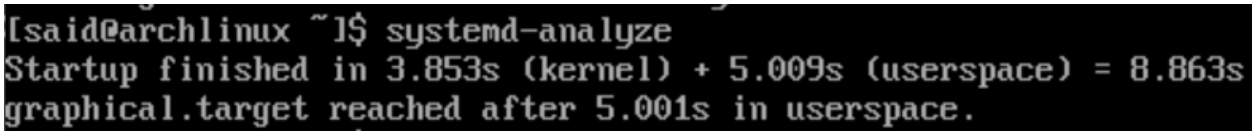
Linux ЛР4 | Гаджиев Саид M3304

Часть 1. Получение информации о времени загрузки

1. Вывод общей информации о времени загрузки.

```
systemd-analyze
```

Покажет суммарное время загрузки системы, разделённое на время ядра, загрузчика и пользователейских процессов.



2. Список сервисов в порядке уменьшения времени запуска.

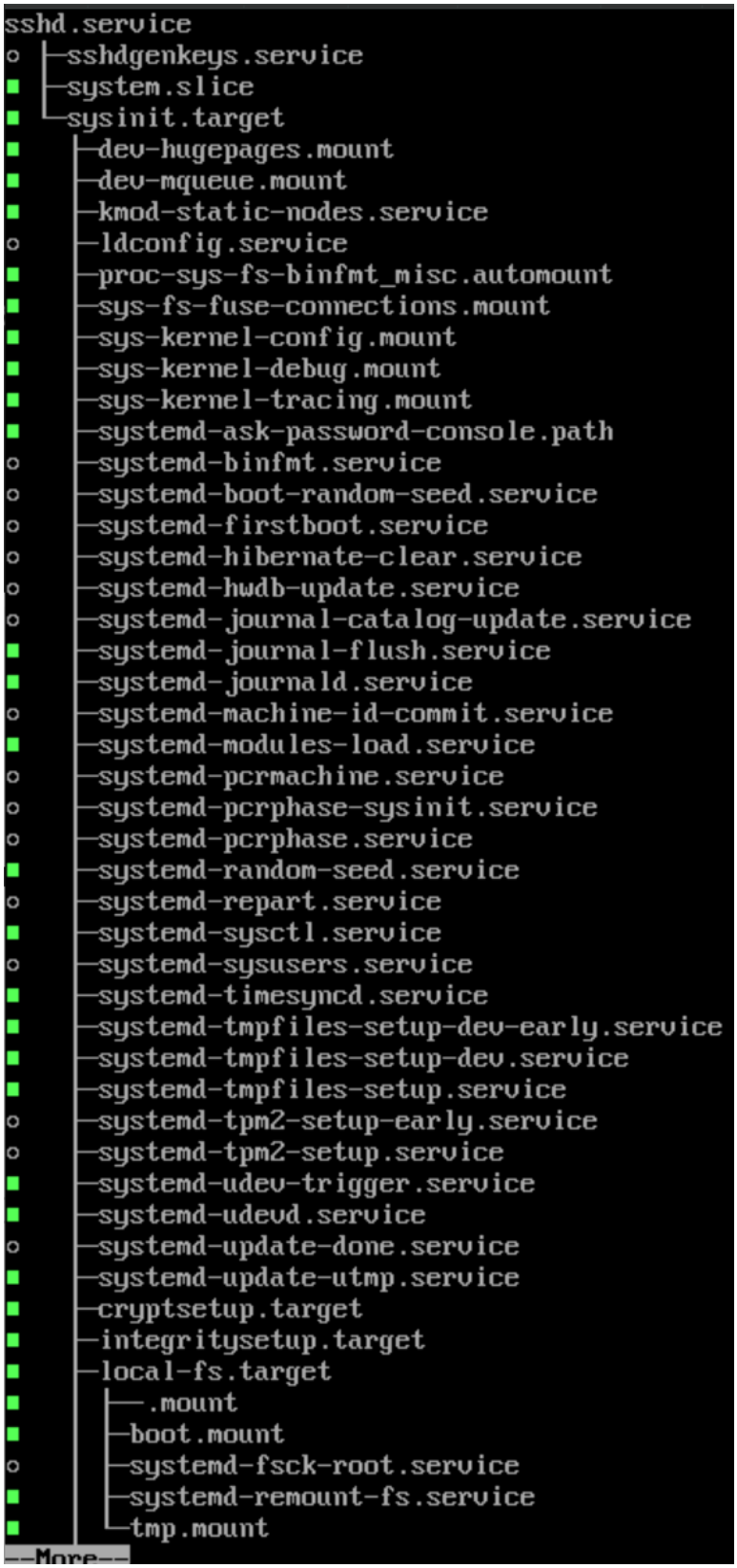
```
systemd-analyze blame
```

Увидим список юнитов-сервисов с указанием времени, затраченного на их запуск.

3. Вывод списка зависимостей для сервиса sshd.

```
systemctl list-dependencies sshd.service
```

В выводе будут показаны все юниты, от которых зависит sshd.

