Отчёт по лабораторной работе №5

Дисциплина: Основы администрирования операционных систем

Верниковская Екатерина Андреевна

Содержание

# 1 Цель работы

Получить навыки управления системными службами операционной системы посредством systemd.

# 2 Задание

1. Выполнить основные операции по запуску (останову), определению статуса, добавлению (удалению) в автозапуск и пр. службы Very Secure FTP.
2. Продемонстрировать навыки по разрешению конфликтов юнитов для служб firewalld и iptables.
3. Продемонстрировать навыки работы с изолированными целями.

# 3 Выполнение лабораторной работы

## 3.1 Управление сервисами

В консоли переходим в режим работы суперпользователя, используя *su -* (рис. 1)

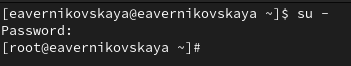


Рис. 1: Режим суперпользователя

Проверяем статус службы Very Secure FTP: *systemctl status vsftpd*. Вывод команды показывает, что сервис в настоящее время отключён, так как служба Very Secure FTP не установлена (рис. 2)

Статус службы Very Secure FTP (1)

Рис. 2: Статус службы Very Secure FTP (1)

Устанавливаем службу Very Secure FTP: *dnf -y install vsftpd* (рис. 3)

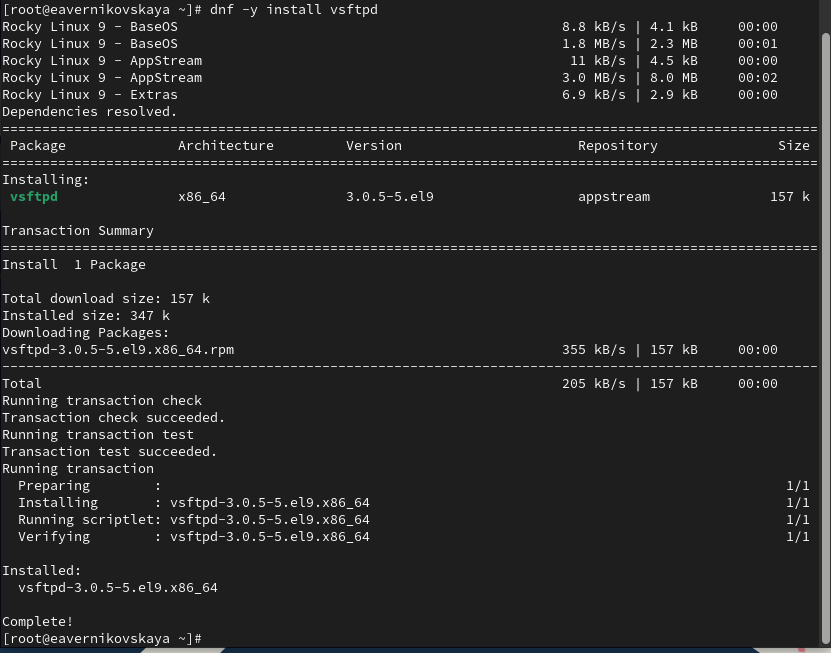


Рис. 3: Установка службы Very Secure FTP

Запускаем службу Very Secure FTP:: *systemctl start vsftpd* (рис. 4)

Запуск службы Very Secure FTP

Рис. 4: Запуск службы Very Secure FTP

Снова проверяем статус службы Very Secure FTP. Вывод команды показывает, что служба в настоящее время работает, но не будет активирована при перезапуске операционной системы (рис. 5)

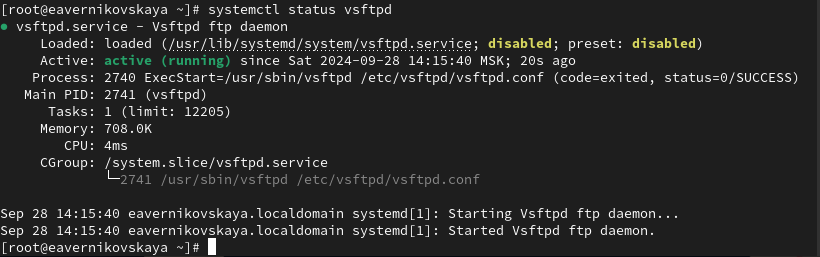


Рис. 5: Статус службы Very Secure FTP после установки

Добавляем службу Very Secure FTP в автозапуск при загрузке операционной системы, используя команду *systemctl enable vsftpd* (рис. 6)

