

Лабораторная работа №5

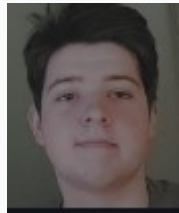
Артём Дмитриевич Петлин

2025-10-03

1. Информация
2. Цель работы
3. Задание
4. Теоретическое введение
5. Выполнение лабораторной работы
6. Выводы

1. Информация

- Петлин Артём Дмитриевич
- студент
- группа НПИбд-02-24
- Российский университет дружбы народов
- 1132246846@pfur.ru
- https://github.com/hikrim/study_2025-2026_os2



2. Цель работы

2.1 Цель работы

Получить навыки управления системными службами операционной системы посредством systemd.

3. 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).

4. Теоретическое введение

Systemd (system daemon) — система инициализации служб операционной системы. Под системой инициализации в данном случае понимается набор скриптов, выполняемых на этапе загрузки операционной системы. Для выполнения операций над системными службами в systemd используются юниты (units) — файлы конфигурации с определённым синтаксисом.

5. Выполнение лабораторной работы

```
adpetlin@adpetlin:~$ su -  
Password:  
Last login: Fri Sep 26 16:07:05 MSK 2025 on pts/0  
root@adpetlin:~# systemctl status vsftpd  
Unit vsftpd.service could not be found.
```

Рисунок 1

Получаем полномочия администратора. Проверяем статус службы Very Secure FTP.

```
Installed:  
  vsftpd-3.0.5-9.el10.x86_64  
  
Complete!  
root@adpetlin:~#
```

Рисунок 2

Устанавливаем службу Very Secure FTP.

5.3 Ход работы

Запускаем установленную службу.
Снова проверяем статус службы.
Служба будет работать, но не
активируется при перезагрузке
системы.

```
root@adpetlin:~# systemctl start vsftpd
root@adpetlin:~# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled; preset: disabled)
   Active: active (running) since Fri 2025-10-03 09:54:47 MSK; 4s ago
     Invocation: 1ef9fc7f179146448ed58ae26982cffe
    Process: 8484 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited, status=0/SUCCESS)
   Main PID: 8485 (vsftpd)
      Tasks: 1 (limit: 36001)
     Memory: 756K (peak: 1.2M)
        CPU: 6ms
    CGroup: /system.slice/vsftpd.service
           └─8485 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Oct 03 09:54:47 adpetlin systemd[1]: Starting vsftpd.service - Vsftpd ftp daemon...
Oct 03 09:54:47 adpetlin systemd[1]: Started vsftpd.service - Vsftpd ftp daemon.
root@adpetlin:~#
```

Рисунок 3

5.4 Ход работы

```
root@adpetlin:~# systemctl enable vsftpd
Created symlink '/etc/systemd/system/multi-user.target.wants/vsftpd.service' → '/usr/lib/systemd/system/vsftpd.service'.
root@adpetlin:~# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-10-03 09:54:47 MSK; 1min 17s ago
 Invocation: 1ef9fc7f179146448ed58ae26982cffe
    Main PID: 8485 (vsftpd)
      Tasks: 1 (limit: 36001)
     Memory: 756K (peak: 1.2M)
        CPU: 6ms
    CGroup: /system.slice/vsftpd.service
            └─8485 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Oct 03 09:54:47 adpetlin systemd[1]: Starting vsftpd.service - Vsftpd ftp daemon...
Oct 03 09:54:47 adpetlin systemd[1]: Started vsftpd.service - Vsftpd ftp daemon.
root@adpetlin:~#
```

Рисунок 4

Добавляем службу в автозапуск при загрузке системы, проверяем её статус.

5.5 Ход работы

```
root@adpetlin:~# systemctl disable vsftpd
Removed '/etc/systemd/system/multi-user.target.wants/vsftpd.service'.
root@adpetlin:~# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled; preset: disabled)
   Active: active (running) since Fri 2025-10-03 09:54:47 MSK; 1min 44s ago
 Invocation: 1ef9fc7f179146448ed58ae26982cffe
    Main PID: 8485 (vsftpd)
      Tasks: 1 (limit: 36001)
     Memory: 756K (peak: 1.2M)
        CPU: 6ms
     CGroup: /system.slice/vsftpd.service
            └─8485 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Oct 03 09:54:47 adpetlin systemd[1]: Starting vsftpd.service - Vsftpd ftp daemon...
Oct 03 09:54:47 adpetlin systemd[1]: Started vsftpd.service - Vsftpd ftp daemon.
root@adpetlin:~#
```

Рисунок 5

Затем удаляем службу из автозапуска и снова проверяем статус.

