

Отчёт по лабораторной работе №5

Дисциплина: Основы администрирования операционных систем

Бызова Мария Олеговна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Управление сервисами	7
3.2	Конфликты юнитов	9
3.3	Изолируемые цели	12
3.4	Цель по умолчанию	13
4	Ответы на контрольные вопросы	15
5	Выводы	17
	Список литературы	18

Список иллюстраций

3.1	Открытие режима работа суперпользователя, проверка статуса, установка и запуск службы Very Secure FTP.	7
3.2	Проверка статуса службы Very Secure FTP.	7
3.3	Добавление службы Very Secure FTP в автозапуск и проверка её статуса.	8
3.4	Удаление службы Very Secure FTP из автозапуска и проверка её статуса.	8
3.5	Вывод на экран символических ссылок, добавление службы Very Secure FTP в автозапуск, проверка статуса службы.	9
3.6	Список зависимостей юнита.	9
3.7	Список юнитов, которые зависят от данного юнита.	9
3.8	Получение полномочий администратора и установка iptables.	9
3.9	Проверка статуса firewalld и iptables.	10
3.10	Попытка запуска firewalld и iptables.	10
3.11	Настройки конфликтов для юнитов.	11
3.12	Выгрузка службы iptables, загрузка службы firewalld, блокировка запуска iptables, попытка запуска iptables и добавления iptables в автозапуск.	12
3.13	Открытие каталога. Нахождение списка всех целей, которые можно изолировать. Переключение операционной системы в режим восстановления.	12
3.14	Перезапуск операционной системы.	13
3.15	Получение полномочий администратора, установка запуска по умолчанию текстового режима, последующая перезагрузка системы.	13
3.16	Загрузка системы в текстовом режиме, получение полномочий администратора, установка запуска по умолчанию графического режима и последующая перезагрузка системы.	13
3.17	Загрузка системы в графическом режиме.	14

Список таблиц

1 Цель работы

Целью данной работы является получение навыков управления системными службами операционной системы посредством systemd.

2 Задание

1. Выполните основные операции по запуску (останову), определению статуса, добавлению (удалению) в автозапуск и пр. службы Very Secure FTP (раздел 5.4.1).
2. Продемонстрируйте навыки по разрешению конфликтов юнитов для служб firewalld и iptables (раздел 5.4.2).
3. Продемонстрируйте навыки работы с изолированными целями (разделы 5.4.3, 5.4.4).

3 Выполнение лабораторной работы

3.1 Управление сервисами

Для начала получим полномочия администратора su -. Затем проверим 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. Установим службу Very Secure FTP: `dnf -y install vsftpd` и запустим: `systemctl start vsftpd` (рис. 3.1).

```
root@moonhive:~# su
root@moonhive:~# systemctl status vsftpd
root@moonhive:~# dnf -y install vsftpd
Last metadata expiration check: 5:08:13 Sep 28 on Sun 28 Sep 2024 07:39:46 PM MSD.
Dependencies resolved.
Package
Installing
vsftpd
Architecture
x86_64
Version
3.0.5-5.el9
Repository
appstream
Size
157 k
Transaction Summary
Install 1 Package
Total download size: 157 k
Installed size: 367 k
Downloading Packages
vsftpd-3.0.5-5.el9.x86_64.rpm
Total
133 kB/s | 157 kB | 00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing
Installing
vsftpd-3.0.5-5.el9.x86_64
Running scriptlet: vsftpd-3.0.5-5.el9.x86_64
Installing
vsftpd-3.0.5-5.el9.x86_64
Installed:
vsftpd-3.0.5-5.el9.x86_64
Complete!
root@moonhive:~# systemctl start vsftpd
root@moonhive:~#
```

Рис. 3.1: Открытие режима работы суперпользователя, проверка статуса, установка и запуск службы Very Secure FTP.