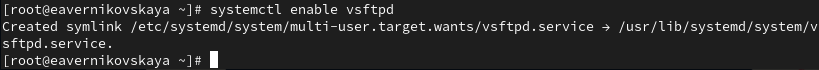


Рис. 6: Добавление службы Very Secure FTP в автозапуск (1)

Проверяем статус службы Very Secure FTP (рис. 7)

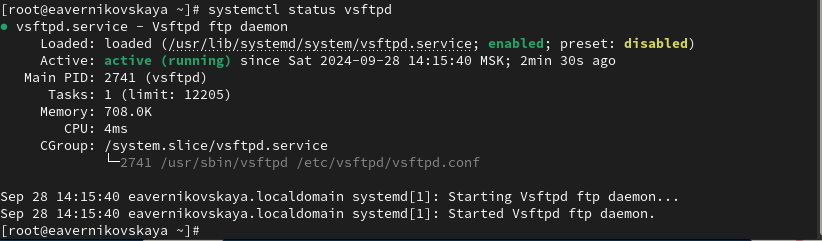


Рис. 7: Статус службы Very Secure FTP после добавления в автозапуск

Удаляем службу из автозапуска, используя команду *systemctl disable vsftpd*, и снова проверяем её статус (рис. 8), (рис. 9)

Удаление службы Very Secure FTP из автозапуска

Рис. 8: Удаление службы Very Secure FTP из автозапуска

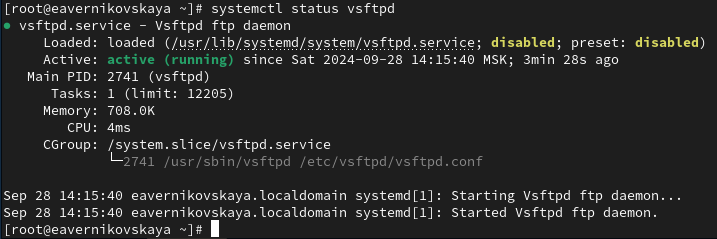


Рис. 9: Статус службы Very Secure FTP после удаления из автозапуска

Выводим на экран символические ссылки, ответственные за запуск различных сервисов: *ls /etc/systemd/system/multi-user.target.wants*. После ввода этой команды отображается, что ссылки на vsftpd.service не существует (рис. 10)

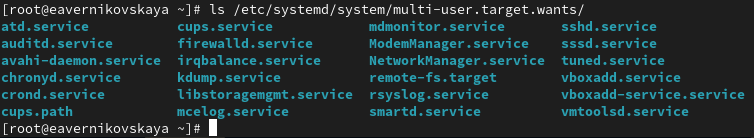


Рис. 10: Символические ссылки, ответственные за запуск различных сервисов

Снова добавляем службу Very Secure FTP в автозапуск и опять выводим на экран символические ссылки, ответственные за запуск различных сервисов. На этот раз вывод команды показывает, что создана символическая ссылка для файла /usr/lib/systemd/system/vsftpd.service в каталоге /etc/systemd/system/multi-user.target.wants (рис. 11), (рис. 12)

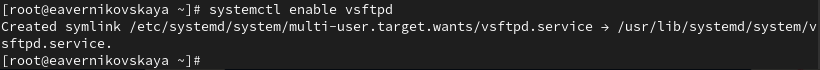


Рис. 11: Добавление службы Very Secure FTP в автозапуск (2)

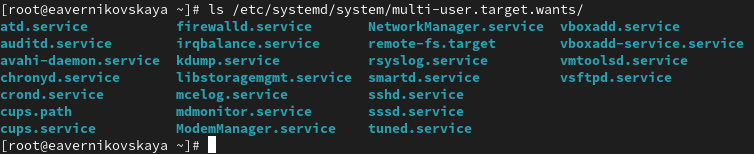


Рис. 12: Символические ссылки, ответственные за запуск различных сервисов после добавления службы в автозапуск