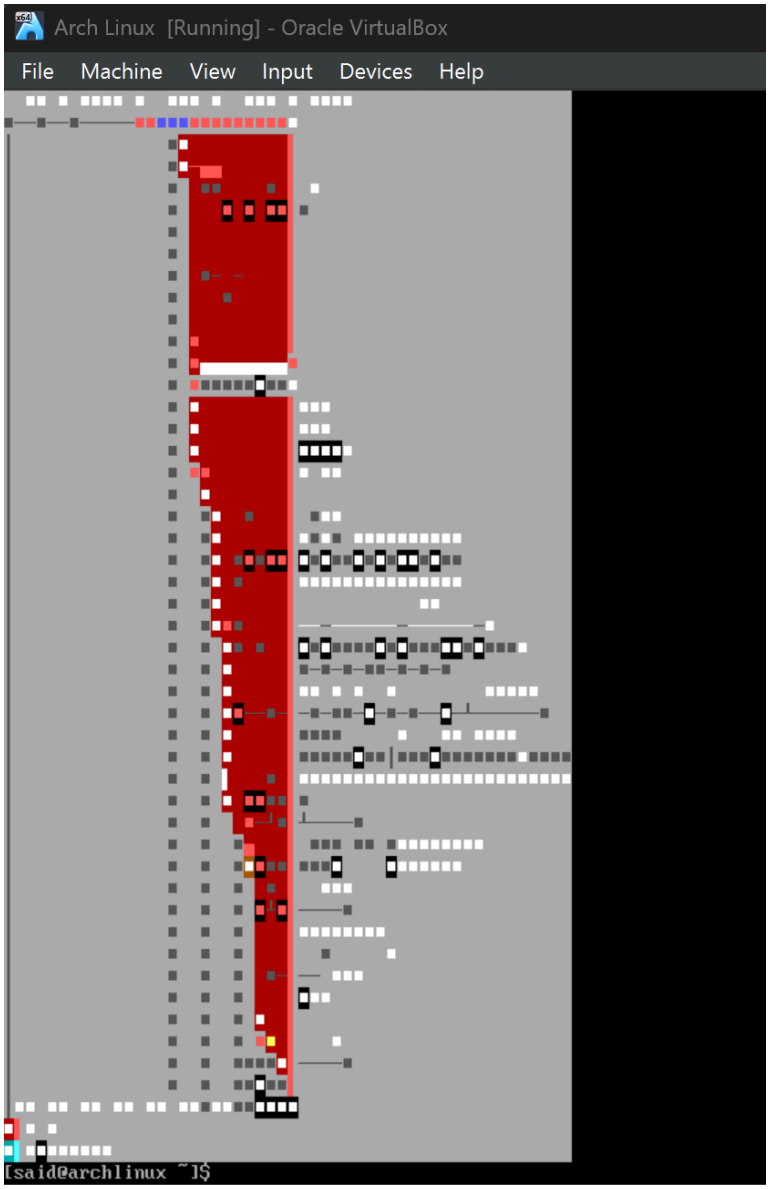


4. Построение графика загрузки в формате SVG.

```
systemd-analyze plot > boot.svg
```

Файл `boot.svg` можно открыть в браузере или графическом просмотрщике.

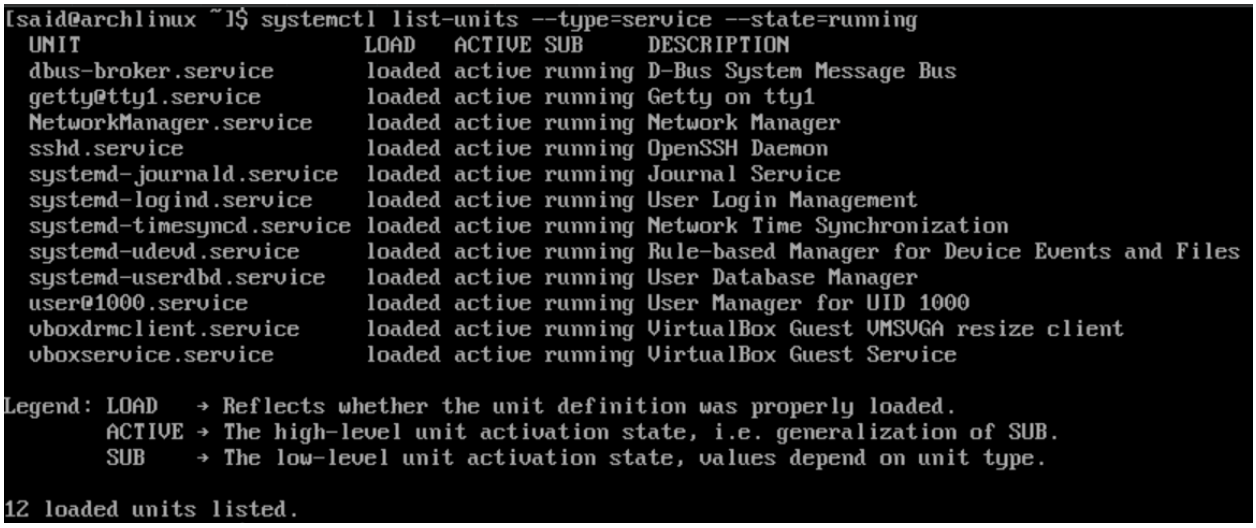
Тут я немного закостылил, ибо работал в Arch Linux с minimal template, пришлось схитрить и использовать `rsvg-convert` из пакета `librsvg`, чтобы конвертировать `.svg` в ASCII-графику. Затем вывел PNG в консоли с помощью `chafa`. Если не устроит, на сдаче сделаю общую папку в ВМ, чтобы вытянуть картинку (пометка для себя =))



Часть 2. Управление юнитами

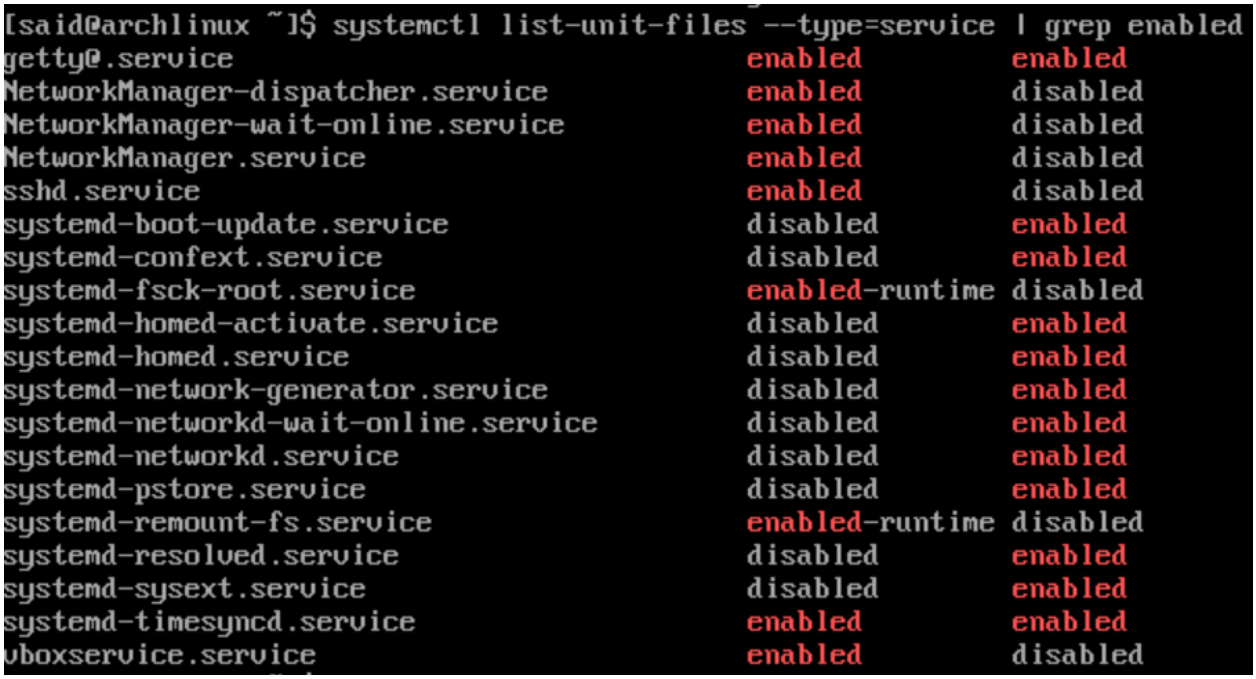
1. Получение списка запущенных юнитов сервисов.

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



2. Перечень юнитов с автозапуском.