Снова проверим статус службы Very Secure FTP: `systemctl status vsftpd`. Вывод команды показывает, что служба в настоящее время работает, но не будет активирована при перезапуске операционной системы (рис. 3.2).

```
root@moonhive:~# systemctl status vsftpd
vsftpd.service - vsftpd ftp daemon
Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled; preset: disabled)
Active: inactive (dead) since Sun 2024-09-28 20:40:12 MSD; 44s ago
Process: 6028 ExecStartPost=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited, status=0/SUCCESS)
Main PID: 6028 (vsftpd)
Tasks: 1 (limit: 26872)
Memory: 268.0K
CGroup: /system.slice/vsftpd.service
└─ 6028 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Sep 28 19:40:18 moonhive.localdomain systemd[1]: Starting vsftpd ftp daemon...
Sep 28 19:40:18 moonhive.localdomain systemd[1]: Started vsftpd ftp daemon.
root@moonhive:~#
```

Рис. 3.2: Проверка статуса службы Very Secure FTP.

Добавим службу Very Secure FTP в автозапуск при загрузке операционной системы, используя команду `systemctl enable vsftpd`. Затем проверим статус службы и удалим службу из автозапуска, используя команду `systemctl disable vsftpd`, и снова проверим её статус (рис. 3.3, рис. 3.4).

```
root@mobihzova ~]# systemctl enable vsftpd
Created symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service → /usr/lib/systemd/system/vsftpd.service.
root@mobihzova ~]# systemctl status vsftpd
vsftpd.service - Vsftpd ftp daemon
Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)
Active: active (running) since Sun 2024-09-29 19:40:19 MSK; 2min 18s ago
Main PID: 4929 (vsftpd)
Tasks: 1 (limit: 24674)
Memory: 708.0K
CPU: 3ms
CGroup: /system.slice/vsftpd.service
└─4929 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Sep 29 19:40:19 mobihzova.localdomain systemd[1]: Starting Vsftpd ftp daemon...
Sep 29 19:40:19 mobihzova.localdomain systemd[1]: Started Vsftpd ftp daemon.
root@mobihzova ~]#
```

Рис. 3.3: Добавление службы Very Secure FTP в автозапуск и проверка её статуса.

```
root@mobihzova ~]# systemctl disable vsftpd
Removed "/etc/systemd/system/multi-user.target.wants/vsftpd.service".
root@mobihzova ~]# systemctl status vsftpd
vsftpd.service - Vsftpd ftp daemon
Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled; preset: disabled)
Active: active (running) since Sun 2024-09-29 19:40:19 MSK; 3min 46s ago
Main PID: 4929 (vsftpd)
Tasks: 1 (limit: 24674)
Memory: 708.0K
CPU: 3ms
CGroup: /system.slice/vsftpd.service
└─4929 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Sep 29 19:40:19 mobihzova.localdomain systemd[1]: Starting Vsftpd ftp daemon...
Sep 29 19:40:19 mobihzova.localdomain systemd[1]: Started Vsftpd ftp daemon.
root@mobihzova ~]#
```

Рис. 3.4: Удаление службы Very Secure FTP из автозапуска и проверка её статуса.

Далее выведем на экран символические ссылки, ответственные за запуск различных сервисов. Отображается, что ссылка на `vsftpd.service` не существует. Снова добавляем службу Very Secure FTP в автозапуск: `systemctl enable vsftpd` и выводим на экран символические ссылки, ответственные за запуск различных сервисов. Вывод команды показывает, что создана символическая ссылка для файла `/usr/lib/systemd/system/vsftpd.service` в каталоге `/etc/systemd/system/multi-user.target.wants`. Проверяем статус службы Very Secure FTP: `systemctl status vsftpd`. Теперь мы видим, что для файла юнита состояние изменено с `disabled` на `enabled` (рис. 3.5).


```

root@mobihzova ~]# systemctl enable vsftpd
Created symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service → /usr/lib/systemd/system/vsftpd.service.
root@mobihzova ~]# systemctl status vsftpd
vsftpd.service - vsftpd ftp daemon
Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled; preset: disabled)
Active: active (running) since Sun 2024-09-20 10:40:18 MSK; 5min ago
Main PID: 6002 (vsftpd)
Tasks: 1 (limit: 2048)
Memory: 108.0K
CPU: 3ms
CGroup: /system.slice/vsftpd.service
          └─vsftpd /usr/sbin/vsftpd -j /usr/sbin/vsftpd.conf

Sep 20 10:40:18 mobihzova.localdomain systemd[1]: Starting vsftpd ftp daemon...
Sep 20 10:40:18 mobihzova.localdomain systemd[1]: Started vsftpd ftp daemon.
root@mobihzova ~]#

