Лабораторная работа №4.

Работа с systemd

Выполнила студентка группы М3311

Авсюкевич Анастасия

**Цель работы:** получить практические навыки по управлению ОС Linux с помощью systemd

**Задание:**

***Часть 1. Получение информацию о времени загрузки***.

1. Выведите информацию о времени, затраченном на загрузку системы

systemd-analyze



Эта команда выводит информацию о времени, затраченном на загрузку ядра, время для загрузки init-системы и время загрузки всех сервисов.

2. Выведите список всех запущенных при страте системы сервисов, в

порядке уменьшения времени, затраченного на загрузку сервиса.

systemd-analyze blame (--no-pager)

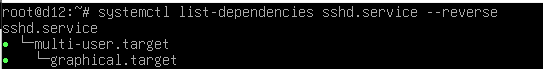




3. Выведите список сервисов, запуск которых с необходимостью

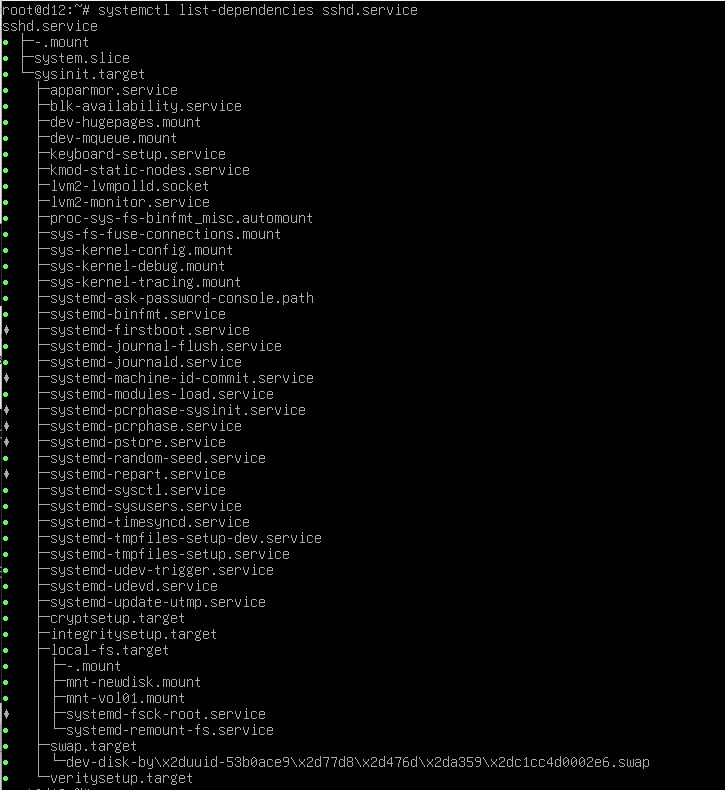
предшествовал запуску сервиса sshd.

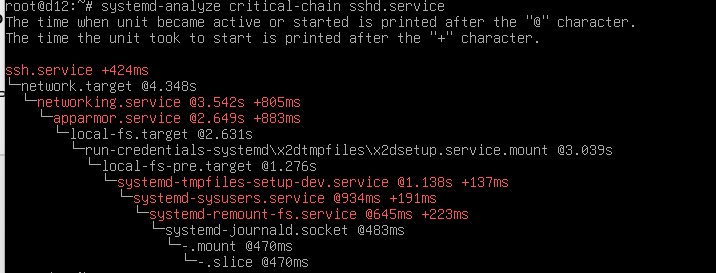
systemctl list-dependencies sshd.service –reverse



Эта команда показывает, какие сервисы или юниты **зависят от** sshd.service, то есть какие сервисы будут ожидать его запуска или успешного завершения перед своим собственным запуском.

И наоборот, то, отчего зависит sshd.service:





Эта команда отображает цепочку зависимостей, которая необходима для запуска сервиса. Она показывает, какие сервисы должны быть запущены до того, как начнется запуск sshd.service.