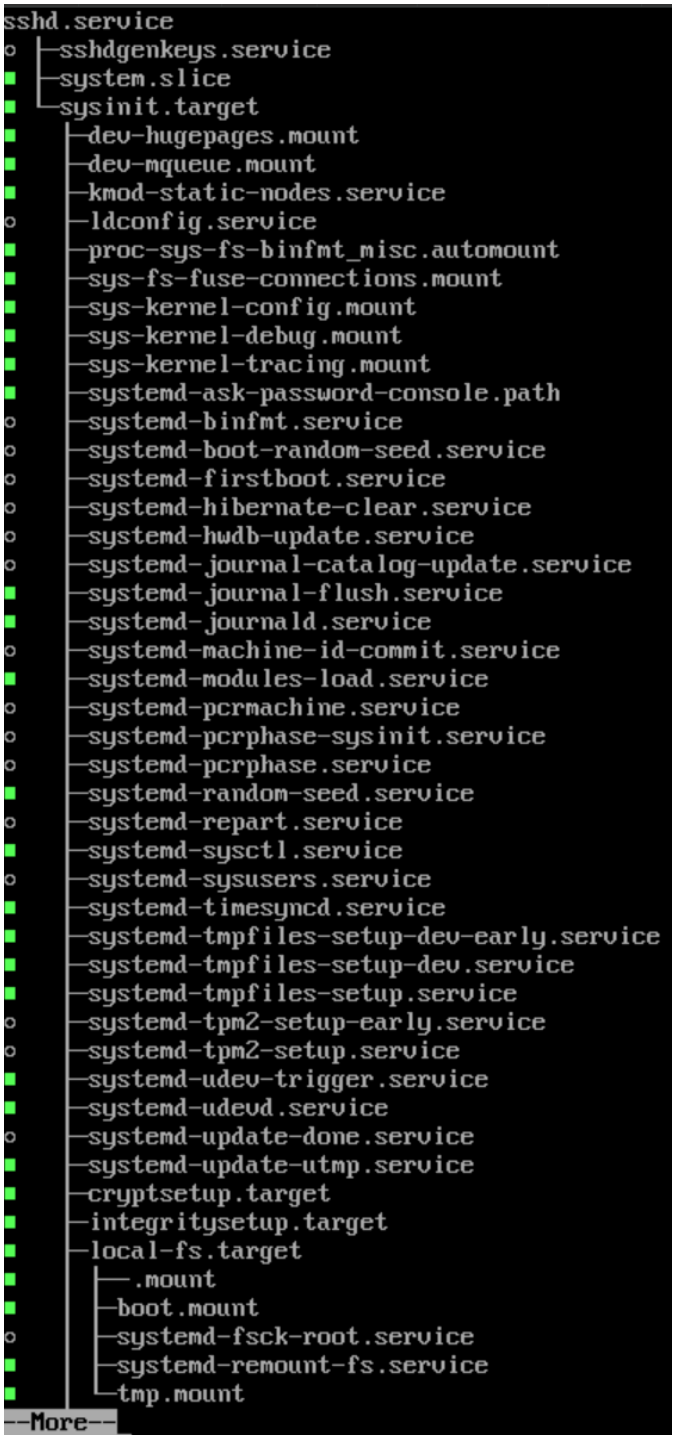
```
systemctl list-unit-files --type=service | grep enabled
```



3. Определение зависимостей сервиса sshd.

```
systemctl list-dependencies sshd.service
```

Вывод список юнитов, от которых зависит сервис sshd.

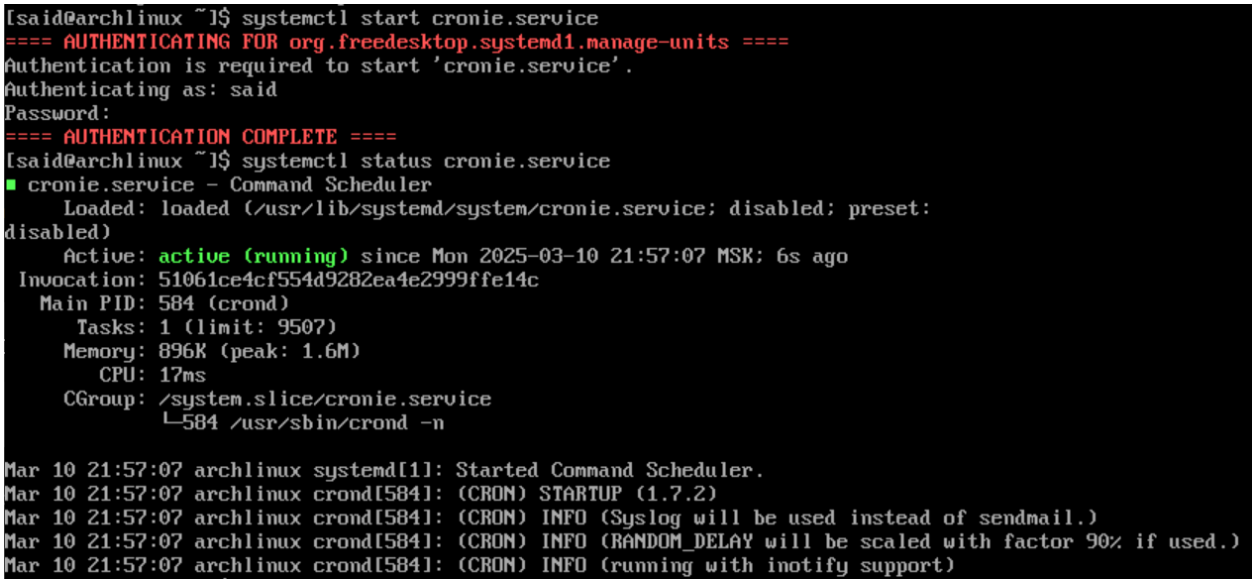


4. Проверка и запуск сервиса cron.

```
systemctl status cronie.service
```

Если сервис не запущен, его можно запустить:

```
systemctl start cronie.service
```