Опять проверяем статус службы Very Secure FTP. Теперь мы видим, что для файла юнита состояние изменено с disabled на enabled (рис. 13)

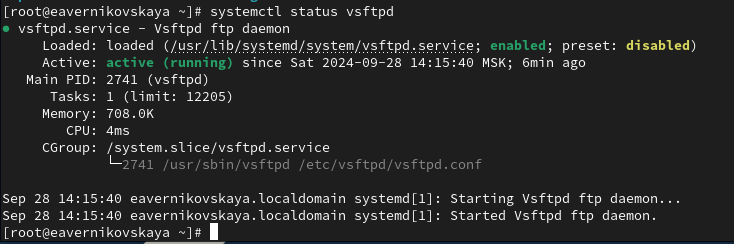


Рис. 13: Статус службы Very Secure FTP (2)

Выводим на экран список зависимостей юнита: *systemctl list-dependencies vsftpd* (рис. 14)

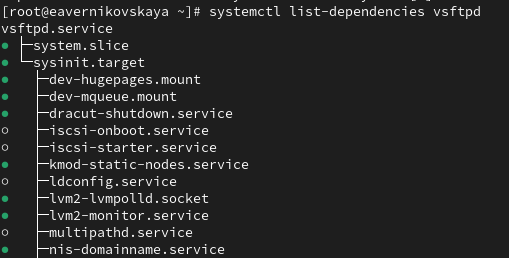


Рис. 14: Список зависимостей юнита

Выводим на экран список юнитов, которые зависят от данного юнита: *systemctl list-dependencies vsftpd –reverse* (рис. 15)

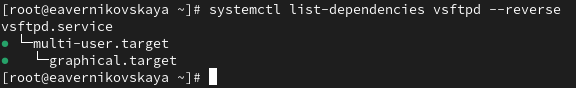


Рис. 15: Список юнитов, которые зависят от данного юнита

## 3.2 Конфликты юнитов

Устанавливаем iptables: dnf -y install iptables\* (рис. 16)

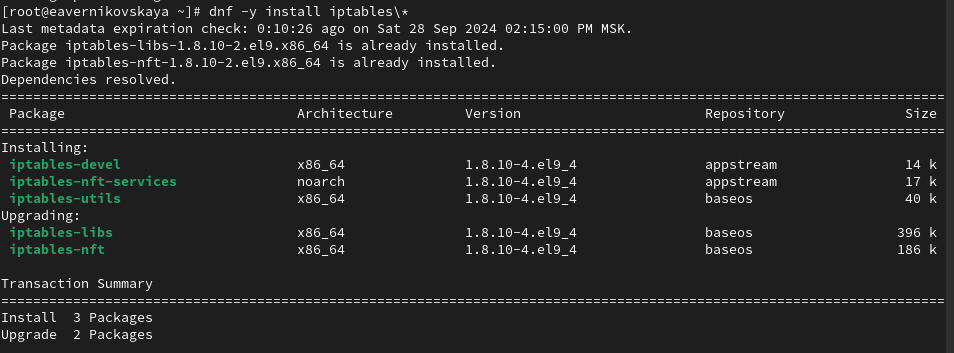


Рис. 16: Установка iptables

Далее проверяем статус firewalld и iptables (рис. 17), (рис. 18)

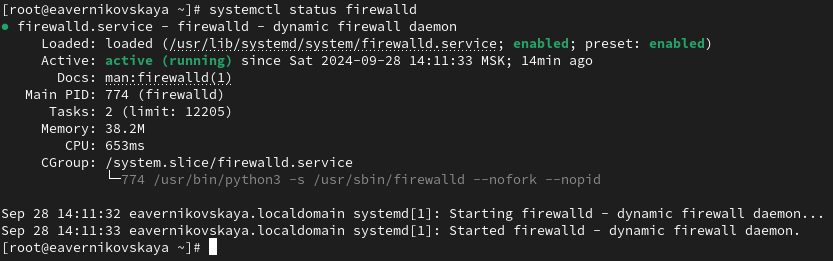


Рис. 17: Статус firewalld

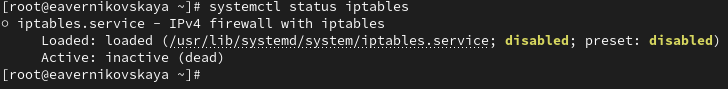


Рис. 18: Статус iptables (1)