```

Рис. 3.5: Вывод на экран символических ссылок, добавление службы Very Secure FTP в автозапуск, проверка статуса службы.

Выведем на экран список зависимостей юнита: `systemctl list-dependencies vsftpd` и список юнитов, которые зависят от данного юнита: `systemctl list-dependencies vsftpd --reverse` (рис. 3.6, рис. 3.7).

```

[root@mobihzova ~]# systemctl list-dependencies vsftpd
vsftpd.service
├─system.slice
├─sysinit.target
│   ├──dev-hugepages.mount
│   ├──dev-mqueue.mount
│   └─dracut-shutdown.service
├─iscsi-onboot.service
├─iscsi-starter.service
├─kmod-static-nodes.service
├─ldconfig.service
├─lvm2-lvmpolld.socket
├─lvm2-monitor.service
├─multipathd.service
├─nis-domainname.service
├─plymouth-read-write.service
├─plymouth-start.service
├─proc-sys-fs-binfmt_misc.automount
├─selinux-autorelabel-mark.service
└─sys-fs-fuse-connections.mount

```

Рис. 3.6: Список зависимостей юнита.

```

root@mobihzova ~]# systemctl list-dependencies vsftpd --reverse
vsftpd.service
└─multi-user.target
  └─systemd.target
root@mobihzova ~]#

```

Рис. 3.7: Список юнитов, которые зависят от данного юнита.

3.2 Конфликты юнитов

Получим полномочия администратора `su -` и установим `iptables`: `dnf -y install iptables*` (рис. 3.8).

```

root@mobihzova ~]# dnf -y install iptables*
Last metadata expiration check: 0:10:22 ago on Sun 20 Sep 2024 07:39:46 PM MSK.
Package iptables-libs-1.8.10-4.el9_4.x86_64 is already installed.
Package iptables-rt-1.8.10-4.el9_4.x86_64 is already installed.
Dependencies resolved.
=====
Package                               Architecture      Version              Repository           Size
=====
Installing:
iptables                               x86_64            1.8.10-4.el9_4       appstream            14
iptables-services                     x86_64            1.8.10-4.el9_4       appstream            17
iptables                               x86_64            1.8.10-4.el9_4       baseos               46
Transaction Summary
-----
Install 3 Packages

Total download size: 72 k
Installed size: 142 k
Downloading Packages:
(1/3) iptables-libs-1.8.10-4.el9_4.x86_64.rpm                00 kB/s |  40 B | 00:00
(2/3) iptables-rt-1.8.10-4.el9_4.x86_64.rpm                 35 kB/s |  14 B | 00:00
(3/3) iptables-services-1.8.10-4.el9_4.x86_64.rpm            23 kB/s |  17 B | 00:00
-----
Total: 45 kB/s |  72 B | 00:01
Running transaction check

```

Рис. 3.8: Получение полномочий администратора и установка `iptables`.