5. Вывод всех параметров юнита cron.

```
systemctl show cronie.service
```

```
Type=simple
ExitType=main
Restart=on-failure
RestartMode=normal
NotifyAccess=none
RestartUSec=30s
RestartSteps=0
RestartMaxDelayUSec=infinity
RestartUSecNext=30s
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
TimeoutAbortUSec=1min 30s
TimeoutStartFailureMode=terminate
TimeoutStopFailureMode=terminate
RuntimeMaxUSec=infinity
RuntimeRandomizedExtraUSec=0
WatchdogUSec=0
WatchdogTimestampMonotonic=0
RootDirectoryStartOnly=no
RemainAfterExit=no
GuessMainPID=yes
MainPID=584
ControlPID=0
FileDescriptorStoreMax=0
NFileDescriptorStore=0
FileDescriptorStorePreserve=restart
StatusErrno=0
Result=success
ReloadResult=success
CleanResult=success
LiveMountResult=success
UID=[not set]
GID=[not set]
NRestarts=0
OOMPolicy=stop
ReloadSignal=1
ExecMainStartTimestamp=Mon 2025-03-10 21:57:07 MSK
ExecMainStartTimestampMonotonic=160660162
ExecMainExitTimestampMonotonic=0
ExecMainHandoffTimestamp=Mon 2025-03-10 21:57:07 MSK
ExecMainHandoffTimestampMonotonic=160689212
ExecMainPID=584
ExecMainCode=0
ExecMainStatus=0
ExecStart={ path=/usr/sbin/crond ; argv[]=/usr/sbin/crond -n $CRONDARGS ; ignore_errors=no ; start_time=[Mon 2025-03-10 21:57:07 MSK] ; stop_time=[n/a] ; pid=58
4 ; code=(null) ; status=0/0 }
ExecStartEx={ path=/usr/sbin/crond ; argv[]=/usr/sbin/crond -n $CRONDARGS ; flags= ; start_time=[Mon 2025-03-10 21:57:07 MSK] ; stop_time=[n/a] ; pid=584 ; code
=(null) ; status=0/0 }
ExecReload={ path=/bin/kill ; argv[]=/bin/kill -URG $MAINPID ; ignore_errors=no ; start_time=[n/a] ; stop_time=[n/a] ; pid=0 ; code=(null) ; status=0/0 }
--More--
```

```
ExecStart={ path=/usr/sbin/crond ; argv[]=/usr/sbin/crond -n $CRONDARGS ; ignore_errors=no ; start_time=[Mon 2025-03-10 21:57:07 MSK] ; stop_time=[n/a] ; pid=58
4 ; code=(null) ; status=0/0 }
ExecStartEx={ path=/usr/sbin/crond ; argv[]=/usr/sbin/crond -n $CRONDARGS ; flags= ; start_time=[Mon 2025-03-10 21:57:07 MSK] ; stop_time=[n/a] ; pid=584 ; code
=(null) ; status=0/0 }
ExecReload={ path=/bin/kill ; argv[]=/bin/kill -URG $MAINPID ; ignore_errors=no ; start_time=[n/a] ; stop_time=[n/a] ; pid=0 ; code=(null) ; status=0/0 }
ExecReloadEx={ path=/bin/kill ; argv[]=/bin/kill -URG $MAINPID ; flags= ; start_time=[n/a] ; stop_time=[n/a] ; pid=0 ; code=(null) ; status=0/0 }
Slice=system.slice
ControlGroup=/system.slice/cronie.service
ControlGroupId=3102
MemoryCurrent=917504
MemoryPeak=1683456
MemorySwapCurrent=0
MemorySwapPeak=0
MemoryZSwapCurrent=0
MemoryAvailable=7939215360
EffectiveMemoryMax=8325648384
EffectiveMemoryHigh=8325648384
CPUUsageNSec=17881000
TasksCurrent=1
EffectiveTasksMax=9507
IPIngressBytes=[no data]
IPIngressPackets=[no data]
IPEgressBytes=[no data]
IPEgressPackets=[no data]
IOReadBytes=[not set]
IOReadOperations=[not set]
IOWriteBytes=[not set]
IOWriteOperations=[not set]
Delegate=no
CPUAccounting=yes
CPUWeight=[not set]
StartupCPUWeight=[not set]
CPUShares=[not set]
StartupCPUShares=[not set]
CPUQuotaPerSecUSec=infinity
CPUQuotaPeriodUSec=infinity
IOAccounting=no
IOWeight=[not set]
StartupIOWeight=[not set]
BlockIOAccounting=no
BlockIOWeight=[not set]
StartupBlockIOWeight=[not set]
MemoryAccounting=yes
DefaultMemoryLow=0
DefaultStartupMemoryLow=0
DefaultMemoryMin=0
MemoryMin=0
MemoryLow=0
StartupMemoryLow=0
--More--
```

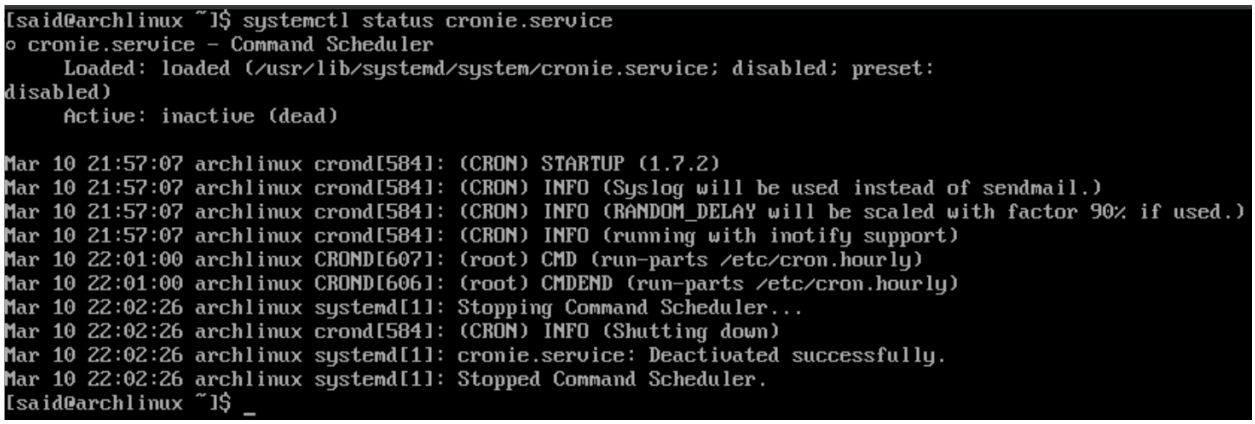
```
StartupMemoryLow=0
MemoryHigh=infinity
StartupMemoryHigh=infinity
MemoryMax=infinity
StartupMemoryMax=infinity
MemorySwapMax=infinity
StartupMemorySwapMax=infinity
Memory2SwapMax=infinity
StartupMemory2SwapMax=infinity
Memory2SwapWriteback=yes
MemoryLimit=infinity
DevicePolicy=auto
TasksAccounting=yes
TasksMax=9507
IPAccounting=no
ManagedOOMSwap=auto
ManagedOOMMemoryPressure=auto
ManagedOOMMemoryPressureLimit=0
ManagedOOMMemoryPressureDurationUSec=[not set]
ManagedOOMPreference=none
MemoryPressureWatch=auto
MemoryPressureThresholdUSec=200ms
CoredumpReceive=no
EnvironmentFiles=/etc/sysconfig/crond (ignore_errors=yes)
UMask=0022
LimitCPU=infinity
LimitCPUSoft=infinity
LimitFSIZE=infinity
LimitFSIZESoft=infinity
LimitDATA=infinity
LimitDATASoft=infinity
LimitSTACK=infinity
LimitSTACKSoft=8388608
LimitCORE=infinity
LimitCORESoft=infinity
LimitRSS=infinity
LimitRSSSoft=infinity
LimitNOFILE=524288
LimitNOFILESoft=1024
LimitAS=infinity
LimitASSoft=infinity
LimitNPROC=31691
LimitNPROCSoft=31691
LimitMEMLOCK=8388608
LimitMEMLOCKSoft=8388608
LimitLOCKS=infinity
LimitLOCKSSoft=infinity
LimitSIGPENDING=31691
LimitSIGPENDINGSoft=31691
--More--
```

```
LimitSIGPENDINGSoft=31691
LimitMSGQUEUE=819200
LimitMSGQUEUESoft=819200
LimitNICE=0
LimitNICESoft=0
LimitRTPRIO=0
LimitRTPRIOSoft=0
LimitRTTIME=infinity
LimitRTTIMESoft=infinity
RootEphemeral=no
OOMScoreAdjust=0
CoredumpFilter=0x33
Nice=0
IOSchedulingClass=2
IOSchedulingPriority=4
CPUSchedulingPolicy=0
CPUSchedulingPriority=0
CPUAffinityFromNUMA=no
NUMAPolicy=n/a
TimerSlackNSec=50000
CPUSchedulingResetOnFork=no
NonBlocking=no
StandardInput=null
StandardOutput=journal
StandardError=inherit
TTYReset=no
TTYVHangup=no
TTYVTDisallocate=no
SyslogPriority=30
SyslogLevelPrefix=yes
SyslogLevel=6
SyslogFacility=3
LogLevelMax=-1
LogRateLimitIntervalUSec=0
LogRateLimitBurst=0
SecureBits=0
CapabilityBoundingSet=cap_chown cap_dac_override cap_dac_read_search cap_fowner cap_fsetid cap_kill cap_setgid cap_setuid cap_setpcap cap_linux_immutable cap_ne
t_bind_service cap_net_broadcast cap_net_admin cap_net_raw cap_ipc_lock cap_ipc_owner cap_sys_module cap_sys_rawio cap_sys_chroot cap_sys_ptrace cap_sys_pacct c
ap_sys_admin cap_sys_boot cap_sys_nice cap_sys_resource cap_sys_time cap_sys_tty_config cap_mknod cap_lease cap_audit_write cap_audit_control cap_setfcap cap_na
c_override cap_mac_admin cap_syslog cap_wake_alarm cap_block_suspend cap_audit_read cap_perfmon cap_bpf cap_checkpoint_restore
DynamicUser=no
SetLoginEnvironment=no
RemoveIPC=no
PrivateTmp=no
PrivateTmpEx=no
PrivateDevices=no
ProtectClock=no
ProtectKernelTunables=no
ProtectKernelModules=no
--More--
```

