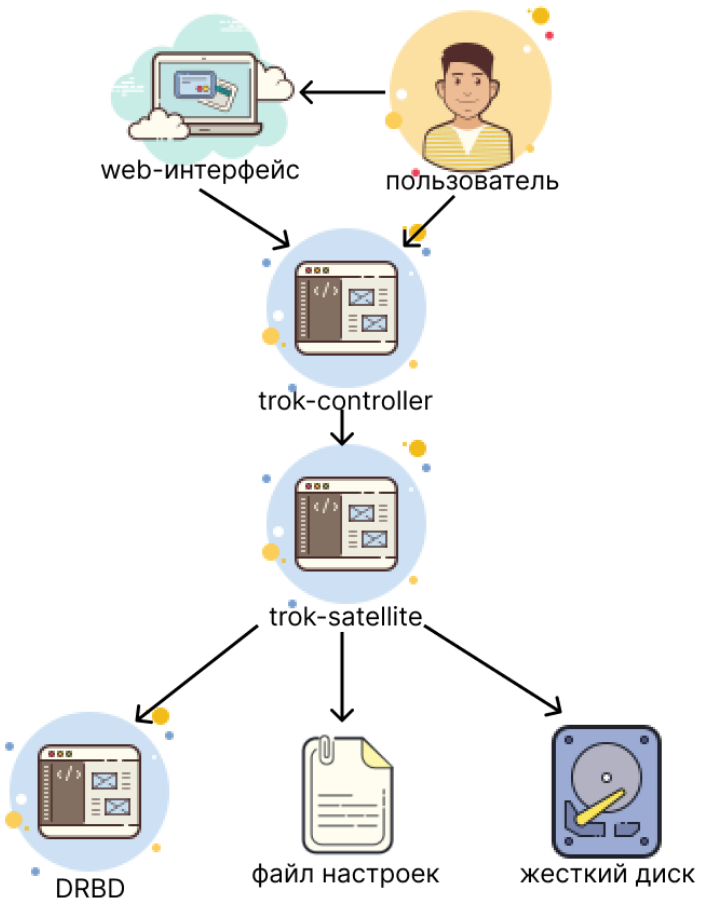
Москва, 2025г.

1. Общая информация

Руководство по настройке ллюстрирует процесс создания кластерной системы хранения данных на примере конфигурации, состоящей из трёх узлов. Кластер представляют собой три физических сервера с установленной операционной системой Astra Linux. В кластере рекомендуется использовать не менее трёх серверов для обеспечения устойчивости к сбоям и достижения кворума при принятии решений. Подобная конфигурация минимизирует риск разделения сети и повышает общую отказоустойчивость системы. Все серверы подключены к одной локальной подсети, обеспечивая сетевое взаимодействие внутри единой инфраструктуры. Для обеспечения избыточности и повышенной отказоустойчивости системы репликация данных осуществляется между двумя из указанных узлов.

Техническая реализация программного обеспечения включает интеграцию четырех основных компонентов:

- trok-controller – центральный модуль, отвечающий за оркестрацию кластерных ресурсов и управление конфигурацией распределенного хранилища данных.
- trok-satellite – контроллер, обеспечивающий взаимодействие между узлами кластера.
- TROK WEB GUI – веб-интерфейс для интуитивного администрирования функций системы хранения данных.
- DRBD-reactor – автономная служба для мониторинга событий на DRBD-устройствах и автоматического выполнения действий в ответ на изменения их состояния.



Рсунок 1 – Взаимодействие основных компонентов программного обеспечения

Перед началом конфигурации необходимо обеспечить соблюдение критических условий.

1.1. Установка и настройка базовых компонентов

Базовые компоненты, необходимые, для работы системы указаны в таблице ниже.

