

Управление системными службами

Часть 1

Славинский В.В.

4 октября 2025

Российский университет дружбы народов, Москва, Россия Россия

Информация

..... {.columns align=center} ::: {.column width="70%"}

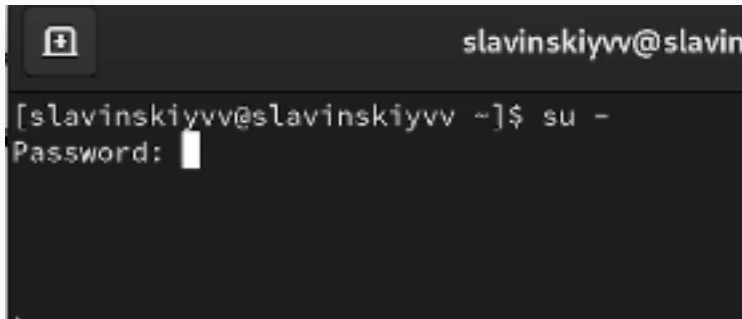
- Славинский Владислав Вадимович
- Студент
- Российский университет дружбы народов
- [1132246169@pfur.ru]

::: {.column width="30%"}

Вводная часть

Переход в режим суперпользователя

В консоли перейдем в режим работы суперпользователя, используя команду `su -`.

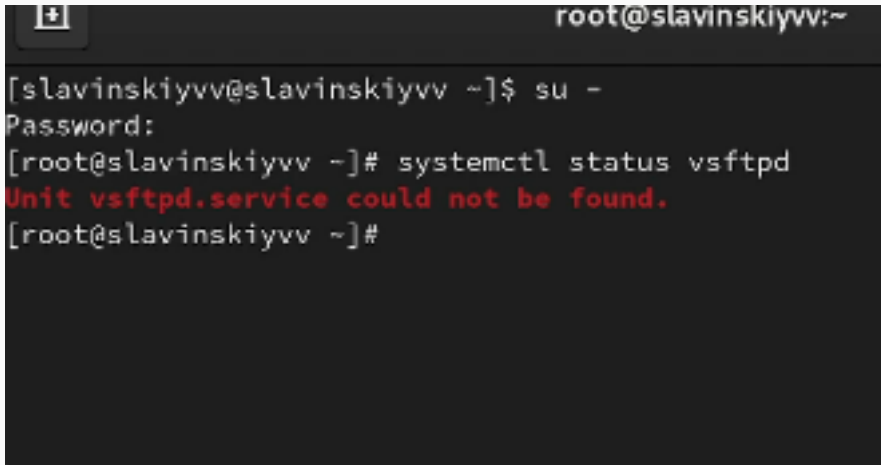
A terminal window with a dark background. The title bar shows a window icon and the text 'slavinskiyv@slavin'. The terminal content shows the prompt '[slavinskiyv@slavinskiyv ~]\$' followed by the command 'su -'. Below the command, the text 'Password:' is displayed with a white cursor character (a vertical bar) immediately following it.

```
slavinskiyv@slavin  
[slavinskiyv@slavinskiyv ~]$ su -  
Password: |
```

Рис. 1: sc1

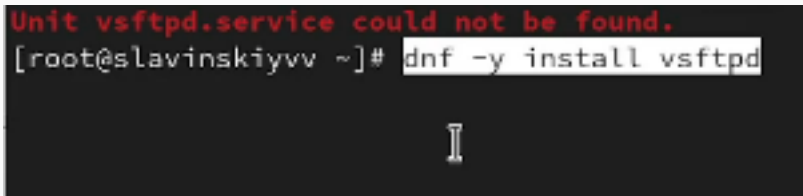
Проверка статуса службы

Проверим статус службы Very Secure FTP с помощью команды `systemctl status vsftpd`. Служба отключена, так как она не установлена



```
root@slavinskiyv:~  
[slavinskiyv@slavinskiyv ~]$ su -  
Password:  
[root@slavinskiyv ~]# systemctl status vsftpd  
Unit vsftpd.service could not be found.  
[root@slavinskiyv ~]#
```

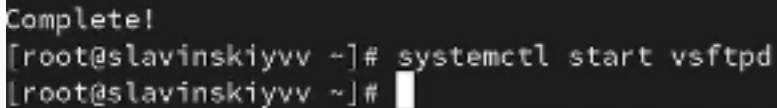
Установим службу Very Secure FTP: `dnf -y install vsftpd`.