Дальше не вижу смысле делать скрины, слишком много инфы.

6. Отключение автозапуска для cron.

```
systemctl stop cronie.service
systemctl disable cronie.service
```



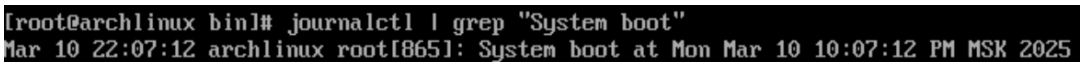
Часть 3. Создание собственного сервиса mymsg

1. Подготовка скрипта

Создадим исполняемый скрипт, который при запуске запишет в системный журнал текущие дату и время. Например в `/usr/local/bin/mymsg.sh`:

```
#!/bin/bash
/usr/bin/logger "System boot at $(date)"
```

И исполняемым его сделаем:
`chmod +x /usr/local/bin/mymsg.sh`



2. Создание файла юнита

Файл `/etc/systemd/system/mymsg.service`:

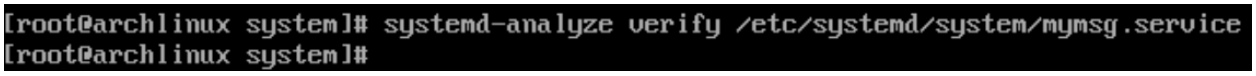
```
[Unit]
Description=My Message Service: writes date and time to the system log
After=network-online.target
Wants=network-online.target

[Service]
Type=oneshot
ExecStart=/usr/local/bin/mymsg.sh
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

Пояснения:

- **After=network-online.target** и **Wants=network-online.target** – гарантируют, что сервис запустится только после того, как сеть будет доступна.
- **Type=oneshot** используется для разового выполнения команды.
- **RemainAfterExit=yes** позволяет считать сервис запущенным даже после завершения скрипта.

Проверка корректности файла:

```
systemd-analyze verify /etc/systemd/system/mymsg.service
```



3. Автоматизация и запуск

```
systemctl enable mymsg.service

systemctl start mymsg.service
```

```
[root@archlinux system]# systemctl enable mymsg.service
Created symlink '/etc/systemd/system/multi-user.target.wants/mymsg.service' → '/etc/systemd/system/mymsg.service'.
[root@archlinux system]# systemctl start mymsg.service
[root@archlinux system]#
[root@archlinux system]# systemctl status mymsg.service
● mymsg.service - My Message Service: writes date and time to the system log
     Loaded: loaded (/etc/systemd/system/mymsg.service; enabled; preset: disabled)
     Active: active (exited) since Mon 2025-03-10 22:14:44 MSK; 4s ago
 Invocation: b971a125b21f4de99002eeda861a5e38
    Process: 955 ExecStart=/usr/local/bin/mymsg.sh (code=exited, status=0/SUCCESS)
   Main PID: 955 (code=exited, status=0/SUCCESS)
   Mem peak: 1.6M
        CPU: 26ms

Mar 10 22:14:44 archlinux systemd[1]: Starting My Message Service: writes date and time to the system log...
Mar 10 22:14:44 archlinux root[958]: System boot at Mon Mar 10 10:14:44 PM MSK 2025
Mar 10 22:14:44 archlinux systemd[1]: Finished My Message Service: writes date and time to the system log.
[root@archlinux system]#
```

Часть 4. Работа с системным журналом

1. Вывод всего системного журнала.

`journalctl`