Далее проверим статус firewalld и iptables: `systemctl status firewalld` и `systemctl status iptables` (рис. 3.9).

```
root@mozhova:~# systemctl status firewalld
firewalld.service - Firewall - dynamic Firewall daemon
Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; preset: enabled)
Active: active (running) since Sun 2024-09-29 18:20:15 MSK; 2min ago
Docs: man:firewalld(1)
Main PID: 796 (firewalld)
Tasks: 2 (limit: 24674)
Memory: 12.0M
CPU: 25ms
CGroup: /system.slice/firewalld.service
└─ /usr/bin/python3 /usr/sbin/firewalld --nofork --quiet

Sep 29 18:20:16 mozhova.localdomain systemd[1]: Starting Firewall - dynamic Firewall daemon...
Sep 29 18:20:16 mozhova.localdomain systemd[1]: Started Firewall - dynamic Firewall daemon.
root@mozhova:~# systemctl status iptables
iptables.service - IPv4 firewall with iptables
Loaded: loaded (/usr/lib/systemd/system/iptables.service; disabled; preset: disabled)
Active: inactive (dead)
Docs: man:iptables(8)
Main PID: 0 (code=exited, status=0/SUCCESS)
Active: inactive (dead)
root@mozhova:~#
```

Рис. 3.9: Проверка статуса firewalld и iptables.

Попробуем запустить firewalld и iptables: `systemctl start firewalld` и `systemctl start iptables`. Мы видим, что при запуске одной службы вторая деактивируется или не запускается (рис. 3.10).

```
root@mozhova:~# systemctl start firewalld
root@mozhova:~# systemctl start iptables
root@mozhova:~# systemctl status iptables
iptables.service - IPv4 firewall with iptables
Loaded: loaded (/usr/lib/systemd/system/iptables.service; disabled; preset: disabled)
Active: active (running) since Sun 2024-09-29 18:32:09 MSK; 28s ago
Process: 5488 ExecStart=/usr/sbin/iptables/iptables --wait start (code=exited, status=0/SUCCESS)
Main PID: 5488 (code=exited, status=0/SUCCESS)
CPU: 8ms

Sep 29 18:32:08 mozhova.localdomain systemd[1]: Starting IPv4 firewall with iptables...
Sep 29 18:32:08 mozhova.localdomain systemd[1]: iptables: Applying Firewall rules: [ OK ]
Sep 29 18:32:08 mozhova.localdomain systemd[1]: Finished IPv4 firewall with iptables.
root@mozhova:~# systemctl status firewalld
firewalld.service - Firewall - dynamic Firewall daemon
Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; preset: enabled)
Active: active (running) since Sun 2024-09-29 18:32:08 MSK; 3s ago
Docs: man:firewalld(1)
Main PID: 5497 (firewalld)
Tasks: 2 (limit: 24674)
Memory: 12.0M
CPU: 25ms
CGroup: /system.slice/firewalld.service
└─ /usr/bin/python3 /usr/sbin/firewalld --nofork --quiet

Sep 29 18:32:08 mozhova.localdomain systemd[1]: Starting Firewall - dynamic Firewall daemon...
Sep 29 18:32:08 mozhova.localdomain systemd[1]: Started Firewall - dynamic Firewall daemon.
root@mozhova:~#
```

Рис. 3.10: Попытка запуска firewalld и iptables.

Введем `cat /usr/lib/systemd/system/firewalld.service` и опишем настройки конфликтов для этого юнита при наличии, далее введём `cat /usr/lib/systemd/system/iptables.service` и опишем настройки конфликтов для этого юнита (рис. 3.11).

```

root@mobihzova ~]# cat /usr/lib/systemd/system/firewalld.service
[Unit]
Description=firewalld - dynamic firewall daemon
Before=network-pre.target
Wants=network-pre.target
After=dbus.service
After=polkit.service
Conflicts=iptables.service ip6tables.service ebtables.service ipset.service nftables.service
Documentation=man:firewalld(1)

[Service]
EnvironmentFile=/etc/sysconfig/firewalld
ExecStart=/usr/sbin/firewalld --nofork --nopid $FIREWALLD_ARGS
ExecReload=/bin/kill -HUP $MAINPID
# suppress to log debug and error output also to /var/log/messages
StandardOutput=null
StandardError=null
Type=dbus
BusName=org.fedoraproject.FirewallD1
KillMode=mixed

[Install]
WantedBy=multi-user.target
Alias=dbus-org.fedoraproject.FirewallD1.service
root@mobihzova ~]# cat /usr/lib/systemd/system/iptables.service
[Unit]
Description=IPv4 firewall with iptables
AssertPathExists=/etc/sysconfig/iptables
Before=network-pre.target
Wants=network-pre.target

[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/usr/libexec/iptables/iptables.init start
ExecReload=/usr/libexec/iptables/iptables.init reload
ExecStop=/usr/libexec/iptables/iptables.init stop
Environment=BOOTUP=serial
Environment=CONSOLETYPE=serial

[Install]
WantedBy=multi-user.target
root@mobihzova ~]#