A terminal window with a black background. The first line shows a red error message: "Unit vsftpd.service could not be found." The second line shows a command being entered: "[root@slavinskiyv ~]# dnf -y install vsftpd". A white cursor is visible at the end of the command.

```
Unit vsftpd.service could not be found.  
[root@slavinskiyv ~]# dnf -y install vsftpd
```

Рис. 3: sc3

Запустим службу Very Secure FTP: `systemctl start vsftpd`.

A terminal window with a black background and white text. The first line shows 'Complete!'. The second line shows the command '[root@slavinskiyvv ~]# systemctl start vsftpd'. The third line shows the prompt '[root@slavinskiyvv ~]#' followed by a white cursor block.

```
Complete!  
[root@slavinskiyvv ~]# systemctl start vsftpd  
[root@slavinskiyvv ~]#
```

Рис. 4: sc4

Проверка статуса службы

Проверим статус службы Very Secure FTP с помощью команды `systemctl status vsftpd`.

```
Installed:
  vsftpd-3.0.5-6.el9.x86_64

Complete!
[root@slavinskiyvv ~]# systemctl start vsftpd
[root@slavinskiyvv ~]# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled; preset: >
   Active: active (running) since Fri 2025-10-03 18:48:17 MSK; 8s ago
   Process: 7240 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exit>
  Main PID: 7241 (vsftpd)
     Tasks: 1 (limit: 48747)
    Memory: 740.0K
       CPU: 2ms
    CGroup: /system.slice/vsftpd.service
            └─7241 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Oct 03 18:48:17 slavinskiyvv systemd[1]: Starting Vsftpd ftp daemon...
Oct 03 18:48:17 slavinskiyvv systemd[1]: Started Vsftpd ftp daemon.
lines 1-13/13 (END)
```

Добавление в автозапуск

У нас служба работает, но у нас она не будет работать при автоматическом запуске операционной системы, давайте её добавим в автоматический запуск с помощью команды `systemctl enable vsftpd`. И как видим, служба добавилась в автозапуск.

```
[1]+  Stopped                                systemctl status vsftpd
[root@slavinskiyv ~]# systemctl enable vsftpd
Created symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service → /usr/lib/systemd/system/vsftpd.service.
[root@slavinskiyv ~]# 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 18:48:17 MSK; 1min 36s ago
 Main PID: 7241 (vsftpd)
    Tasks: 1 (limit: 48747)
   Memory: 740.0K
      CPU: 2ms
   CGroup: /system.slice/vsftpd.service
           └─7241 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf
```

Отключение службы

Теперь удалим службу из автозапуска через команду `systemctl disable vsftpd`. Теперь служба удалилась из автозапуска.

```
Tasks: 1 (limit: 48747)
Memory: 740.0K
CPU: 2ms
CGroup: /system.slice/vsftpd.service
└─7241 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Oct 03 18:48:17 slavinskiyvv systemd[1]: Starting Vsftpd ftp daemon...
Oct 03 18:48:17 slavinskiyvv systemd[1]: Started Vsftpd ftp daemon.
[root@slavinskiyvv ~]# systemctl disable vsftpd
Removed "/etc/systemd/system/multi-user.target.wants/vsftpd.service".
[root@slavinskiyvv ~]# 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 18:48:17 MSK; 2min 3s ago
 Main PID: 7241 (vsftpd)
    Tasks: 1 (limit: 48747)
 Memory: 740.0K
    CPU: 2ms
   CGroup: /system.slice/vsftpd.service
```