```
Mar 10 22:02:32 archlinux sudo[792]: pam_unix(sudo:session): session opened for user root(uid=0) by said(uid=1000)
Mar 10 22:02:32 archlinux systemd[1]: Reload requested from client PID 795 ('systemctl') (unit session-1.scope)...
Mar 10 22:02:32 archlinux systemd[1]: Reloading...
Mar 10 22:02:33 archlinux systemd[1]: Reloading finished in 282 ms.
Mar 10 22:02:33 archlinux sudo[792]: pam_unix(sudo:session): session closed for user root
Mar 10 22:04:37 archlinux NetworkManager[381]: <info>  [1741633477.3847] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:04:37 archlinux NetworkManager[381]: <info>  [1741633477.4111] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:04:37 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:04:37 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:04:47 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:05:07 archlinux sudo[854]:     said : TTY=tty1 ; PWD=/usr/local/bin ; USER=root ; COMMAND=/usr/bin/su
Mar 10 22:05:07 archlinux sudo[854]: pam_unix(sudo:session): session opened for user root(uid=0) by said(uid=1000)
Mar 10 22:05:07 archlinux su[857]: (to root) root on pts/0
Mar 10 22:05:07 archlinux su[857]: pam_unix(su:session): session opened for user root(uid=0) by said(uid=0)
Mar 10 22:07:12 archlinux root[865]: System boot at Mon Mar 10 10:07:12 PM MSK 2025
Mar 10 22:09:36 archlinux systemd[1]: Starting Cleanup of Temporary Directories...
Mar 10 22:09:36 archlinux systemd[1]: systemd-tmpfiles-clean.service: Deactivated successfully.
Mar 10 22:09:36 archlinux systemd[1]: Finished Cleanup of Temporary Directories.
Mar 10 22:09:38 archlinux NetworkManager[381]: <info>  [1741633778.0115] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:09:38 archlinux NetworkManager[381]: <info>  [1741633778.0308] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:09:38 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:09:38 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:09:48 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:14:37 archlinux systemd[1]: Reload requested from client PID 903 ('systemctl') (unit session-1.scope)...
Mar 10 22:14:37 archlinux systemd[1]: Reloading...
Mar 10 22:14:37 archlinux systemd[1]: Reloading finished in 277 ms.
Mar 10 22:14:38 archlinux NetworkManager[381]: <info>  [1741634078.0175] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:14:38 archlinux NetworkManager[381]: <info>  [1741634078.0439] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:14:38 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:14:38 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:14:43 archlinux systemd[1]: Starting Network Manager Wait Online...
Mar 10 22:14:44 archlinux systemd[1]: Finished Network Manager Wait Online.
Mar 10 22:14:44 archlinux systemd[1]: Reached target Network is Online.
Mar 10 22:14:44 archlinux systemd[1]: Starting My Message Service: writes date and time to the system log...
Mar 10 22:14:44 archlinux root[958]: System boot at Mon Mar 10 10:14:44 PM MSK 2025
Mar 10 22:14:44 archlinux systemd[1]: Finished My Message Service: writes date and time to the system log.
Mar 10 22:14:48 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:19:38 archlinux NetworkManager[381]: <info>  [1741634378.0300] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:19:38 archlinux NetworkManager[381]: <info>  [1741634378.0442] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:19:38 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:19:38 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:19:48 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:20:46 archlinux kernel: 18:58:13.900034 Timer    VBoxDRMClient: push screen layout data of 1 display(s) to DRM stack, fPartialLayout=fa
lse
Mar 10 22:20:46 archlinux kernel: hrtimer: interrupt took 4741953 ns
[root@archlinux system]#
```

Для просмотра сообщений текущей загрузки можно добавить параметр `-b`.

```
Mar 10 22:02:32 archlinux sudo[792]: pam_unix(sudo:session): session opened for user root(uid=0) by said(uid=1000)
Mar 10 22:02:32 archlinux systemd[1]: Reload requested from client PID 795 ('systemctl') (unit session-1.scope)...
Mar 10 22:02:32 archlinux systemd[1]: Reloading...
Mar 10 22:02:33 archlinux systemd[1]: Reloading finished in 282 ms.
Mar 10 22:02:33 archlinux sudo[792]: pam_unix(sudo:session): session closed for user root
Mar 10 22:04:37 archlinux NetworkManager[381]: <info>  [1741633477.3847] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:04:37 archlinux NetworkManager[381]: <info>  [1741633477.4111] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:04:37 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:04:37 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:04:47 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:05:07 archlinux sudo[854]:     said : TTY=tty1 ; PWD=/usr/local/bin ; USER=root ; COMMAND=/usr/bin/su
Mar 10 22:05:07 archlinux sudo[854]: pam_unix(sudo:session): session opened for user root(uid=0) by said(uid=1000)
Mar 10 22:05:07 archlinux su[857]: (to root) root on pts/0
Mar 10 22:05:07 archlinux su[857]: pam_unix(su:session): session opened for user root(uid=0) by said(uid=0)
Mar 10 22:07:12 archlinux root[865]: System boot at Mon Mar 10 10:07:12 PM MSK 2025
Mar 10 22:09:36 archlinux systemd[1]: Starting Cleanup of Temporary Directories...
Mar 10 22:09:36 archlinux systemd[1]: systemd-tmpfiles-clean.service: Deactivated successfully.
Mar 10 22:09:36 archlinux systemd[1]: Finished Cleanup of Temporary Directories.
Mar 10 22:09:38 archlinux NetworkManager[381]: <info>  [1741633778.0115] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:09:38 archlinux NetworkManager[381]: <info>  [1741633778.0308] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:09:38 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:09:38 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:09:48 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:14:37 archlinux systemd[1]: Reload requested from client PID 903 ('systemctl') (unit session-1.scope)...
Mar 10 22:14:37 archlinux systemd[1]: Reloading...
Mar 10 22:14:37 archlinux systemd[1]: Reloading finished in 277 ms.
Mar 10 22:14:38 archlinux NetworkManager[381]: <info>  [1741634078.0175] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:14:38 archlinux NetworkManager[381]: <info>  [1741634078.0439] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:14:38 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:14:38 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:14:43 archlinux systemd[1]: Starting Network Manager Wait Online...
Mar 10 22:14:44 archlinux systemd[1]: Finished Network Manager Wait Online.
Mar 10 22:14:44 archlinux systemd[1]: Reached target Network is Online.
Mar 10 22:14:44 archlinux systemd[1]: Starting My Message Service: writes date and time to the system log...
Mar 10 22:14:44 archlinux root[958]: System boot at Mon Mar 10 10:14:44 PM MSK 2025
Mar 10 22:14:44 archlinux systemd[1]: Finished My Message Service: writes date and time to the system log.
Mar 10 22:14:48 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:19:38 archlinux NetworkManager[381]: <info>  [1741634378.0300] dhcp4 (enp0s3): error parsing DHCP option 15 (domain_name): '.' is not a
in
Mar 10 22:19:38 archlinux NetworkManager[381]: <info>  [1741634378.0442] dhcp4 (enp0s3): state changed new lease, address=10.0.2.5
Mar 10 22:19:38 archlinux systemd[1]: Starting Network Manager Script Dispatcher Service...
Mar 10 22:19:38 archlinux systemd[1]: Started Network Manager Script Dispatcher Service.
Mar 10 22:19:48 archlinux systemd[1]: NetworkManager-dispatcher.service: Deactivated successfully.
Mar 10 22:20:46 archlinux kernel: 18:58:13.900034 Timer    VBoxDRMClient: push screen layout data of 1 display(s) to DRM stack, fPartialLayout=fa
lse
Mar 10 22:20:46 archlinux kernel: hrtimer: interrupt took 4741953 ns
```

2. Просмотр сообщений, касающихся сервиса mymsg.