```

Рис. 3.11: Настройки конфликтов для юнитов.

Выгрузим службу iptables (на всякий случай, чтобы убедиться, что данная служба не загружена в систему): `systemctl stop iptables` и загрузим службу firewalld: `systemctl start firewalld`. Далее заблокируем запуск iptables, введя: `systemctl mask iptables`. Видим, как создана символическая ссылка на /dev/null для /etc/systemd/system/iptables.service. Поскольку юнит-файлы в /etc/systemd имеют приоритет над файлами в /usr/lib/systemd, то это сделает невозможным случайный запуск сервиса iptables. Для проверки попробуем запустить iptables: `systemctl start iptables`. После попытки запуска появилось сообщение об ошибке, указывающее, что служба замаскирована и по этой причине не может быть запущена. Теперь попробуем добавить iptables в автозапуск: `systemctl enable iptables`. Показывает, что сервис неактивен, а статус загрузки отображается как замаскированный (рис. 3.12).

```

[root@mobihzova ~]# systemctl stop iptables
[root@mobihzova ~]# systemctl start firewalld
[root@mobihzova ~]# systemctl mask iptables
Created symlink /etc/systemd/system/iptables.service → /dev/null.
[root@mobihzova ~]# systemctl start iptables
Failed to start iptables.service: Unit iptables.service is masked.
[root@mobihzova ~]# systemctl enable iptables
Failed to enable unit: Unit file /etc/systemd/system/iptables.service is masked.
[root@mobihzova ~]#

```

Рис. 3.12: Выгрузка службы iptables, загрузка службы firewalld, блокировка запуска iptables, попытка запуска iptables и добавления iptables в автозапуск.

3.3 Изолируемые цели

Получим полномочия администратора su – и перейдём в каталог systemd, найдём список всех целей, которые можно изолировать: `cd /usr/lib/systemd/system` `grep Isolate *.target`. На следующем шаге переключим операционную систему в режим восстановления: `systemctl isolate rescue.target` (рис. 3.13).

```

[root@mobihzova ~]# systemctl start iptables
Failed to start iptables.service: Unit iptables.service is masked.
[root@mobihzova ~]# systemctl enable iptables
Failed to enable unit: Unit file /etc/systemd/system/iptables.service is masked.
[root@mobihzova ~]# cd /usr/lib/systemd/system
[root@mobihzova system]# grep Isolate *.target
ctrl-alt-del.target:AllowIsolate=yes
default.target:AllowIsolate=yes
emergency.target:AllowIsolate=yes
exit.target:AllowIsolate=yes
graphical.target:AllowIsolate=yes
halt.target:AllowIsolate=yes
initrd-switch-root.target:AllowIsolate=yes
initrd.target:AllowIsolate=yes
kexec.target:AllowIsolate=yes
multi-user.target:AllowIsolate=yes
poweroff.target:AllowIsolate=yes
reboot.target:AllowIsolate=yes
rescue.target:AllowIsolate=yes
runlevel0.target:AllowIsolate=yes
runlevel1.target:AllowIsolate=yes
runlevel2.target:AllowIsolate=yes
runlevel3.target:AllowIsolate=yes
runlevel4.target:AllowIsolate=yes
runlevel5.target:AllowIsolate=yes
runlevel6.target:AllowIsolate=yes
system-update.target:AllowIsolate=yes
[root@mobihzova system]# systemctl isolate rescue.target

```

Рис. 3.13: Открытие каталога. Нахождение списка всех целей, которые можно изолировать. Переключение операционной системы в режим восстановления.