Выведем на экран символические ссылки, ответственные за запуск различных сервисов: `ls /etc/systemd/system/multi-user.target.wants`. В данном случае мы не видим `vsftpd.service`.

```
Oct 03 18:48:17 slavinskiyvv systemd[1]: Started Vsftpd ftp daemon.
[root@slavinskiyvv ~]# ls /etc/systemd/system/multi-user.target.wants
avahi-daemon.service  cups.path          mcelog.service      rsyslog.service
auditd.service        cups.service       mdmonitor.service    smartd.service
avahi-daemon.service  irqbalance.service ModemManager.service sshd.service
chronyd.service       kdump.service     NetworkManager.service sssd.service
crond.service         libstoragemgmt.service remote-fs.target      tuned.service
[root@slavinskiyvv ~]#
```

Рис. 8: sc8

Вывод символических ссылок после добавления vsftp в автозапуск

Теперь снова добавим vsftp в автозапуск и проверим, появился ли vsftpd.service. Как видим, у нас vsftpd появился.

```
lavinskiyvv ~]# ls /etc/systemd/system/multi-user.target.wants/ | grep vsftpd.service
vsftpd.service
```

vice	cups.path	mcelog.service	rsyslog.service	vboxadd.service
service	cups.service	mdmonitor.service	smartd.service	vboxadd-service.service
aemon.service	irqbalance.service	ModemManager.service	sshd.service	vmtoolsd.service
.service	kdump.service	NetworkManager.service	sssd.service	vsftpd.service
service	libstoragemgmt.service	remote-fs.target	tuned.service	

```
lavinskiyvv ~]#
```

Рис. 9: sc9

Проверка статуса службы

Снова проверим статус службы Very Secure FTP. У нас служба будет включена после перезапуска системы.

```
[root@slavinskiyv ~]# 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 18:48:17 MSK; 3min 31s ago
     Main PID: 7241 (vsftpd)
       Tasks: 1 (limit: 48747)
      Memory: 740.0K
         CPU: 2ms
       CGroup: /system.slice/vsftpd.service
              └─7241 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Oct 03 18:48:17 slavinskiyv systemd[1]: Starting Vsftpd ftp daemon...
Oct 03 18:48:17 slavinskiyv systemd[1]: Started Vsftpd ftp daemon.
[root@slavinskiyv ~]#
```

Рис. 10: sc10

Вывод списка зависимостей юнита

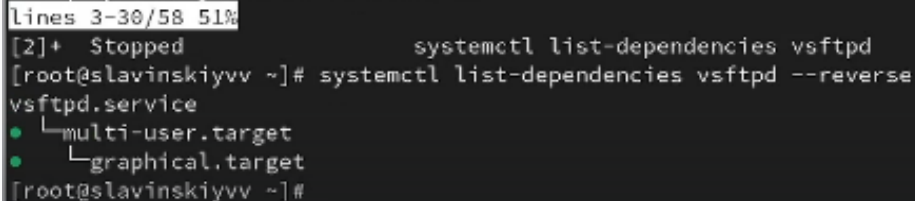
Выведем на экран список зависимостей юнита: `systemctl list-dependencies vsftpd`.

```
Oct 03 18:48:17 slavinskiyvv systemd[1]: Starting Vsftpd ftp daemon...
Oct 03 18:48:17 slavinskiyvv systemd[1]: Started Vsftpd ftp daemon.
[root@slavinskiyvv ~]# 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
```

Рис. 11: sc11

Вывод списка юнитов, которые зависят от данного типа

Выведем на экран список юнитов, которые зависят от данного юнита: `systemctl list-dependencies vsftpd -reverse`.



```
lines 3-30/58 51%
[2]+  Stopped                  systemctl list-dependencies vsftpd
[root@slavinskiyv ~]# systemctl list-dependencies vsftpd --reverse
vsftpd.service
● └─multi-user.target
●   └─graphical.target
[root@slavinskiyv ~]#
```