Далее пробуем запустить firewalld и iptables. При запуске одной службы мы видим, что вторая дезактивируется или не запускается (рис. 19)

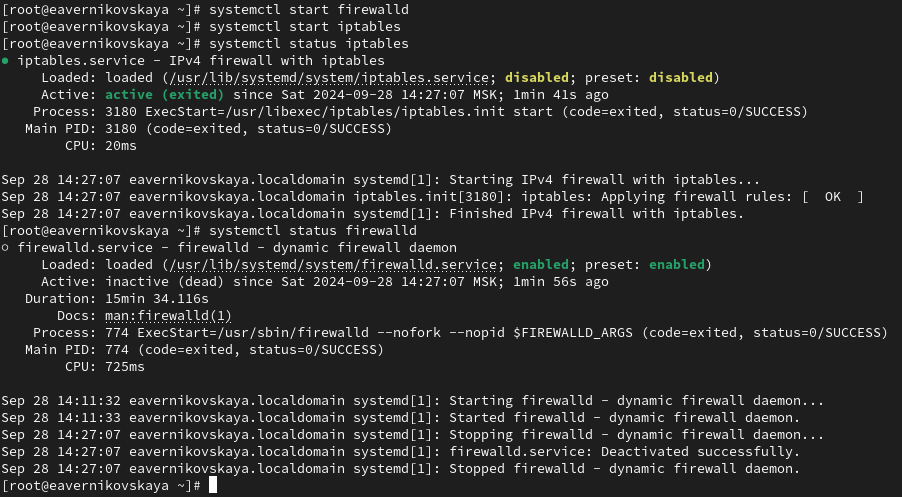


Рис. 19: Запуск firewalld и iptables и проверка их статусов

Вводим *cat /usr/lib/systemd/system/firewalld.service*. Описание накстроек конфликтов: Conflicts=iptables.service ebtables.service ipset.service nftables.service. Этот параметр задает службы, которые конфликтуют с firewalld. Это означает, что одновременно с firewalld не могут быть запущены службы iptables.service, ebtables.service, ipset.service и nftables.service. (рис. 20)

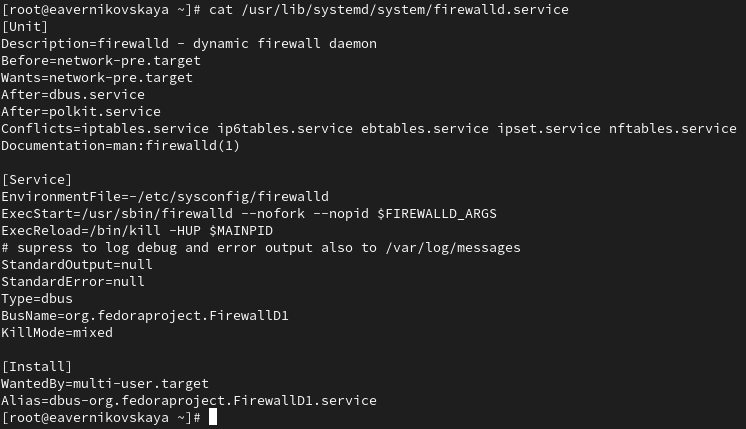


Рис. 20: Команда cat /usr/lib/systemd/system/firewalld.service

Вводим *cat /usr/lib/systemd/system/iptables.service*. Описание накстроек конфликтов: в данном юните параметр Conflicts отсутствует, что означает, что конфликтов с другими службами не указано. Хотя в юните iptables не указаны конфликты, мы знаем из предыдущей конфигурации firewalld, что firewalld указывает iptables как конфликтующую службу. Это означает, что если firewalld работает, то iptables не должен быть запущен одновременно, так как это может привести к конфликтам в управлении firewalld. (рис. 21)