```
journalctl -u mymsg.service
```

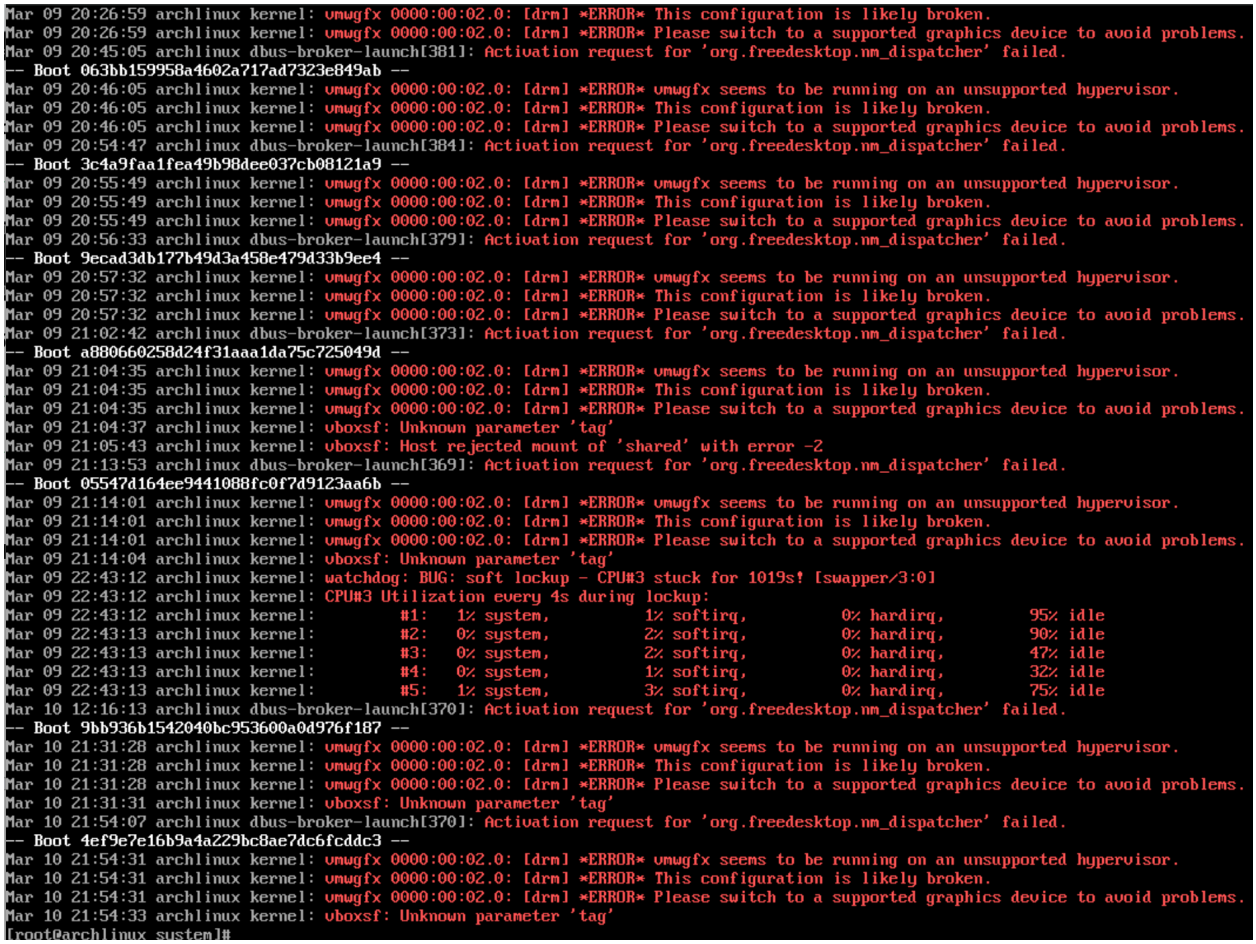
```
[root@archlinux system]# journalctl -u mymsg.service
Mar 10 22:14:44 archlinux systemd[1]: Starting My Message Service: writes date and time to the system log...
Mar 10 22:14:44 archlinux root[958]: System boot at Mon Mar 10 10:14:44 PM MSK 2025
Mar 10 22:14:44 archlinux systemd[1]: Finished My Message Service: writes date and time to the system log.
[root@archlinux system]#
```

3. Вывод сообщений об ошибках.

```
journalctl -p err
```

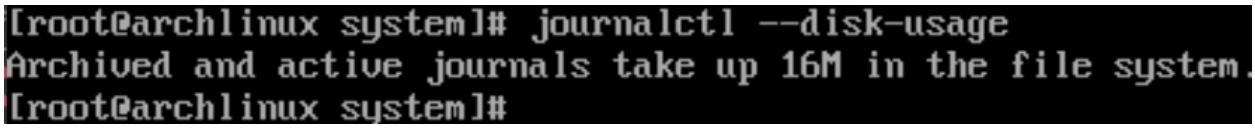
или

```
journalctl -p 3
```



4. Определение размера журнала.

```
journalctl --disk-usage
```



Часть 5. Создание и настройка .mount юнита

1. Подготовка файловой системы

1.1 Раздел/устройство:

Если ещё нет выделенного раздела, создайте его (например, с помощью утилит fdisk/parted). Пусть устройство будет `/dev/sdb1`.

```
[root@archlinux said]# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda      8:0    0    8G  0 disk
├─sda1   8:1    0    1G  0 part /boot
└─sda2   8:2    0    7G  0 part /
sdb      8:16   0    4G  0 disk
└─sdb1   8:17   0    4G  0 part
zram0  254:0    0  3.9G  0 disk [SWAP]
```

1.2 Форматирование в ext4:

```
mkfs.ext4 /dev/sdb1
```

```
[root@archlinux said]# mkfs.ext4 /dev/sdb1
mke2fs 1.47.2 (1-Jan-2025)
Creating filesystem with 1048320 4k blocks and 262144 inodes
Filesystem UUID: 447af993-76e4-4c70-a36e-523edd9de307
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

1.3 Создание точки монтирования:

```
mkdir -p /mnt/mydata
```

2. Создание файла .mount юнита

Файл `/etc/systemd/system/mnt-mydata.mount`:

```
[Unit]
Description=Mounting a partition /mnt/mydata

[Mount]
What=/dev/sdb1
```

```
Where=/mnt/mydata
Type=ext4
Options=defaults

[Install]
WantedBy=multi-user.target
```

3. Запуск и проверка

3.1 Включение юнита:

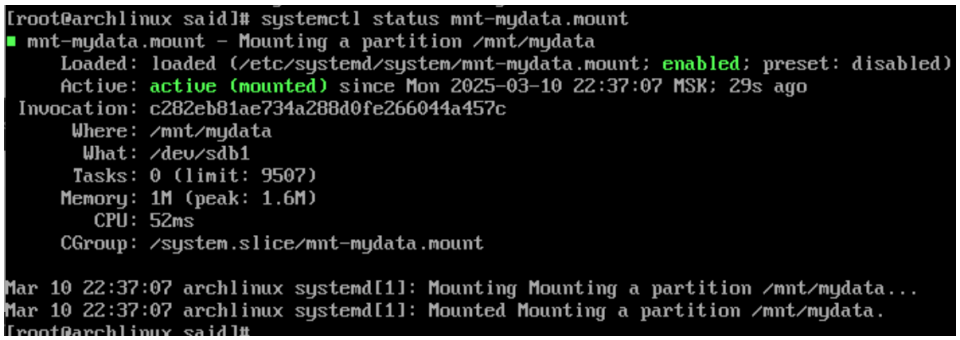
```
systemctl enable mnt-mydata.mount
```

3.2 Запуск:

```
systemctl start mnt-mydata.mount
```

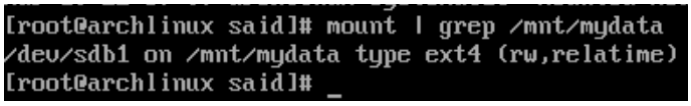
3.3 Проверка статус:

```
systemctl status mnt-mydata.mount
```



3.4 Убедимся, что раздел смонтирован:

```
mount | grep /mnt/mydata
```



Часть 6. Использование .automount для отложенного монтирования

1. Подготовка

Предполагается, что после части 5 уже настроен .mount юнит для `/mnt/mydata`.

2. Создание файла .automount юнита

Файл `/etc/systemd/system/mnt-mydata.automount`:

```
[Unit]
Description=Automount for /mnt/mydata

[Automount]
Where=/mnt/mydata
TimeoutIdleSec=30

[Install]
WantedBy=multi-user.target
```

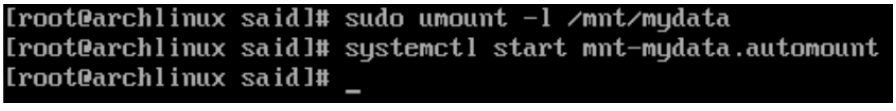
3. Запуск и проверка

3.1 Включаем юнит:

```
systemctl enable mnt-mydata.automount
```

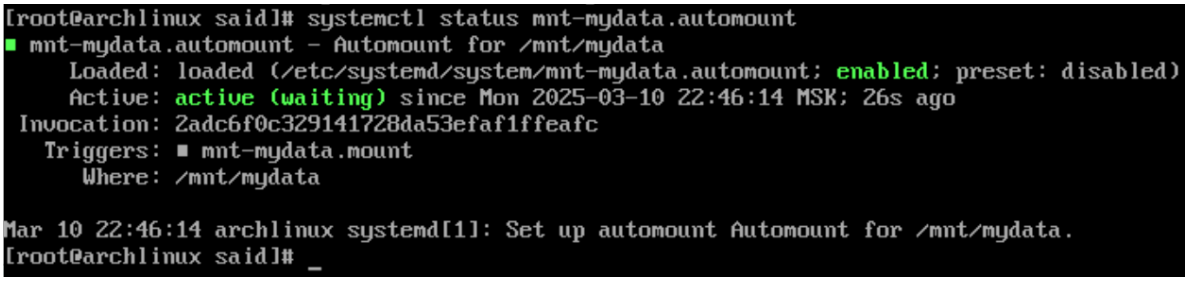
3.2 Запуск:

```
systemctl start mnt-mydata.automount
```



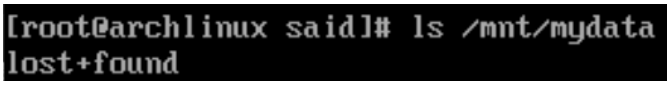
3.3 Проверка статуса:

```
systemctl status mnt-mydata.automount
```



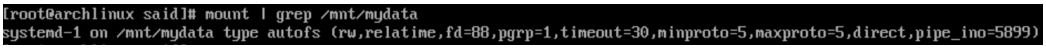
3.4 Проверка автоматического монтирования, обратившись к точке:

```
ls /mnt/mydata
```



3.5 После периода бездействия (30 секунд) проверяем, что раздел отмонтировался:

```
mount | grep /mnt/mydata
```



Вопросы и задания

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

   systemctl restart:
   Останавливает запущенный сервис (если он работает) и запускает его заново. При этом, если сервис не запущен, он будет запущен.

   systemctl try-restart:
   Перезапускает сервис только если он уже работает. Если сервис не запущен, то команда ничего не делает.

   Таким образом, try-restart используется для «мягкого» перезапуска, когда не нужно запускать сервис, который и так не активен.

2. **Как с помощью systemctl запустить Linux в однопользовательском режиме?**

   Для перехода в однопользовательский режим (режим спасения) можно выполнить:

   ```
   systemctl isolate rescue.target
   ```

   Либо, если требуется более минимальная среда,

   ```
   systemctl isolate emergency.target
   ```

3. **Создание сервиса mysrv с условиями:**

Требования:

- Сервис **mysrv** не должен запускаться автоматически.
- Его запуск возможен только после того, как сервис **mymsg** был принудительно остановлен уже после старта системы.

Реализация:

В файле юнита mymsg.service:
(подразумевается что уже создан ранее)

```
[Service]
Type=oneshot
ExecStart=/usr/local/bin/mymsg.sh
RemainAfterExit=yes
```

Здесь параметр **RemainAfterExit=yes** гарантирует, что даже после завершения работы сервиса он будет считаться активным.

Создадим юнит для mysrv, например /etc/systemd/system/mysrv.service:

```
[Unit]
Description=My SRV Service – запуск возможен только если mymsg
остановлен
# Опция Conflicts гарантирует, что если mymsg активен, запуск
mysrv приведёт к конфликту:
Conflicts=mymsg.service
After=mymsg.service

[Service]
Type=simple
ExecStart=/usr/bin/echo "mysrv запущен, mymsg остановлен"

# В разделе [Install] не указываем WantedBy, чтобы юнит не
стартовал автоматически.
```

Пояснения:

- **Conflicts=mymsg.service** – данный параметр не позволяет запускать mysrv одновременно с mymsg. Если mymsg активен, запуск mysrv приведёт к остановке mymsg.
- Отсутствие секции **[Install]** (или отсутствие её включения через systemctl enable) гарантирует, что mysrv не будет запускаться автоматически при загрузке.
- Таким образом, сервис mysrv можно запустить вручную (или по зависимости других юнитов) только после того, как сервис mymsg будет остановлен (например, командой `systemctl stop mymsg.service`).