Компонент	Требования	Проверка выполнения
LINSTOR (DRBD + LVM)	DRBD версии ≥ 9.0 LVM2 или LVM Thin Provisioning Ядро Linux ≥ 4.14 с поддержкой DRBD	sudo linstor –version sudo lvs --version sudo drbdsetup –version uname -r проверка наличия модуля lsmod | grep drbd
DRBD Reactor	Python ≥ 3.6 Доступ к API LINSTOR Конфигурационные файлы в /etc/drbd-reactor/	sudo drbd-reactor --version
WEB-UI	Подключение к API LINSTOR	sudo systemctl status nginx sudo systemctl status apache2

	Веб-сервер (Nginx/Apache) с HTTPS Порт 8080 (или кастомный)	
Linstor-gateway	Linstor-gateway 1.0.0	linstor-gateway check-health --iscsi-backends lio netstat -tulnp | grep 3260 (для подтверждения, что порт не занят, вывод команды должен быть пустым) rpcinfo -p
iSCSI (LIO)	Целевая служба targetcli Открытые порты 3260/TCP LUN на основе DRBD-устройств	
NVMe-oF	Поддержка NVMe в ядре Конфигурация nvmet Сетевой транспорт (TCP/RDMA)	
NFS (опционально)	Сервер nfs-kernel-server Экспорт томов через /etc/exports Порты 111, 2049/TCP	

1.2. Подготовка топологии сети и соблюдение аппаратых требований

Для комфортной работы желательно использовать минимум 3 серверные ноды, имеющие следующие минимальные характеристики:

- Процессор: 4+ ядер (x86_64);
- Размер оперативной памяти: 8+ GB.

Каждая нода должна иметь неразмеченный диск, например, /dev/nvme0n1, с RAW-емкостью не менее 100 ГБ.

1.1 Сетевые интерфейсы

Рекомендации по сорости соединения:

- Минимум: 1 Гбит/с (1 GbE);
- Рекомендуется: 10 Гбит/с (10 GbE) и выше (особенно для репликации DRBD, чтобы избежать задержек при синхронизации данных между узлами)

Максимальный размер блока данных (в байтах), который может быть передан через сетевой интерфейс без фрагментации (размер MTU):

- Минимум: 1500 байт (стандартный Ethernet);
- Рекомендуется: Jumbo Frames (MTU ≥ 9000) (рекомендуется для NVMe-oF, так как увеличивает пропускную способность и снижает накладные расходы при передаче данных).

1.3. Сетевые требования

Все ноды должны находиться в одной подсети без NAT (например, 192.168.100.2/24).

Каждое устройство должно иметь статический IP-адрес или использовать резервирование DHCP.

1.4. Резервное копирование данных

Перед началом любых работ с настройкой системы хранения данных важно сделать резервное копирование. Это поможет избежать проблем, таких как случайное удаление разделов или повреждение данных из-за неправильных команд или неожиданных ошибок.

.

2. Настройка узла через CLI-интерфейс LINSTOR

Для создания узла с типом COMBINED (который объединяет функции контроллера и сателлита), нужно выполнить команду, указанную ниже:

```
sudo linstor node create --node-type combined <имя_узла> <ip_адрес_узла>
```

--node-type combined – указывает, что нода будет:

Комбинированной: гибридная роль, при которой узел выполняет функции контроллера и сателлита одновременно.

Она также может быть

Контроллером: управляет кластером (хранит метаданные, координирует репликацию).

Сателлитом: хранит данные и участвует в репликации DRBD.

Имя узла (должно быть уникальным в кластере).

В случае, если узел имеет локальное имя программа автоматически его определит и будет использовать в дальнейшем для конфигурации ресурсов DRBD. В данной операции не будет участвовать заданное указанной выше командой произвольное имя узла, но возможность задать альтернативное имя в системе все же присутствует. Чтобы избежать путаницы для себя и других пользователей, рекомендуется при создании узла использовать имя хоста узла.

При выборе уникального имени узла (ноды) следует учитывать следующие правила:

- разрешенные базовые символы – латинские буквы: a-z, A-Z; цифры: 0-9; спецсимволы: - (дефис); . (точка).
- начало имени не может начинаться со спецсимвола (например, имя .server недопустимо)
- длина уникального имени должна составлять не более 48 символов.
- регистр букв не учитывается (например, Node1 и node1 считаются идентичными).
- запрещенные символы и форматы – пробелы, кириллица, символы: _, @, #, $, %, ^, &, *, (), {}, [], /, \, :, ;, ", ', ~, !.
- запрещено использовать IP-адреса в качестве имён (например, 192.168.1.1 недопустимо).

Если Ваши имя ноды в системе TROK и имя устройства (хоста) отличаются, можно воспользоваться следующими сценариями:

- изменить имя устройства, используйте команду:
  ```
  sudo hostnamectl set-hostname <имя_узла>
  ```
  Важно. Данная операция может повлечь за собой необходимость корректировки конфигурационных файлов иных программных компонентов, установленных в операционной системе. Поэтому перед выполнением изменения следует тщательно оценить потенциальное влияние на функционирование программного обеспечения и убедиться в безопасности данной процедуры для текущих процессов.