Рис. 12: sc12

Установка iptables

Дальше установим iptables: `dnf -y install iptables*`.

```
[root@slavinskiyvv ~]# dnf -y install iptables\*
Last metadata expiration check: 1:25:26 ago on Fri 03 Oct 2025 05:27:59 |
Dependencies resolved.
=====
Package                                Architecture      Version
=====
Installing:
  iptables-devel                        x86_64            1.8.10-11.el9_5
  iptables-legacy                      x86_64            1.8.10-11.1.el9
  iptables-legacy-devel                x86_64            1.8.10-11.1.el9
  iptables-legacy-libs                 x86_64            1.8.10-11.1.el9
  iptables-libs                        x86_64            1.8.10-11.el9_5
  iptables-nft                         x86_64            1.8.10-11.el9_5
  iptables-nft-services                noarch            1.8.10-11.el9_5
  iptables-services                   noarch            1.8.10-11.1.el9
  iptables-utils                       x86_64            1.8.10-11.el9_5
Installing dependencies:
  libnetfilter_conntrack               x86_64            1.0.9-1.el9
  libnfnetlink                         x86_64            1.0.1-23.el9_5
  libnftnl                             x86_64            1.2.6-4.el9_4
Transaction Summary
=====
Install 12 Packages

Total download size: 952 k
```

Проверка статуса firewalld

Проверим статус firewalld: `systemctl status firewalld`.

```
[root@slavinskiyvv ~]# 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 18:58:44 MSK; 3min 15s ago
     Docs: man:firewalld(1)
  Main PID: 8532 (firewalld)
    Tasks: 2 (limit: 48747)
   Memory: 26.4M
      CPU: 318ms
   CGroup: /system.slice/firewalld.service
           └─8532 /usr/bin/python3 -s /usr/sbin/firewalld --nofork --nopid

Oct 03 18:58:44 slavinskiyvv systemd[1]: Starting firewalld - dynamic firewall daemon...
Oct 03 18:58:44 slavinskiyvv systemd[1]: Started firewalld - dynamic firewall daemon.
[root@slavinskiyvv ~]#
```

Рис. 14: sc14

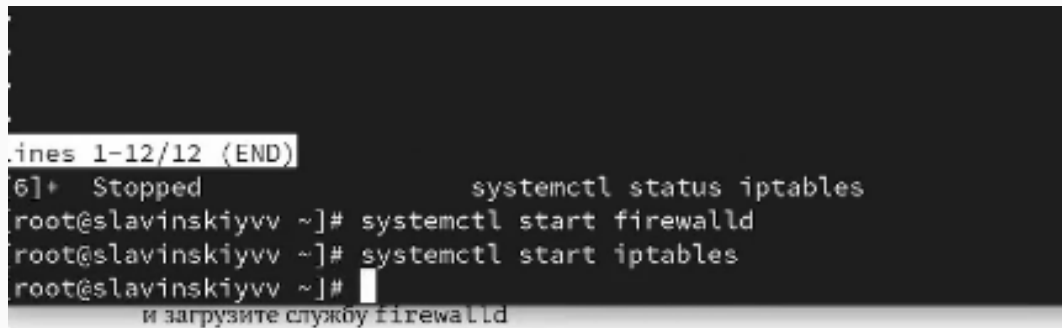
Проверим статус iptables: `systemctl status iptables`. Здесь мы видим, что служба неактивна, и не запущена а автозапуске.

```
└─8532 /usr/bin/python3 -s /usr/sbin/firewalld --nofork --nopid
Oct 03 18:58:44 slavinskiyvv systemd[1]: Starting firewalld - dynamic firewall daemon...
Oct 03 18:58:44 slavinskiyvv systemd[1]: Started firewalld - dynamic firewall daemon.
[root@slavinskiyvv ~]# systemctl status iptables
○ iptables.service - IPv4 firewall with iptables
   Loaded: loaded (/usr/lib/systemd/system/iptables.service; disabled; preset: disabled)
   Active: inactive (dead)
```

Рис. 15: sc15

Запуск firewalld и iptables

Попробуем запустить firewalld и iptables: `systemctl start firewalld,systemctl start iptables`.