Только сервисы:

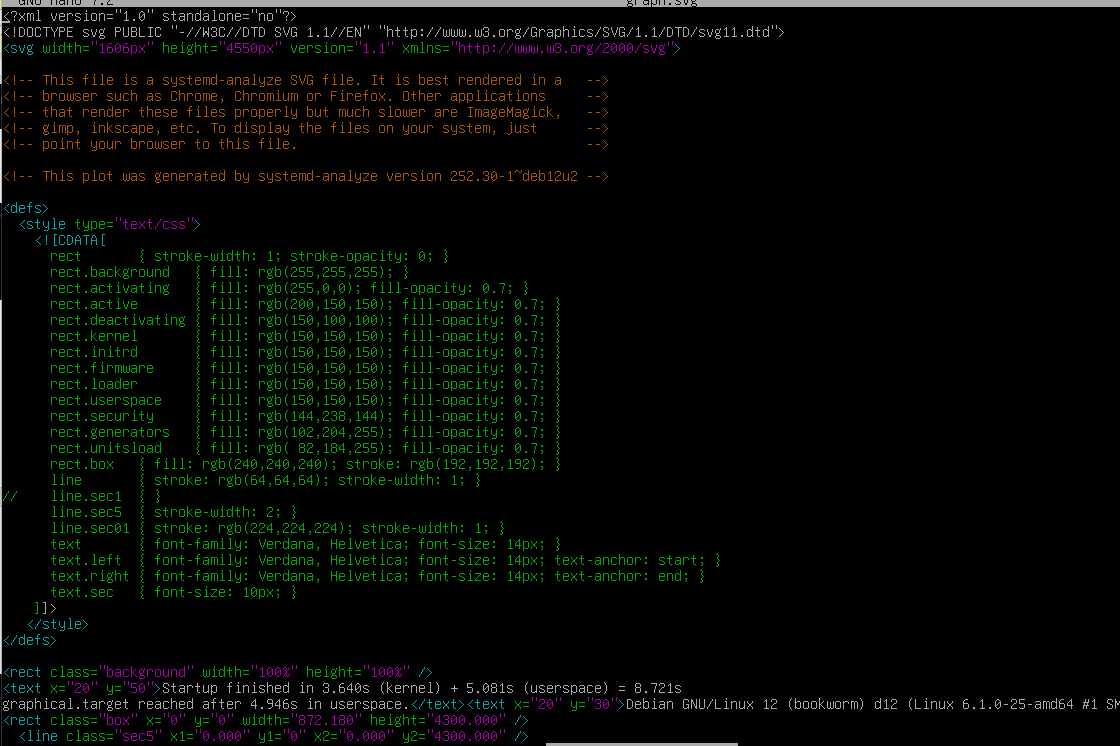


4. Сформируйте изображение в формате svg с графиком загрузки системы,

сохраните его в файл.



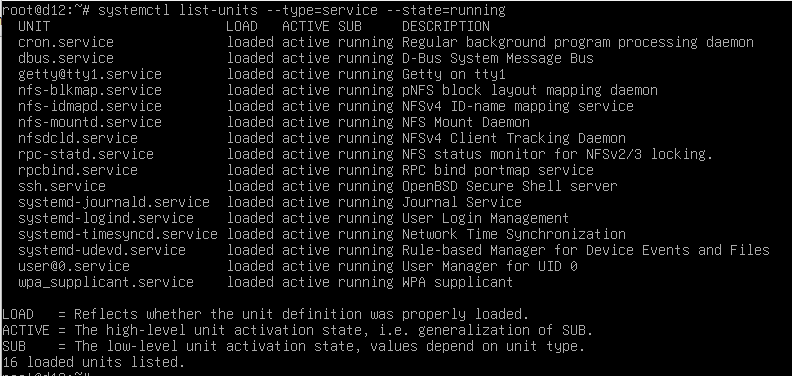
Открытый с помощью nano файл:



***Часть 2. Управление юнитами.***

1. Получите список всех запущенных юнитов сервисов

systemctl list-units --type=service --state=running

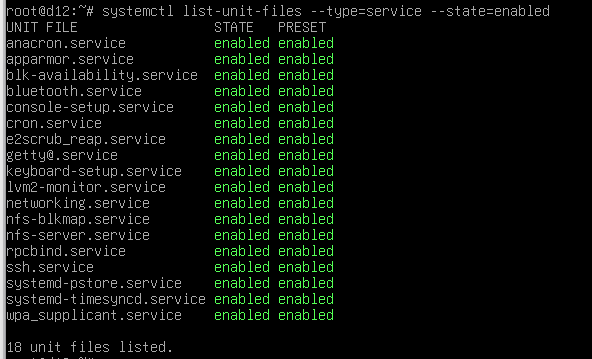


Команда systemctl list-units используется для вывода **активных юнитов**

2. Выведите перечень всех юнитов сервисов, для которых назначена

автозагрузка.

systemctl list-unit-files --type=service --state=enabled



Команда systemctl list-unit-files используется для вывода списка **конфигурационных файлов** юнитов, которые определяют, как и когда сервисы должны быть запущены или остановлены. Эта команда не показывает текущие состояния юнитов, а отображает их **файлы конфигурации** и **статус автозагрузки.**

3.Определите от каких юнитов зависит сервис sshd.

systemctl show -p Requires,RequiresOverridable sshd.service



Requires: Это свойство показывает, какие другие юниты должны быть активированы, чтобы запустился данный сервис. Иными словами, это список обязательных зависимостей для запуска юнита. Если любой из этих юнитов не запущен, то и основной сервис не будет запущен.

RequiresOverridable: Это свойство показывает, какие юниты являются опциональными зависимостями для данного юнита. Если они не могут быть запущены, это не приведет к сбою самого юнита. Однако если эти юниты запущены, они могут влиять на запуск основного сервиса.

4.Определите запущен ли сервис cron, если нет, запустите его.

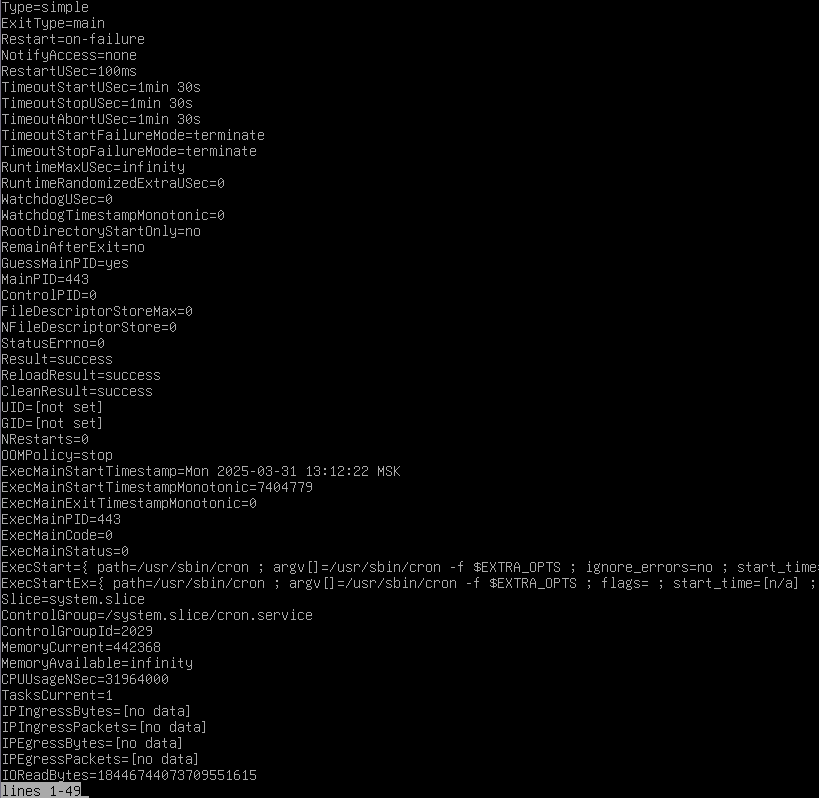
systemctl is-active cron



5. Выведите все параметры юнита cron, даже те, которые были назначены

автоматически, и не были прописаны в файле юнита.

systemctl show cron.service



6. Запретите автозагрузку сервиса cron, но оставите ему возможность

запускаться по зависимостям.

systemctl disable cron.service



Команда disable только отключает автозагрузку, но сервис все равно может быть запущен в случае, если он нужен другим сервисам в рамках их зависимостей.