- изменить имя ноды в системе TROK, но система не позволяет переименовать узел напрямую, так как имя узла жестко связано с его идентификатором в базе данных контроллера, поэтому можно воспользоваться процедурой удаления и создания альтернативной ноды на этом же устройстве:

Добавьте новый узел с правильным именем, используйте тот же IP-адрес, что у старого узла:

sudo linstor node create --node-type combined <новое_имя_узла> <тот_же_ip_адрес_узла>

Удалите старый узел используя команду

linstor node delete <старое_имя_узла>

которая удаляет метаданные, связанные с данным узлом. Данная операция исключает узел из управления ресурсами DRBD в LINSTOR, однако при этом не затрагивает физические данные — файлы, диски и LVM-тома сохраняются на носителях. Изменения фиксируются в базе данных LINSTOR (при наличии контроллера), но при этом запущенные DRBD-ресурсы на удаляемом узле не останавливаются автоматически и требуют ручного вмешательства. Кроме того, удаление узла не оказывает непосредственного влияния на остальные узлы кластера, однако в случае, если узел участвовал в репликации, необходимо выполнить её перестройку. Рекомендуется применять данную процедуру только в ситуациях, когда узел более не должен входить в состав кластера, перед переустановкой системы или выводом сервера из эксплуатации, а также если узел был добавлен ошибочно.

<ip_адрес_узла> – IP-адрес, используемый для связи между нодами (используется локальный адрес во внутренней сети). При выполнении команды вводить нужно непосредственно ip-адрес, а не доменное имя.

Для создания других узлов используются аналогичные команды.

3. Настройка параметров DRBD в контроллере

DRBD-reactor – легковесная служба ватоматизации для управления ресурсами DRBD в реальном времени.

DRBD – технология ядра Linux для синхронной или асинхронной репликации блоков данных между серверами в реальном времени. DRBD функционирует как модуль ядра операционной системы, работая на самом низком уровне в качестве драйвера виртуальных блочных устройств.

Для настройки параметров DRBD нужно выполнить 2 дополнительные команды.

Параметры сетевых буферов и управления памятью задаются при помощи команды:

```
sudo linstor controller drbd-options \
  --max-buffers=36k \
  --rcvbuf-size=2048k \
  --sndbuf-size=1024k
```

--max-buffers=36k – задает максимальное количество буферов памяти, выделяемых DRBD для обработки операций ввода-вывода.

--rcvbuf-size=2048k – устанавливает размер буфера приема (receive buffer) для сетевых сокетов.

--sndbuf-size=1024k – размер буфера отправки (send buffer).

Для проверки введенных настроек используйте команду:

```
sudo linstor controller list-properties | grep "DrbdOptions/Net
```

Настройка контроля скорости синхронизации осуществляется с помощью данной команды:

```
sudo linstor controller drbd-options \
    --c-fill-target=24M \
    --c-max-rate=720M \
    --c-min-rate=80M \
    --c-plan-ahead=15
```

Команда настройки контроля скорости используется в случае:

- добавления нового диска (инициализация репликации);
- восстановления узла после сбоя;
- когда синхронизация идёт слишком медленно (можно увеличить c-max-rate);
- когда синхронизация перегружает сеть (уменьшите c-max-rate).

При использовании нескольких ресурсов DRBD, объединённых в общую сеть репликации и синхронизации, фиксированная скорость передачи данных может оказаться неэффективной и привести к неоптимальному распределению пропускной способности. В связи с этим, по умолчанию активирован режим переменной скорости синхронизации, в котором DRBD применяет адаптивный алгоритм управления потоком данных. Этот алгоритм динамически определяет и регулирует скорость синхронизации, что позволяет минимизировать влияние фоновой репликации на основную операционную нагрузку ввода-вывода (I/O).