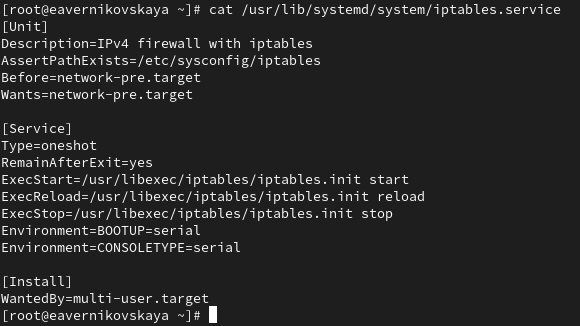


Рис. 21: Команда cat /usr/lib/systemd/system/iptables.service

Выгружаем службу iptables (на всякий случай, чтобы убедиться, что данная служба не загружена в систему): *systemctl stop iptables*. После загружаем службу firewalld (рис. 22)

Выгрузка службы iptables и загрузка службы firewalld

Рис. 22: Выгрузка службы iptables и загрузка службы firewalld

Далее блокируем запуск iptables, введя: *systemctl mask iptables*. При этом будет создана символическая ссылка на /dev/null для /etc/systemd/system/iptables.service. Поскольку юнитфайлы в /etc/systemd имеют приоритет над файлами в /usr/lib/systemd, то это сделает невозможным случайный запуск сервиса iptables (рис. 23)

Блокировка запуска iptables

Рис. 23: Блокировка запуска iptables

Пробуем запустить iptables. При попытке запустить iptables появляется сообщение об ошибке, указывающее, что служба замаскирована и по этой причине не может быть запущена (рис. 24)

Попытка запустить iptables

Рис. 24: Попытка запустить iptables

Далее пробуем добавить iptables в автозапуск. Сервис будет неактивен, а статус загрузки отобразится как замаскированный (рис. 25), (рис. 26)

Попытка добавить iptables в автозапуск

Рис. 25: Попытка добавить iptables в автозапуск

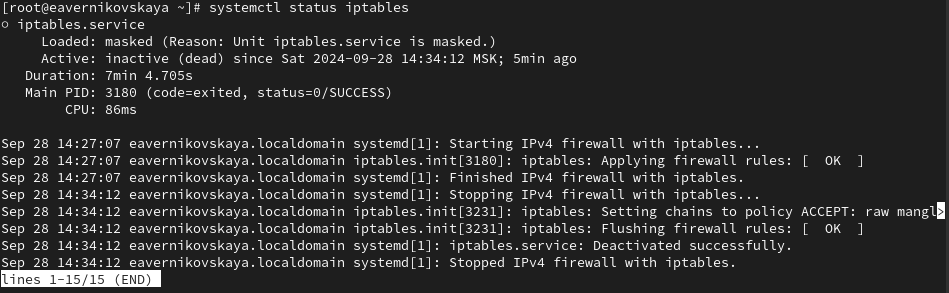


Рис. 26: Статус iptables (2)

## 3.3 Изолируемые цели

Получем список всех активных загруженных целей: *systemctl –type=target*, а затем список всех целей: *systemctl –type=target –all* (рис. 27), (рис. 28)