***Часть 3. Создание сервиса***

1. Создайте собственный сервис mymsg. Сервис mymsg должен:

a. при старте системы записывать в системный журнал дату и время

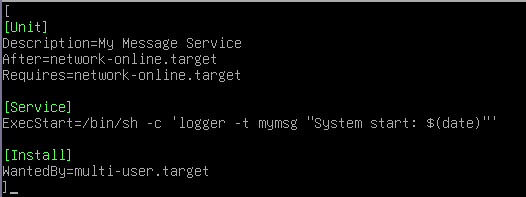
b. должен запускаться только если запущен сервис network.

Примечание: писать в системный журнал позволяет команда logger.

Проверить корректность юнит-файла service позволяет команда

Systemd-analyze.

nano /etc/systemd/system/mymsg.service



Различия между network.target и network-online.target:

1. **network.target**: Это целевой юнит, который сигнализирует о том, что ядро Linux активировало сетевые интерфейсы, но это не означает, что сеть готова к использованию. Некоторые интерфейсы могут быть еще не настроены, а протоколы могут быть еще не инициализированы.
2. **network-online.target**: Это более строгий целевой юнит, который сигнализирует о том, что сеть полностью готова к использованию, и все необходимые сетевые сервисы и интерфейсы находятся в рабочем состоянии. Например, если сеть настроена через DHCP, network-online.target будет активирован только после получения IP-адреса.

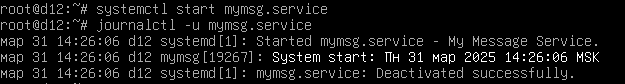


Ошибок нет

2.Настройте автоматический запуск сервиса mymsg при старте системы.