Для оптимальной настройки параметра c-fill-target, отвечающего за целевой размер буфера синхронизации, рекомендуется исходить из расчёта, основанного на величине BDP (Bandwidth-Delay Product), которая представляет собой произведение пропускной способности канала передачи данных на задержку сигнала:

Оптимальное значение c-fill-target=2×BDP

Например, для локальной сети с пропускной способностью 10 Гбит/с и задержкой порядка 50 микросекунд (50 × 10^{-6} с) BDP рассчитывается следующим образом:

BDP=1200МБ/с×0,00005с=0,06МБ

С учётом практических требований и запасов, рекомендуемое значение параметра c-fill-target составляет от 1 до 2 МБ, при этом оптимальным считается установка на уровне 1 МБ, округляя значение вверх для обеспечения стабильности работы.

--c-fill-target=24M – задает объем данных, который DRBD пытается держать в буфере перед отправкой.
Важно. Слишком большое значение этого параметра при недостаточном объеме оперативной памяти может ухудшить работу системы. Это происходит из-за увеличения нагрузки на память и возможного вытеснения других критически важных процессов системы.

--c-max-rate=720M – максимальная скорость синхронизации данных между узлами.
Важно. Максимальная скорость синхронизации данных между узлами не должна превышать 80% доступной полосы сети (пример: Для 10 GbE (~1.2 GB/s) максимум — 960M (960 МБ/с)).

--c-min-rate=80 – минимальная гарантированная скорость синхронизации (предотвращает полную остановку синхронизации при высокой нагрузке).

--c-plan-ahead=15 – время (в секундах), на которое DRBD планирует передачу данных (оптимально: 10-20 для сетей с переменной задержкой и меньшие значения для стабильных сетей).

Возможные сценарии использования команды:

Быстрая синхронизация (например, в случае высокой пропускной способности сети – 10 Gbps и большого объема оперативной памяти):

```
sudo linstor controller drbd-options \
  --c-fill-target=64M \
  --c-max-rate=1200M \
  --c-min-rate=100M \
  --c-plan-ahead=20
```

Для ограничения нагрузки (например, низкая пропускная способность сети – 1 Gbps):

```
sudo linstor controller drbd-options \
  --c-fill-target=16M \
  --c-max-rate=100M \
```

```
--c-min-rate=10M \
--c-plan-ahead=10
```

Если используются накопители с относительно низкой скоростью чтения и записи данных (HDD), увеличение скорости не даст ожидаемого эффекта.

Для проверки внесенных изменений используйте команду:

```
sudo linstor controller list-properties | grep -E "c-fill-target|c-max-rate|c-min-rate|c-plan-ahead"
```

Далее в документе описан процесс создания новых ресурсов с использованием программного обеспечения.

После окончания настройки проверьте список нод командой `sudo linstor node list`.

Ожидаемый вывод:

```
┊ Node          ┊ NodeType   ┊ Addresses                  ┊ State  ┊
╞═══════════════════════════════════════════════════════════════════╡
┊ <имя_узла-1>  ┊ COMBINED   ┊ <ip_адрес:порт> (PLAIN)    ┊ Online ┊
┊ <имя_узла-2>  ┊ CONTROLLER ┊ <ip_адрес:порт> (PLAIN)    ┊ Online ┊
┊ <имя_узла-3>  ┊ SATELLITE  ┊ <ip_адрес:порт> (PLAIN)    ┊ Online ┊
```

4. Разметка дисков для томов типа LVM

Разметка дисков производится на нодах типа сателлит.

Подготовьте место на диске для дальнейшей работы с помощью утилиты для работы с разделами fdisk:

```
sudo fdisk /dev/<имя_диска>
```

Важно: Все данные на диске будут удалены. Убедитесь, что выбрали правильный диск.

Внутри fdisk введите команды по порядку:

n → создать новый раздел.

p → выбрать тип раздела: основной (primary).

Enter → принять номер раздела по умолчанию.

Enter → принять первый сектор по умолчанию (2048).

Enter → принять последний сектор по умолчанию (весь диск).

w → сохранить изменения и выйти.

Для инициализации раздела под LVM создайте физический том (PV) используя команду:

```
sudo pvcreate /dev/<имя_раздела>
```

pvcreate – команда для инициализации диска/раздела под LVM.

Имя тома (раздела) обычно присваивается по принципу <disk_name> + p + <part_number>, например, nvme0n1p1.