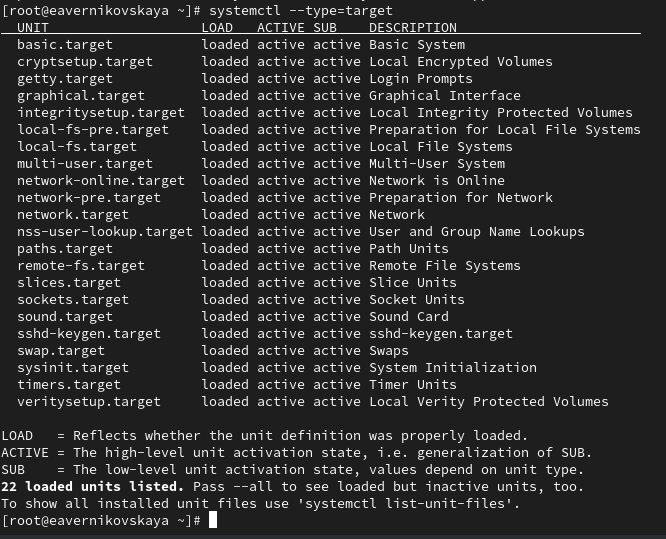


Рис. 27: Cписок всех активных загруженных целей

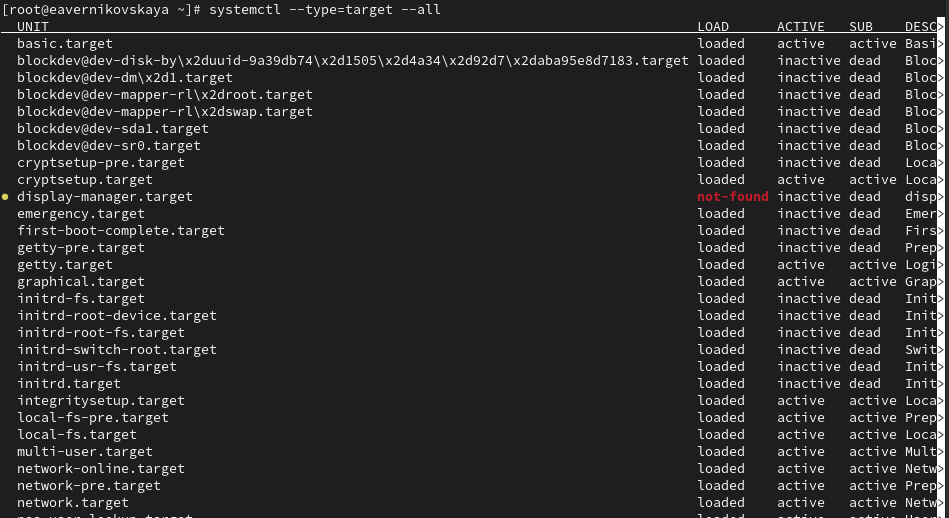


Рис. 28: Список всех целей

Переходим в каталог systemd и находим список всех целей, которые можно изолировать:

* *cd /usr/lib/systemd/system*
* *grep Isolate* .target\* (рис. 29)

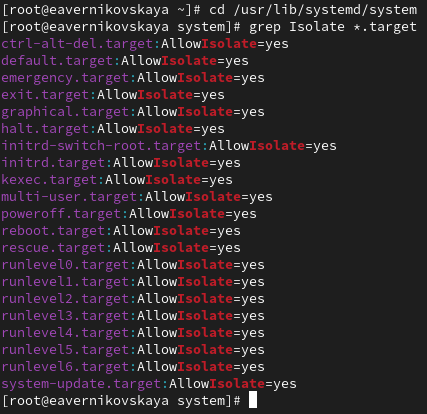


Рис. 29: Список всех целей, которые можно изолировать

Далее переключаем операционную систему в режим восстановления: *systemctl isolate rescue.target* (рис. 30), (рис. 31)

Переключение операционной системы в режим восстановления

Рис. 30: Переключение операционной системы в режим восстановления

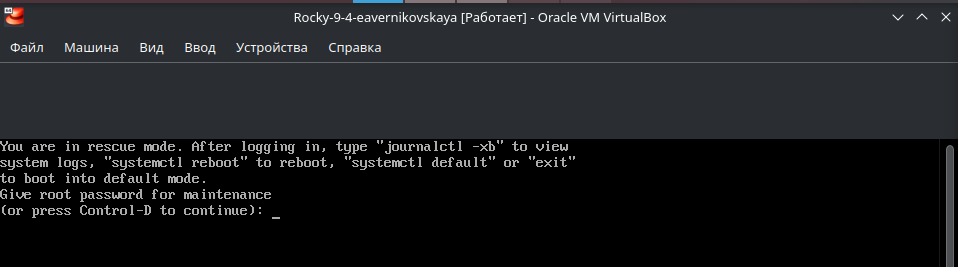


Рис. 31: Режим восстановления

Переходим в режим root и перезапускаем операционную систему: *systemctl isolate reboot.target* (рис. 32), (рис. 33)

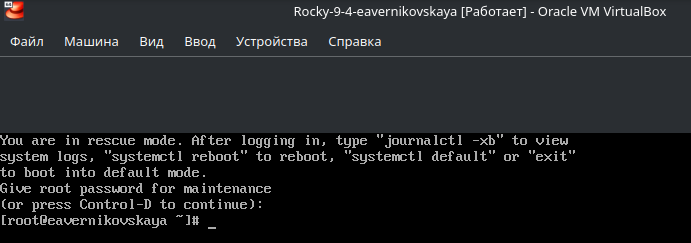


Рис. 32: Режим root

Перезапуск операционной системы

Рис. 33: Перезапуск операционной системы

## 3.4 Цель по умолчанию

ПОлучаем права администратора. Далее выводим на экран цель, установленную по умолчанию: *systemctl get-default*. Сейчас графический режим (рис. 34)