3. Запустите сервис

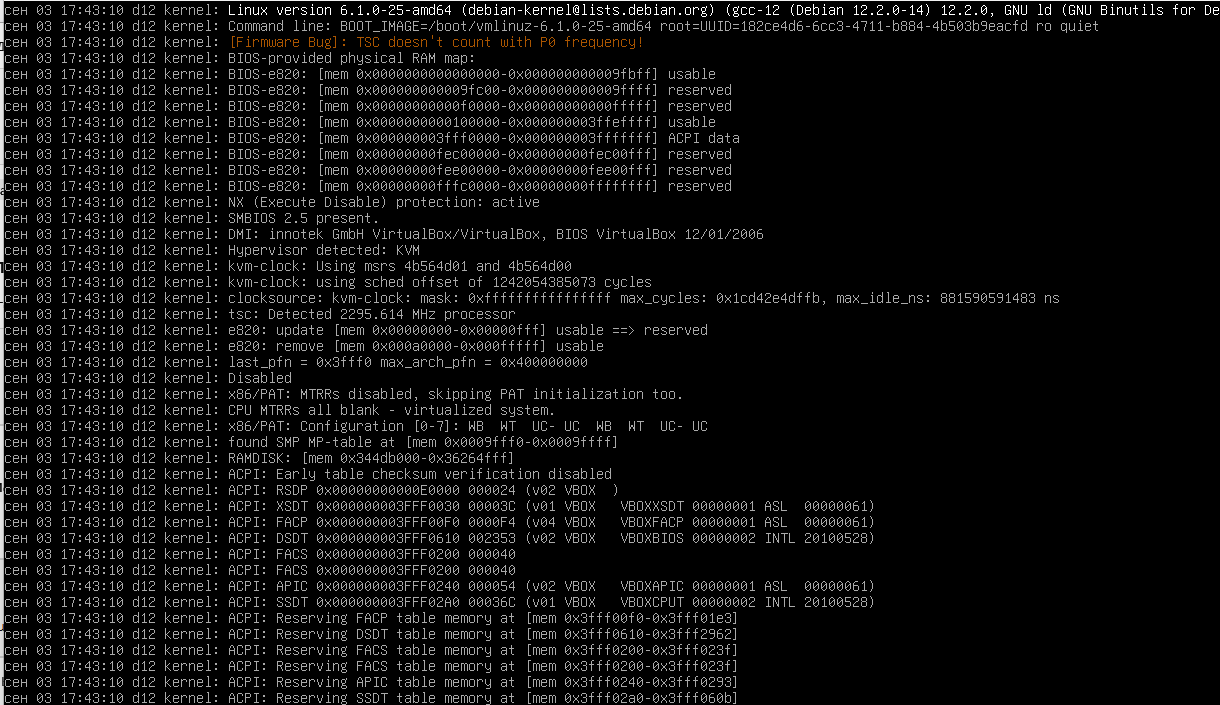


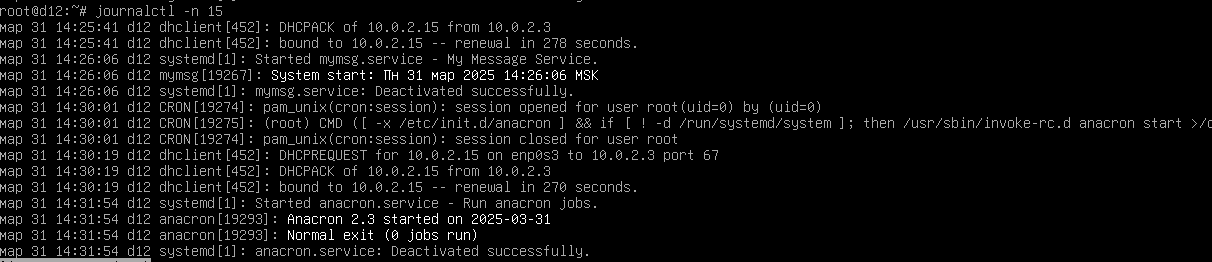
***Часть 4. Работа с системным журналом***

1. Выведите на консоль системный журнал. Убедитесь, что сервис mymsg

отработал корректно.

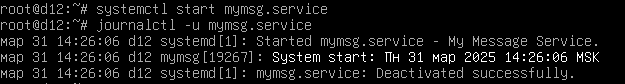
journalctl





2. Выведите на консоль все сообщения системного журнала, касающиеся

сервиса mymsg.

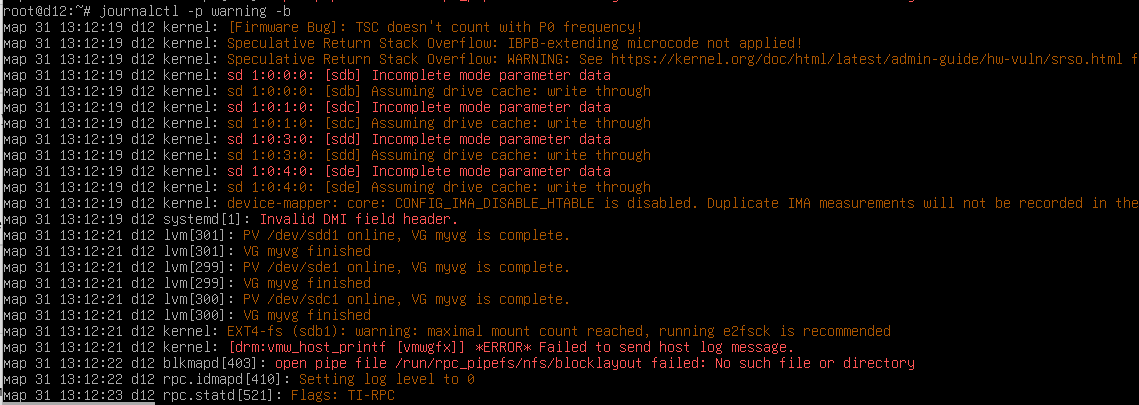


3.Выведите на экран все сообщения об ошибках в журнале.