5.6 Ход работы

Выводим на экран символические ссылки, ответственные за запуск различных сервисов. Ссылка на `vsftpd.service` отсутствует.

```
root@adpetlin:~# ls /etc/systemd/system/multi-user.target.wants
atd.service      cups.path        mcelog.service  rsyslog.service  vboxadd-service.service
auditd.service   cups.service     mdmonitor.service  smartd.service   vmtoolsd.service
audit-rules.service  firewallld.service  ModemManager.service  sshd.service
avahi-daemon.service  irqbalance.service  NetworkManager.service  sssd.service
chronyd.service      kdump.service      remote-cryptsetup.target  tuned.service
crond.service        libstoragemgmt.service  remote-fs.target  vboxadd.service

root@adpetlin:~# systemctl enable vsftpd
Created symlink '/etc/systemd/system/multi-user.target.wants/vsftpd.service' → '/usr/lib/systemd/system/vsftpd.service'.

root@adpetlin:~# ls /etc/systemd/system/multi-user.target.wants
atd.service      cups.path        mcelog.service  rsyslog.service  vboxadd-service.service
auditd.service   cups.service     mdmonitor.service  smartd.service   vmtoolsd.service
audit-rules.service  firewallld.service  ModemManager.service  sshd.service
avahi-daemon.service  irqbalance.service  NetworkManager.service  sssd.service
chronyd.service      kdump.service      remote-cryptsetup.target  tuned.service
crond.service        libstoragemgmt.service  remote-fs.target  vboxadd.service
```

Рисунок 6

5.7 Ход работы

Снова добавляем службу в автозапуск и выводим на экран символические ссылки. Создается символическая ссылка для файла службы в соответствующем каталоге. Проверяем статус службы и наблюдаем изменение состояния юнита. Выводим на экран список зависимостей юнита.

```
root@adpetlin:~# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-10-03 09:54:47 MSK; 3min 0s ago
  Invocation: 1ef9fc7f179146448ed58ae26982cffe
    Main PID: 8485 (vsftpd)
      Tasks: 1 (limit: 36001)
     Memory: 756K (peak: 1.2M)
        CPU: 6ms
      CGroup: /system.slice/vsftpd.service
             └─8485 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Oct 03 09:54:47 adpetlin systemd[1]: Starting vsftpd.service - Vsftpd ftp daemon...
Oct 03 09:54:47 adpetlin systemd[1]: Started vsftpd.service - Vsftpd ftp daemon.
root@adpetlin:~# systemctl list-dependencies
default.target
├─accounts-daemon.service
├─gdm.service
├─nvmefc-boot-connections.service
├─rtkit-daemon.service
├─switcheroo-control.service
├─systemd-update-utmp-runlevel.service
├─tuned-ppd.service
├─udisks2.service
├─upower.service
├─multi-user.target
│   ├──atd.service
│   ├──audit-rules.service
│   ├──auditd.service
│   └─avahi-daemon.service
```

Рисунок 7

```
root@adpetlin:~# systemctl list-dependencies vsftpd --reverse
vsftpd.service
● └─multi-user.target
●   └─graphical.target
root@adpetlin:~#
```

Рисунок 8

Выводим на экран список юнитов, которые зависят от данного юнита.

```
Upgraded:
  iptables-libs-1.8.11-9.el10_0.x86_64                ipt
Installed:
  iptables-devel-1.8.11-9.el10_0.x86_64                iptable
  iptables-utils-1.8.11-9.el10_0.x86_64

Complete!
root@adpetlin:~#
```

Рисунок 9

Получаем полномочия администратора и устанавливаем iptables.