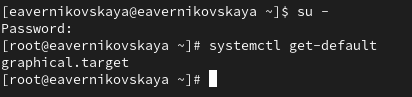


Рис. 34: Цель, установленна по умолчанию

Для установки цели по умолчанию используется команда *systemctl set-default*. Запускаем по умолчанию текстовый режим введя команду *systemctl set-default multi-user.target* и перезагружаем систему командой *reboot* (рис. 35), (рис. 36)

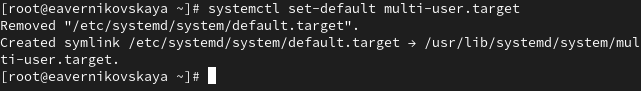


Рис. 35: Запуск по умолчанию текстового режима

Перезагрузка системы (1)

Рис. 36: Перезагрузка системы (1)

Система загрузилась в текстовом режиме. Далее получаем полномочия пользователя root и запускаем по умолчанию графический режим введя команду *systemctl set-default graphical.target*. После снова перезагружаем систему командой *reboot* (рис. 37), (рис. 38), (рис. 39), (рис. 40)

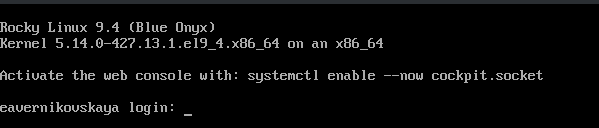


Рис. 37: Система в текстовом режиме

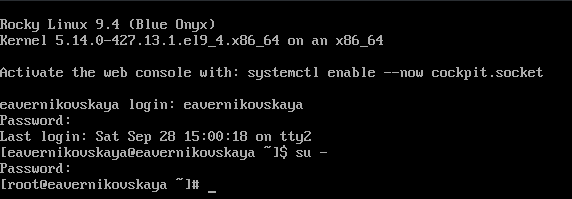


Рис. 38: Учётная запись пользователя root

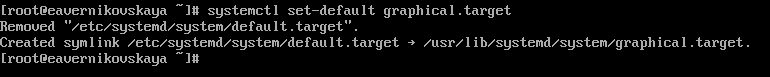


Рис. 39: Запуск по умолчанию графического режима

Перезагрузка системы (2)

Рис. 40: Перезагрузка системы (2)

Система загрузилась в графическом режиме. Всё хорошо! (рис. 41)

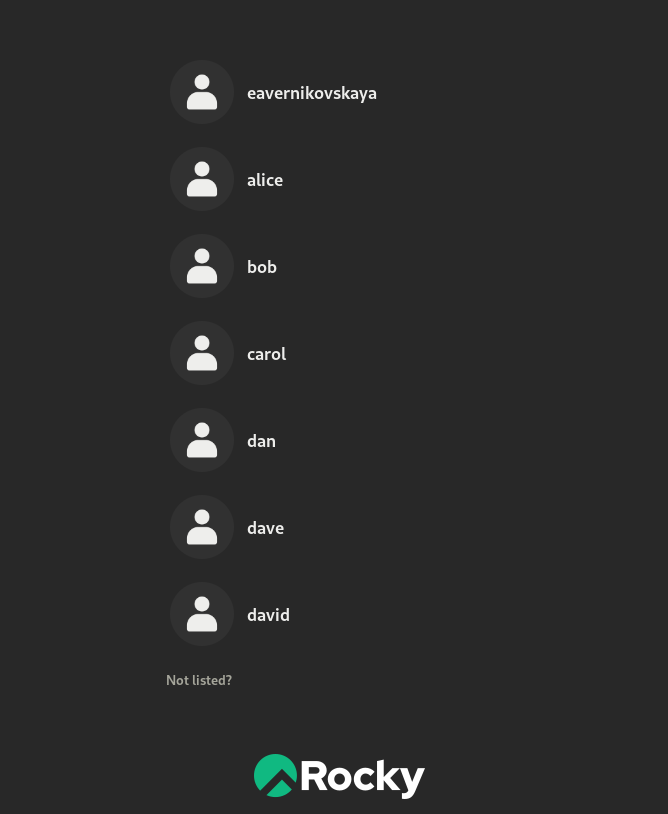


Рис. 41: Система в графическом режиме

# 4 Контрольные вопросы + ответы

1. Что такое юнит (unit)? Приведите примеры.