A terminal window with a dark background. The prompt is root@slavinskiyv ~]. The user enters 'systemctl status iptables', followed by 'systemctl start firewalld', and then 'systemctl start iptables'. The output for the first command is partially visible as '...ines 1-12/12 (END)'. The output for the second command is 'Stopped'. The output for the third command is not yet visible. A Russian text overlay 'и загрузите службу firewalld' is at the bottom.

```
...ines 1-12/12 (END)
[6]+  Stopped                               systemctl status iptables
root@slavinskiyv ~]# systemctl start firewalld
root@slavinskiyv ~]# systemctl start iptables
root@slavinskiyv ~]#
```

и загрузите службу firewalld

Рис. 16: sc16

Посмотрим статус firewalld. firewalld у нас теперь не запущена.

```
[root@slavinskiyvv ~]# systemctl start firewalld
[root@slavinskiyvv ~]# systemctl start iptables
[root@slavinskiyvv ~]# 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 19:03:27 MSK; 8s ago
     Duration: 4min 42.877s
       Docs: man:firewalld(1)
    Process: 8532 ExecStart=/usr/sbin/firewalld --nofork --nopid $FIREWALLD_ARGS (code=exited, stat
 Main PID: 8532 (code=exited, status=0/SUCCESS)
      CPU: 352ms
```

Рис. 17: sc17

Проверка статуса iptables

Посмотрим статус iptables. И тут уже понятно, что одна служба деактивируется, а другая включается, поскольку iptables запустилась.

```
Oct 03 19:03:27 slavinskiyvv systemd[1]: Stopped firewalld - dynamic firewall daemon.  
lines 1-14/14 (END)  
[7]+ Stopped                  systemctl status firewalld  
[root@slavinskiyvv ~]# 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 19:03:27 MSK; 22s ago  
   Process: 8686 ExecStart=/usr/libexec/iptables/iptables.init start (code=exited, status=0/SUCCESS)  
   Main PID: 8686 (code=exited, status=0/SUCCESS)  
      CPU: 11ms  
  
Oct 03 19:03:27 slavinskiyvv systemd[1]: Starting IPv4 firewall with iptables...  
Oct 03 19:03:27 slavinskiyvv iptables.init[8686]: iptables: Applying firewall rules: [ OK ]  
Oct 03 19:03:27 slavinskiyvv systemd[1]: Finished IPv4 firewall with iptables.  
[root@slavinskiyvv ~]#
```

Рис. 18: sc18

Ввод команды для анализа ошибок

Введем `cat /usr/lib/systemd/system/firewalld.service`, чтобы посмотреть ошибки. И вот мы видим, с чем конфликтует служба `firewalld`.

```
Before=network-pre.target
Wants=network-pre.target
After=dbus.service
After=polkit.service
Conflicts=iptables.service ip6tables.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
# supress to log debug and error output also to /var/log/messages
StandardOutput=null
StandardError=null
Type=dbus
BusName=org.fedoraproject.FirewallD1
KillMode=mixed
```

Ввод команды для анализа ошибок

Введем то же самое, только для iptables: `cat /usr/lib/systemd/system/iptables.service`. Но тут, мы не видим никаких ошибок.

```
Alias=dbus-org.fedoraproject.FirewallD1.service
[root@slavinskiyvv ~]# 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

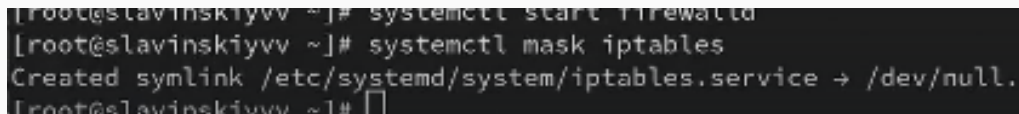
[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/usr/libexec/iptables/iptables.init start
ExecReload=/usr/libexec/iptables/iptables.init reload
ExecStop=/usr/libexec/iptables/iptables.init stop
Environment=BOOTUP=serial
Environment=CONSOLETYPE=serial
```