Как только операционная система переключилась в режим восстановления вводим пароль root. После чего перезапустим операционную систему следующим образом: `systemctl isolate reboot.target` (рис. 3.14).

```

You are in rescue mode. After logging in, type: "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, "systemctl default" or "exit"
to boot into default mode.
Give root password for maintenance
(or press Control-D to continue):

```

Рис. 3.14: Перезапуск операционной системы.

3.4 Цель по умолчанию

Получим полномочия администратора `su` – и выведем на экран цель, установленную по умолчанию: `systemctl get-default`. Для установки цели по умолчанию используется команда `systemctl set-default`. В нашем случае для запуска по умолчанию текстового режима введём `systemctl set-default multi-user.target`. После чего перезагрузим систему командой `reboot` (рис. 3.15).

```

[mobihzova@mobihzova ~]$ su -
Password:
[root@mobihzova ~]# systemctl get-default
graphical.target
[root@mobihzova ~]# systemctl set-default
too few arguments.
[root@mobihzova ~]# systemctl set-default multi-user.target
Removed "/etc/systemd/system/default.target".
Created symlink /etc/systemd/system/default.target → /usr/lib/systemd/system/multi-user.target.
[root@mobihzova ~]# reboot

```

Рис. 3.15: Получение полномочий администратора, установка запуска по умолчанию текстового режима, последующая перезагрузка системы.

Убедимся, что система загрузилась в текстовом режиме, после чего получим полномочия администратора (для начала зайдём в пользователя `mobihzova`, а затем в режим администратора). Для запуска по умолчанию графического режима введём `systemctl set-default graphical.target` и вновь перегрузим систему командой `reboot`. Убедимся, что система загрузилась в графическом режиме (рис. 3.16, рис. 3.17).

```

Rocky Linux 9.4 (Blue Onyx)
Kernel 5.14.0-427.13.1.el9_4.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

mobihzova login:
mobihzova login: mobihzova
Password:
Last login: Sun Sep 29 20:00:55 on tty3
[mobihzova@mobihzova ~]$ su -
Password:
[root@mobihzova ~]# systemctl set-default graphical.target
Removed "/etc/systemd/system/default.target".
Created symlink /etc/systemd/system/default.target → /usr/lib/systemd/system/graphical.target.
[root@mobihzova ~]# reboot

```

Рис. 3.16: Загрузка системы в текстовом режиме, получение полномочий администратора, установка запуска по умолчанию графического режима и последующая перезагрузка системы.