Юнит (или unit) в контексте систем управления, таких как systemd, — это абстрактное представление ресурса или сервиса, которым управляет система. Каждый юнит описывает один ресурс и содержит метаданные и инструкции о том, как управлять этим ресурсом.

Основные типы юнитов:

* Service Unit (.service):
  + Описывает службы или демоны, которые должны быть запущены в системе.
  + Пример: httpd.service для Apache HTTP Server.
* Socket Unit (.socket):
  + Управляет сокетами, которые могут активировать службы при получении соединений.
  + Пример: cups.socket для печатного сервиса CUPS.
* Target Unit (.target):
  + Группирует другие юниты и позволяет управлять целыми наборами.
  + Пример: multi-user.target, который аналогичен режиму “консоль” в других системах.
* Device Unit (.device):
  + Представляет физические или виртуальные устройства в системе.
  + Пример: dev-sda1.device для дискового раздела /dev/sda1.
* Mount Unit (.mount):
  + Описывает точки монтирования файловых систем..
  + Пример: mnt-data.mount для монтирования файловой системы в /mnt/data.

1. Какая команда позволяет вам убедиться, что цель больше не входит в список автоматического запуска при загрузке системы?

systemctl is-enabled “имя\_юнита” (пример: systemctl is-enabled vsftpd.target) (рис. 42)

Вопрос №2

Рис. 42: Вопрос №2

1. Какую команду вы должны использовать для отображения всех сервисных юнитов, которые в настоящее время загружены?

systemctl list-units –type=srvice (рис. 43)

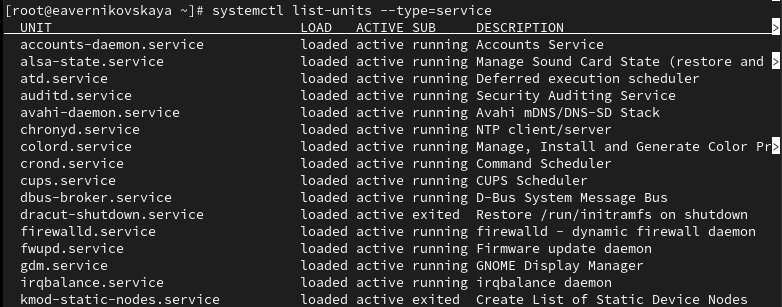


Рис. 43: Вопрос №3

1. Как создать потребность (wants) в сервисе?

Нужно внести всю необходимую информацию в переменную “Wants”, которая находится в файле имя\_сервиса.service

1. Как переключить текущее состояние на цель восстановления (rescue target)?

systemctl set-default rescue.target.

1. Поясните причину получения сообщения о том, что цель не может быть изолирована.

Изолируя цель, мы запускаем эту цель со всеми её зависимостями. Не все цели могут быть изолированы (в случае, если цель является неотъемлемой частью system)

1. Вы хотите отключить службу systemd, но, прежде чем сделать это, вы хотите узнать, какие другие юниты зависят от этой службы. Какую команду вы бы использовали?

systemctl list-dependencies “имя\_юнита” –reverse (пример: systemctl list-dependencies firewalld.service –reverse) (рис. 44)

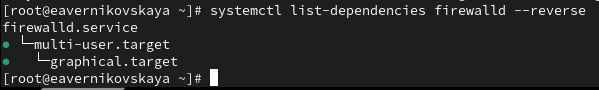


Рис. 44: Вопрос №4

# 5 Выводы

В ходе выполнения лабораторной работы мы получили навыки управления системными службами операционной системы посредством systemd.

# 6 Список литературы

1. Лаборатораня работа №5 [Электронный ресурс] URL: https://esystem.rudn.ru/pluginfile.php/2400698/mod\_resource/content/4/006-service.pdf