Создайте группу томов (VG)

```
sudo vgcreate <имя_группы_ресурсов> /dev/<имя_раздела>
```

Для проверки созданных объектов используйте команды:

```
sudo pvdisplay # Список физических томов
```

```
sudo pvdisplay | grep -B 1 <имя_раздела>  # Фильтрует вывод, показывая строки с интересующим именем и дополнительную строку перед ними для контекстного понимания информации.
```

```
sudo vgdisplay | grep -B 1 <имя_группы_ресурсов> # Аналогичный вывод с фильтрацией по исени тома.
```

5. Разметка дисков для томов типа LVM_THIN

Разметка дисков производится на нодах типа сателлит.

Подготовьте место на диске для дальнейшей работы с помощью утилиты для работы с разделами fdisk. Если на диске не требуется создавать разделы, использовать fdisk не обязательно:

```
sudo fdisk /dev/<имя_диска>
```

Важно: Все данные на диске будут удалены. Убедитесь, что выбрали правильный диск.
Внутри fdisk введите команды по порядку:

n → создать новый раздел.

p → выбрать тип раздела: основной (primary).

Enter → принять номер раздела по умолчанию.

Enter → принять первый сектор по умолчанию (2048).

Enter → принять последний сектор по умолчанию (весь диск).

w → сохранить изменения и выйти.

Для создания LVM Thin Pool необходимо выполнить следующую последовательность команд:

Создать физический том (PV):

```
sudo pvcreate /dev/<имя_раздела> # помечаем раздел как "физический том" для LVM
```

pvcreate – команда для инициализации диска/раздела под LVM.

/dev/<имя_раздела> – созданный раздел (после fdisk).

Создать группу томов (VG):

```
sudo vgcreate <имя_группы_ресурсов> /dev/<имя_раздела>
```

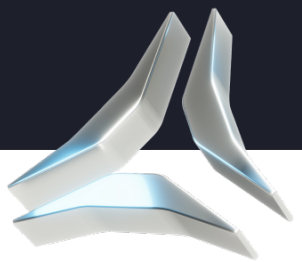
vgcreate – создает группу томов (логический контейнер для дисков).

Имя группы (можно выбрать любое, например, data_vg).

Создать «тонкий пул» в группе (Thin Pool):

```
sudo lvcreate -l 100%FREE -T <имя_группы_ресурсов>/<имя_тома_lvm_thin>
```

lvcreate – создает логический том.



-l 100%FREE – использовать всё свободное место в группе.

-T – указание создать Thin Pool (пул с динамическим выделением места).

<имя_группы_ресурсов>/<имя_тома_lvm_thin> – том располагается внутри группы.

Для проверки созданных сущностей воспользуйтесь командой:

```
sudo lvdisplay | grep -E "LV Name|VG Name
```

В случае возникновения необходимости в удалении созданных ресурсов, можно воспользоваться следующими командами:

- удаление группы томов: `sudo vgremove <имя_группы_ресурсов>`
- удаление физического тома: `sudo pvremove /dev/<имя_раздела>`

7. Создание пула хранения

Введите команду создания пула хранения с использованием технологии LVM Thin Pool (место выделяется по мере записи данных):

```
sudo linstor storage-pool create lvmthin <имя_узла> <имя_пула_хранения>
<имя_группы_ресурсов>/<имя_тома_lvm_thin>
```

<имя_группы_ресурсов>/<имя_тома_lvm_thin> – путь к логическому тому LVM, который вы создали ранее.

На остальных устройствах выполните аналогичные команды.

Для добавления обычного пула (не Thin) в программном комплексе управления распределёнными хранилищами используйте команду:

```
sudo linstor storage-pool create lvm <имя_узла> <имя_группы_ресурсов>
```

Для вызова списка созданных пулов воспользуйтесь командой:

```
sudo linstor storage-pool list
```

DfltDisklessStorPool - это специальный предопределённый пул хранения в TROK, автоматически создаваемый при добавлении узла для работы с diskless-ресурсами (ресурсами без локального диска).