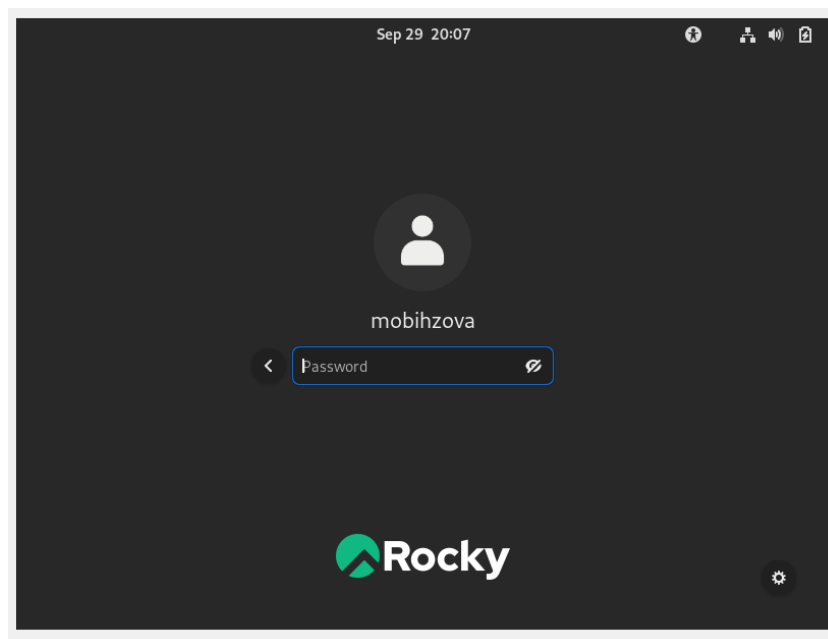


Рис. 3.17: Загрузка системы в графическом режиме.

4 Ответы на контрольные вопросы

1. Что такое юнит (unit)? Приведите примеры. Unit – объект, которым может управлять система.
2. Какая команда позволяет вам убедиться, что цель больше не входит в список автоматического запуска при загрузке системы? `systemctl is-enable` “имя_юнита” (пример: `systemctl is-enable vsftpd.service`).

```
[root@mobihzova mobihzova]# systemctl is-enabled vsftpd.service
enabled
[root@mobihzova mobihzova]#
```

3. Какую команду вы должны использовать для отображения всех сервисных юнитов, которые в настоящее время загружены? `system list-units`.

```
[root@mobihzova mobihzova]# systemctl list-units
UNIT
proc-sys-fs-binfmt_misc.automount
sys-devices-pci0000:00-0000:00:01.1-ata2-host1-target1:0:0-1:0:0:0-block-sr0.
sys-devices-pci0000:00-0000:00:03.0-net-enp0s3.device
sys-devices-pci0000:00-0000:00:05.0-sound-card0-controlC0.device
sys-devices-pci0000:00-0000:00:0d.0-ata3-host2-target2:0:0-2:0:0:0-block-sda.
sys-devices-pci0000:00-0000:00:0d.0-ata3-host2-target2:0:0-2:0:0:0-block-sda.
sys-devices-pci0000:00-0000:00:0d.0-ata3-host2-target2:0:0-2:0:0:0-block-sda.
sys-devices-platform-serial8250-tty-ttyS0.device
sys-devices-platform-serial8250-tty-ttyS1.device
sys-devices-platform-serial8250-tty-ttyS2.device
sys-devices-platform-serial8250-tty-ttyS3.device
sys-devices-virtual-block-dm\x2d0.device
sys-devices-virtual-block-dm\x2d1.device
sys-devices-virtual-misc-rfkill.device
sys-module-configfs.device
sys-module-fuse.device
sys-subsystem-net-devices-enp0s3.device
-.mount
boot.mount
dev-hugepages.mount
dev-mqueue.mount
run-credentials-systemd\x2dsysctl.service.mount
```

4. Как создать потребность (wants) в сервисе? Нужно внести всю необходимую информацию в переменную “Wants”, которая находится в файле имя_сервиса.service.

5. Как переключить текущее состояние на цель восстановления (rescue target)?
systemctl set-default rescue.target.

```
lines 2-24  
[root@mobihzova mobihzova]# systemctl set-default rescue.target
```

6. 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).
7. 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).

```
[root@mobihzova mobihzova]# systemctl list-dependencies firewalld.service --rever  
se
```


5 Выводы

В ходе выполнения лабораторной работы были получены навыки управления системными службами операционной системы посредством systemd.

Список литературы

1. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ-Петербург, 2010.
2. Колисниченко Д. Н. Самоучитель системного администратора Linux. — СПб. : БХВ-Петербург, 2011. — (Системный администратор).
3. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — (Классика Computer Science).
4. Neil N. J. Learning CentOS: A Beginners Guide to Learning Linux. — CreateSpace Independent Publishing Platform, 2016.
5. Unix и Linux: руководство системного администратора / Э. Немец, Г. Снайдер, Т.Хейн, Б. Уэйли, Д. Макни. — 5-е изд. — СПб. : ООО «Диалектика», 2020.