Проверяем статус служб firewalld и iptables.

```
root@adpetlin:~# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; preset: enabled)
   Active: active (running) since Fri 2025-10-03 09:07:50 MSK; 52min ago
     Invocation: 592b7719eba44ba0b9270fc2ec0b45cb
       Docs: man:firewalld(1)
    Main PID: 1225 (firewalld)
       Tasks: 2 (limit: 36001)
      Memory: 48.3M (peak: 50.7M)
         CPU: 346ms
    CGroup: /system.slice/firewalld.service
            └─1225 /usr/bin/python3 -sP /usr/sbin/firewalld --nofork --nopid

Oct 03 09:07:50 adpetlin systemd[1]: Starting firewalld.service - firewalld - dynamic firewall daemon...
Oct 03 09:07:50 adpetlin systemd[1]: Started firewalld.service - firewalld - dynamic firewall daemon.
root@adpetlin:~# systemctl status iptables
○ iptables.service - IPv4 firewall with iptables
   Loaded: loaded (/usr/lib/systemd/system/iptables.service; disabled; preset: disabled)
   Active: inactive (dead)

root@adpetlin:~#
```

Рисунок 10

5.11 Ход работы

Пытаемся запустить обе службы и наблюдаем, что при запуске одной службы вторая деактивируется или не запускается.

```
root@adpetlin:~# systemctl start firewalld
root@adpetlin:~# systemctl start iptables
root@adpetlin:~# systemctl status iptables
● iptables.service - IPv4 firewall with iptables
   Loaded: loaded (/usr/lib/systemd/system/iptables.service; disabled; preset: disabled)
   Active: active (exited) since Fri 2025-10-03 10:01:03 MSK; 5s ago
  Invocation: 7d34355652c4484d97d5ae6f0250b412
     Process: 9582 ExecStart=/usr/libexec/iptables/iptables.init start (code=exited, status=0/SUCCESS)
    Main PID: 9582 (code=exited, status=0/SUCCESS)
   Mem peak: 1.8M
      CPU: 11ms

Oct 03 10:01:03 adpetlin systemd[1]: Starting iptables.service - IPv4 firewall with iptables...
Oct 03 10:01:03 adpetlin iptables.init[9582]: iptables: Applying firewall rules: [ OK ]
Oct 03 10:01:03 adpetlin systemd[1]: Finished iptables.service - IPv4 firewall with iptables.
root@adpetlin:~# systemctl status firewalld
○ firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; preset: enabled)
   Active: inactive (dead) since Fri 2025-10-03 10:01:03 MSK; 11s ago
 Duration: 14.492s
  Invocation: 6eeaf534ab2a4ba58072f3b9f6a94e7e
       Docs: man:firewalld(1)
     Process: 9484 ExecStart=/usr/sbin/firewalld --nofork --nopid $FIREWALLD_ARGS (code=exited, status=0/SUCCESS)
    Main PID: 9484 (code=exited, status=0/SUCCESS)
   Mem peak: 27.3M
      CPU: 261ms

Oct 03 10:00:49 adpetlin systemd[1]: Starting firewalld.service - firewalld - dynamic firewall daemon...
Oct 03 10:00:49 adpetlin systemd[1]: Started firewalld.service - firewalld - dynamic firewall daemon.
Oct 03 10:01:03 adpetlin systemd[1]: Stopping firewalld.service - firewalld - dynamic firewall daemon...
Oct 03 10:01:03 adpetlin systemd[1]: firewalld.service: Deactivated successfully.
Oct 03 10:01:03 adpetlin systemd[1]: Stopped firewalld.service - firewalld - dynamic firewall daemon.
root@adpetlin:~#
```

Рисунок 11

5.12 Ход работы

Изучаем файл юнита firewalld и
смотрим настройки конфликтов для
этого юнита при их наличии.

```
root@adpetlin:~# 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 iptables.service ebtables.service ipset.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
DevicePolicy=closed
KeyringMode=private
LockPersonality=yes
MemoryDenyWriteExecute=yes
PrivateDevices=yes
ProtectClock=yes
ProtectControlGroups=yes
ProtectHome=yes
ProtectHostname=yes
ProtectKernelLogs=yes
ProtectKernelModules=no
ProtectKernelTunables=no
ProtectSystem=yes
RestrictRealtime=yes
RestrictSUIDSGID=yes
SystemCallArchitectures=native

[Install]
WantedBy=multi-user.target
Alias=dbus-org.fedoraproject.FirewallD1.service
root@adpetlin:~#
```