Пример вывода с данными о данном пуле:

StoragePool	Node	Driver	PoolName	FreeCapacity	TotalCapacity	CanSnapshots	State	SharedName
DfltDisklessStorPool	astra-loc-1	DISKLESS			False	Ok	astra-loc-1;DfltDisklessStorPool	
DfltDisklessStorPool	astra-loc-2	DISKLESS			False	Ok	astra-loc-2;DfltDisklessStorPool	
DfltDisklessStorPool	astra-loc-3	DISKLESS			False	Ok	astra-loc-3;DfltDisklessStorPool	
on_db	astra-loc-1	LVM_THIN	linstorpool/ldb	9.97 GiB	9.97 GiB	True	Ok	astra-loc-1;on_db
on_db	astra-loc-2	LVM_THIN	linstorpool/ldb	9.97 GiB	9.97 GiB	True	Ok	astra-loc-2;on_db
on_db	astra-loc-3	LVM_THIN	linstorpool/ldb	9.97 GiB	9.97 GiB	True	Ok	astra-loc-3;on_db

8. Создание группы ресурсов (Resource Group)

Для создания группы ресурсов введите команду:

```
sudo linstor resource-group create <имя_группы_ресурсов> --storage-pool <имя_пула_хранения> --place-count 2
```

--storage-pool <имя_пула_хранения> – указание места хранения данных.

--place-count 2 – количество копий ресурса.

Для просмотра созданных групп ресурсов поспользуйтесь командой:

```
sudo linstor resource-group list
```

DfltRscGrp — это предопределённая группа ресурсов в TROK, которая автоматически применяется к новым ресурсам, если не указана другая группа. Это шаблон конфигурации с базовыми настройками для новых ресурсов.

9. Настройка характеристик ресурса

Создайте новое определение ресурса с произвольным именем (например, res1):

```
sudo linstor resource-definition create <имя_определения_ресурса>
```

Затем введите команду настройки параметров DRBD, которая устанавливает параметр для определения ресурса <resource_definition_name>. Параметр определяет поведение в случае отсутствия кворума.

```
sudo linstor resource-definition drbd-options --on-no-quorum=io-error <имя_определения_ресурса>
```

--on-no-quorum=io-error – если кворум потерян (например, 2 из 3 узлов отключены), DRBD блокирует запись и выводит ошибку.

Отключите автоматическое продвижение ресурса:

```
sudo linstor resource-definition drbd-options --auto-promote=no <имя_определения_ресурса>
```

--auto-promote=no – в случае сбоя одного из узлов, ресурс не будет автоматически переведен в состояние активного на другом узле. Такой режим (--auto-promote=no) обязательно требуется при переводе DRBD-ресурсов в активное состояние с помощью службы drbd-reactor.

10. Создание нового определения тома

Для создания шаблона логического тома, которое задаёт параметры и характеристики будущих томов, введите команду:

```
sudo linstor volume-definition create <имя_определения_тома> 100G
```

Для проверки созданного определения тома можно использовать команду

```
sudo linstor volume-definition list
```

Пример вывода:

```
┊ ResourceName          ┊ VolumeNr ┊ VolumeMinor ┊ Size   ┊ Gross ┊ State ┊
╞═══════════════════════════════════════════════════════════════════════════╡
┊ <имя_определения_тома> ┊ 0        ┊             ┊ 1 GiB  ┊       ┊ ok    ┊
```

11. Создание рабочего экземпляра ресурса

Для создания рабочего экземпляра ресурса с произвольным именем выполните следующую команду:

```
sudo linstor resource create <resource_definition_name> --auto-place 3 -s <st_pool_name>
```

--auto-place 3 указывает, что ресурс будет автоматически распределен на трех узлах, но с учетом ограничения --place-count, которое было установлено в пункте 5 Создание группы ресурсов.

- в случаях, когда в Resource Group указано --place-count 2, ресурс будет реплицирован только на 2 нодах из 3.
- оставшаяся нода остается в режиме арбитра, т.е. принимает участие в кворуме при изменении данных на DRBD-ресурсе и обеспечивает diskless-доступ к ресурсу.

-s <st_pool_name> указывает, что для размещения ресурса используется пул с заданным именем

Программа выбирает 2 ноды из 3 (согласно --place-count).

На выбранных нодах:

- создаются дисковые тома (100 ГБ, как в volume-definition).
- запускается DRBD для синхронизации данных между ними.
- том становится доступным для использования (например, для монтирования в систему).

В случае, если необходимо изменить количество репликаций, можно сделать это двумя способами.