-b: Этот флаг фильтрует сообщения только **с последнего загрузочного сеанса** системы (с момента последней перезагрузки). Это означает, что будут показаны только записи журнала, относящиеся к текущему сеансу работы системы.

Для ошибок более «высокого» уровня:



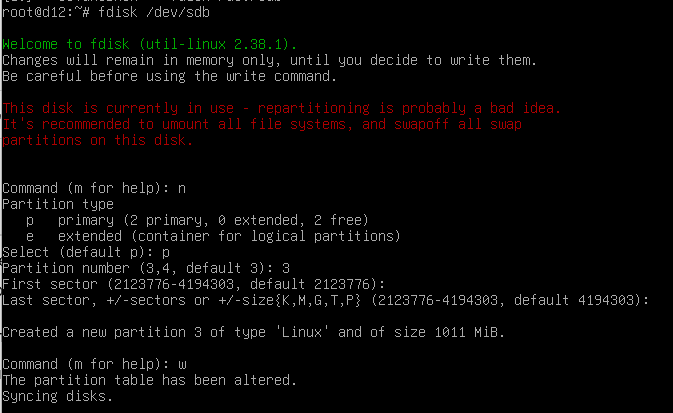
4.Определите размер журнала.



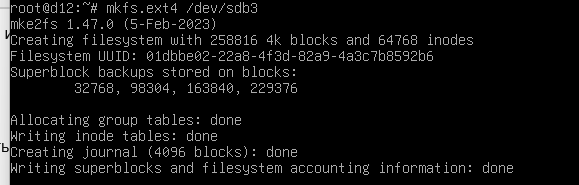
***Часть 5. Создание и настройка .mount юнита***

1. Подготовьте файловую систему.

a. Создайте новый раздел на диске или используйте существующий.



b. отформатируйте его в файловую систему ext4.



c. Создайте директорию для монтирования /mnt/mydata



2. Создание .mount юнита

a. Создайте файл .mount юнита в /etc/systemd/system/mnt-

mydata.mount.

b. Настройте юнит следующим образом:

i. Добавьте описание юнита в секцию [Unit].

ii. В секции [Mount] укажите устройство, точку

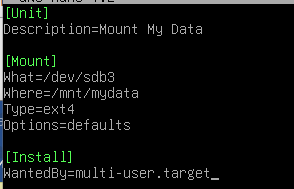
монтирования, тип файловой системы и опции.

iii. В секции [Install] укажите, что юнит должен быть

активирован при достижении multi-user.target.

c. Сохраните файл и выйдите из редактора.

nano /etc/systemd/system/mnt-mydata.mount

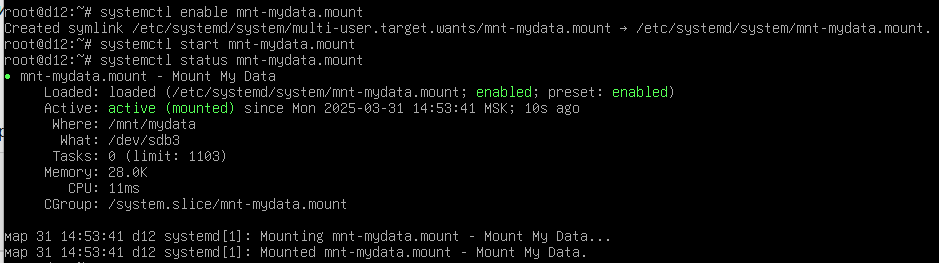


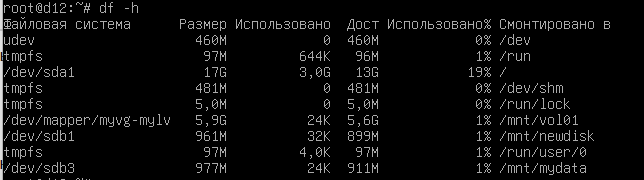
3. Запуск и проверка .mount юнита

a. Включите и запустите юнит.

b. Проверьте статус юнита.

c. Убедитесь, что раздел смонтирован.