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`.

```
wantedby=multi-user.target  
[root@slavinskiyvv ~]# systemctl stop iptables  
[root@slavinskiyvv ~]# systemctl start firewalld  
[root@slavinskiyvv ~]#
```

Рис. 21: sc21

Заблокируем запуск iptables, введя команду `systemctl mask iptables`.



```
[root@slavinskiyvv ~]# systemctl start firewalld
[root@slavinskiyvv ~]# systemctl mask iptables
Created symlink /etc/systemd/system/iptables.service → /dev/null.
[root@slavinskiyvv ~]#
```

Рис. 22: sc22

Теперь попробуем запустить iptables. Видим , что у нас ошибка, так как мы эту службу замаскировали.

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

Рис. 23: sc23

Попробуем добавить iptables в автозапуск, но сервис будет неактивен, а статус загрузки отобразился как замаскированный.

```
[root@slavinskiyvv ~]# systemctl enable iptables
Failed to enable unit: Unit file /etc/systemd/system/iptables.service is masked.
[root@slavinskiyvv ~]#
```

Рис. 24: sc24

Список целей, которые можно изолировать

Дальше перейдем каталог `systemd` и найдите список всех целей, которые можно изолировать: `cd /usr/lib/systemd/system, grep Isolate *.target`.

```
[8]+ Stopped                               systemctl status iptables
[root@slavinskiyvv ~]# cd /usr/lib/systemd/system
[root@slavinskiyvv 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
```

Переключим операционную систему в режим восстановления: `systemctl isolate rescue.target`.

```
runlevel4.target:AllowIsolate=yes  
runlevel5.target:AllowIsolate=yes  
runlevel6.target:AllowIsolate=yes  
system-update.target:AllowIsolate=yes  
[root@slavinskiyv system]# systemctl isolate rescue.target
```

Рис. 26: sc26

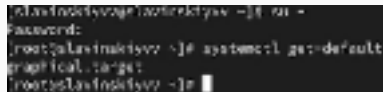
Перезапустим операционную систему следующим образом: `systemctl isolate reboot.target`.



```
systemd 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):  
root@salavskiy:~# systemctl isolate reboot.target
```

Рис. 27: sc27

Теперь вводим команду `systemctl get-default`, чтобы узнать установленную по умолчанию цель. Видим, что запускается система по умолчанию в графическом режиме.



```
[root@slaxinskiyy ~]# cat -  
Password:  
[root@slaxinskiyy ~]# systemctl get-default  
graphical.target  
[root@slaxinskiyy ~]#
```

Рис. 28: sc28

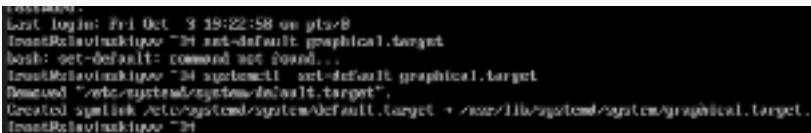
Для запуска по умолчанию текстового режима введём `systemctl set-default multi-user.target` и перезагружаем.

```
graphical.target
[root@elavirskiy ~]# systemctl set-default
See few arguments.
[root@elavirskiy ~]# 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@elavirskiy ~]#
```

Рис. 29: sc29

Возвращение на графический режим

Чтобы нам обратно вернуться в графический режим, нужно перейти на root и ввести команду `systemctl set-default graphical.target`.



```
root@kali:~#  
Last login: Fri Oct 9 19:22:58 on pts/0  
root@kali:~# systemctl set-default graphical.target  
bosh: set-default: command not found...  
root@kali:~# 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@kali:~#
```

Рис. 30: sc30

Запуск в графическом режиме

Перезагружаем и видим, мы снова в графическом режиме.