Первая команда обновляет настройки репликации группы ресурсов:

```
sudo linstor resource-group set-property <имя_группы_ресурсов> --place-count 3
```

Второй набор команд позволяет пересоздать ресурс:

```
sudo linstor resource delete <имя_определения_ресурсов>
sudo linstor resource create <имя_определения_ресурсов> --auto-place 3
```

Для проверки размещения используется команда:

```
sudo linstor resource list
```

Пример вывода:

ResourceName	Node	Port	Usage	Conns	State	CreatedOn
resource_definition_1	astra17-qa-doc-test	7000	Unused	Ok	UpToDate	2025-06-30 14:03:46

12. Настройка Gateway

Для того, чтобы служба drbd-reactor выполняла автоматическую перезагрузку при обнаружении изменений в файлах конфигурации, выполните команды:

```
sudo cp /usr/share/doc/drbd-reactor/examples/drbd-reactor-reload.{path,service} /etc/systemd/system
```

```
sudo systemctl enable --now drbd-reactor-reload.path
```

Настройте конфигурацию на узлах-сателлитах для возможности записи файлов конфигурации drbd-reactor:

```
sudo tee /etc/linstor/linstor_satellite.toml <<EOF
[files]
allowExtFiles = [
   "/etc/systemd/system",
   "/etc/systemd/system/linstor-satellite.service.d",
   "/etc/drbd-reactor.d"
]
EOF
```

```
sudo systemctl restart linstor-satellite.service
```

Отключите авто-запуск сервиса NFS:

```
systemctl disable --now nfs-server.service
```

Команда `sudo linstor-gateway check-health --iscsi-backends lio` используется для проверки состояния Gateway.

Ожидается, что команда выведет информацию о состоянии различных компонентов системы:

[✓] System Utilities – базовые системные утилиты
[✓] LINSTOR – сервис управления хранилищем
[✓] drbd-reactor – служба мониторинга и автоматического восстановления
[✓] Resource Agents – скрипты для управления ресурсами
[✓] iSCSI – поддержка протокола iSCSI (Linux IO Target, lio)
[✓] NVMe-oF – поддержка NVMe over Fabrics
[✓] NFS – поддержка сетевой файловой системы

Если все галочки зеленые ([✓]), система готова к настройке Gateway.

Для создания iSCSI Target (точки доступа к хранилищу) для ресурса res1 используйте команду:

```
sudo linstor-gateway iscsi create res1 --target iqn.2024-04.res1 --portal 185.227.37.214:3260
```

res1 – это имя для новой iSCSI-цели.

--target iqn.2024-04.res1 – уникальный идентификатор Target (формат iqn.год-месяц.название).

--portal 185.227.37.214:3260 – IP-адрес и порт (3260 — стандартный для iSCSI), через который смогут подключаться клиенты.

В результате выполнения команды приложение создаст iSCSI Target на указанном IP, ресурс res1 станет доступен как iSCSI LUN, клиенты (другие серверы) смогут подключаться к Target и использовать том как блочное устройство.

Для проверки создания используется команда:

```
sudo linstor-gateway iscsi list
```

Пример вывода:

```
+-------------------------------------------------+
| Target IQN          | Portal              | State |
| iqn.2024-04.res1    | 185.227.37.214:3260 | OK    |
+-------------------------------------------------+
```

Для подключения к iSCSI Target с клиента на Linux-клиенте выполните команды:

```
sudo iscsiadm --mode discovery --type st --portal 185.227.37.214
```

```
sudo iscsiadm --mode node --targetname iqn.2024-04.res1 --portal 185.227.37.214:3260 --login
```

В результате том появится как блочное устройство (например, /dev/sdb).

В случае возникновения ошибок:

- Проверьте, что ресурс res1 существует и активен:
  `sudo linstor resource list`
- Убедитесь, что порт 3260 открыт в фаерволе:
  `sudo ufw allow 3260/tcp`
- Перезапустите службы:
  `sudo systemctl restart linstor-gateway`

13. Настройка WEB UI

Параметры для входа:

username: "admin"

password: "1234567890"

14. Проверка кластера

Выполните команду sudo linstor resource list, чтобы увидеть список всех ресурсов, доступных в кластере.

Выполните команду sudo linstor storage-pool list, чтобы получить информацию о всех пулах хранения в кластере.

Выполните команду sudo linstor volume list, чтобы увидеть список всех томов, которые созданы в кластере.

Итог

Кластер из 3 нод настроен с репликацией данных между 2 нодами.

Доступны:

- CLI-управление через linstor.
- Веб-интерфейс на порту 80.
- Шлюз для iSCSI/NVMe-oF/NFS.

Ресурсы автоматически перераспределяются при сбоях.