***Часть 6. Использование .automount для отложенного монтирования***

1. Подготовьте соответствующий .mount-юнит

a. После выполнения Части 5 у вас должен был остаться юнит для

монтирования /mnt/mydata

b. Убедитесь, что при остановке раздел отмонтируется, а

монтируется обратно только при запуске юнита или перезагрузке

системы



Корректно отмонтировался

2. Создание .automount юнита

a. Создайте файл .automount юнита в /etc/systemd/system/mnt-

mydata.automount.

b. Настройте юнит:

i. В секции [Unit] добавьте описание.

ii. В секции [Automount] укажите точку монтирования и

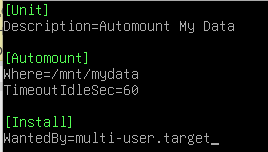
время до размонтирования (TimeoutIdleSec).

iii. В секции [Install] укажите, что юнит должен быть

активирован при достижении multi-user.target.

c. Сохраните файл и выйдите из редактора.

nano /etc/systemd/system/mnt-mydata.automount



3. Запуск и проверка .automount юнита

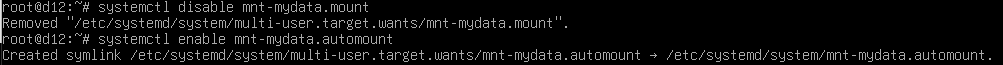
a. Включите и запустите .automount юнит.

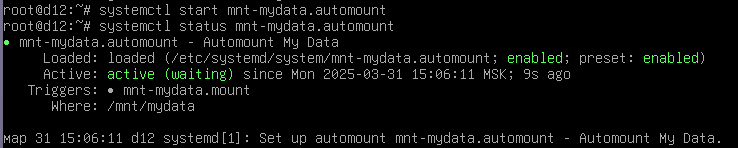
b. Проверьте статус юнита и убедитесь, что раздел монтируется при

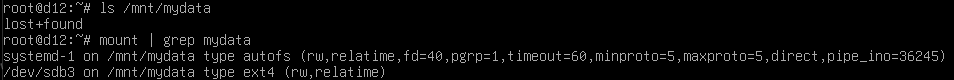
обращении к точке монтирования.

c. Убедитесь, что раздел размонтируется после завершения

работы.







….



***Вопросы и задания:***

1. Чем отличаются команды systemctl restart и systemctl try-restart?

systemctl restart: Эта команда всегда перезапускает сервис, независимо от его текущего состояния.

systemctl try-restart: Эта команда перезапускает сервис только в том случае, если он уже запущен.

2. Как с помощью systemctl запустить Linux в однопользовательском

режиме?

sudo systemctl isolate rescue.target или sudo systemctl isolate rescue

Для того чтобы запустить систему в **однопользовательском режиме** (single-user mode), можно использовать параметр для целевого состояния (target). Однопользовательский режим — это минимальное окружение, в котором загружается только базовая часть системы, и только один пользователь может работать (обычно это root).

3. Пусть вам нужно создать еще один сервис mysrv, который не будет

запускаться автоматически, и может быть выполнен, только если

сервис mymsg будет принудительно остановлен уже после старта

системы. Приведите параметры и их значения из описания юнитов

mymsg и mysrv, которые обеспечат выполнения этих условий.

nano /etc/systemd/system/mysrv.service

[

[Unit]

Description=My Secondary Service

After=mymsg.service

Conflicts=mymsg.service

[Service]

ExecStart=/bin/sh -c 'logger -t mysrv "Secondary service started"'

[Install]

WantedBy=multi-user.target

]