Изучаем файл юнита iptables и настроек конфликтов для этого юнита нет.

```
root@adpetlin:~# 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@adpetlin:~# █
```

Рисунок 13

5.14 Ход работы

Останавливаем службу iptables и запускаем службу firewalld. Блокируем запуск iptables. Проверяем создание символической ссылки, которая делает невозможным случайный запуск сервиса. Пытаемся запустить iptables и ожидаем сообщение об ошибке из-за маскировки службы. Пытаемся добавить iptables в автозапуск и наблюдаем, что сервис остаётся неактивным, а статус загрузки отображается как замаскированный.

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

Рисунок 14

5.15 Ход работы

Получаем полномочия администратора. Переходим в каталог systemd и находим список всех целей, которые можно изолировать.

```
root@adpetlin:~# cd /usr/lib/systemd/system
root@adpetlin:/usr/lib/systemd/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
soft-reboot.target:AllowIsolate=yes
system-update.target:AllowIsolate=yes
root@adpetlin:/usr/lib/systemd/system#
```

5.16 Ход работы

```
adpetlin@adpetlin:~$ su -  
Password:  
Last login: Fri Oct 3 10:12:42 MSK 2025 on pts/0  
root@adpetlin:~# systemctl get-default  
graphical.target  
root@adpetlin:~# 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@adpetlin:~#
```

Рисунок 16

Переключаем операционную систему в режим восстановления. Для входа в систему вводим пароль root на консоли сервера.

Перезапускаем операционную систему, изолируя соответствующую цель.

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

Give root password for maintenance
(or press Control-D to continue):
root@adpetlin:~# systemctl isolate reboot.target
```

Рисунок 17

```
Rocky Linux 10.0 (Red Quartz)
Kernel 6.12.0-55.32.1.el10_0.x86_64 on x86_64

Web console: https://adpetlin:9090/ or https://10.0.2.15:9090/

adpetlin login: adpetlin
Password:
Last login: Fri Oct 3 10:19:37 on tty2
adpetlin@adpetlin:~$
```

Рисунок 18

Получаем полномочия администратора.

```
root@adpetlin:~# 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@adpetlin:~# _
```

Рисунок 19

Устанавливаем цель по умолчанию для запуска в текстовом режиме. Перезагружаем систему и убеждаемся, что система загрузилась в текстовом режиме.

5.20 Ход работы

Устанавливаем цель по умолчанию для запуска в графическом режиме. Снова перезагружаем систему и убеждаемся, что система загрузилась в графическом режиме.

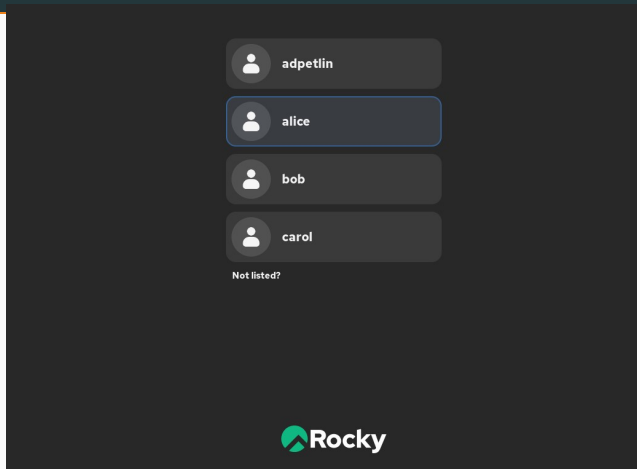


Рисунок 20

6. Выводы

Мы получили навыки управления системными службами операционной системы посредством systemd.

Список литературы

1. Поттеринг Л. Systemd для администраторов: цикл статей. — 2010. — URL: <http://wiki.opennet.ru/Systemd>.
2. Neil N. J. Learning CentOS: A Beginners Guide to Learning Linux. — CreateSpace Independent Publishing Platform, 2016.
3. Systemd. — 2022. — URL: <https://wiki.archlinux.org/title/Systemd>.