



Факультет
биоинженерии и биоинформатики
Московский государственный университет имени М.В.Ломоносова



Менеджер инициализации systemd

Загрузка ядра ОС



GRUB загружает в оперативную память образ ядра Linux (файл `vmlinuz`) и файл `initrd.img` на основании конфигурации `grub.cfg`.

- ▶ Образ ядра поставляется в составе системных пакетов `linux-image-*`.
- ▶ Файл `initrd.img` является специфичным для конкретной системы, он генерируется программой `update-initramfs` из пакета `initramfs-tools`.

Загрузка ядра заканчивается монтированием корневой файловой системы и запуском системного демона `init`.

Временная корневая файловая система initramfs



- ▶ Для монтирования корневой файловой системы нужны драйверы.
- ▶ Включение драйверов для всех возможных устройств, на которых может располагаться корневая ФС, сделало бы ядро слишком большим.
- ▶ Решение – временная (расположена в оперативной памяти) корневая ФС, содержащая всё необходимое для монтирования реальной корневой ФС.
- ▶ Временная корневая ФС заполняется из файла `initrd.img`, в современных системах это просто архив в формате `cpio`.
- ▶ Ядро создает временную ФС (`tmpfs`), заполняет её файлами из архива и запускает программу `/init`, которая монтирует корневую ФС и запускает системный демон `init`.

Инициализация пользовательского пространства



После загрузки ядра ОС требуется запустить системные службы (демоны) и настроить устройства: смонтировать файловые системы, настроить сеть и прочее.

В зависимости от режима работы могут требоваться различные службы и устройства.

Задача системы инициализации – выполнить требуемые загрузочные сценарии (по запуску демонов и подготовке ресурсов) в правильном порядке, учитывая их зависимости.

Менеджер systemd



В большинстве современных дистрибутивов Linux система инициализации представлена менеджером systemd.

- ▶ Все объекты в systemd – юниты разных типов.
- ▶ Юниты описываются простыми текстовыми юнит-файлами.
- ▶ Все юниты связаны зависимостями разных типов, разрешая которые система инициализации определяет порядок активации юнитов.
- ▶ Ручное управление юнитами и менеджером осуществляется с помощью программы systemctl.

Юниты systemd



Основные типы юнитов systemd:

- `service` – запуск и управление демонами;
- `target` – соответствуют состояниям системы, нужны для упрощения сети зависимостей;
- `mount` – монтирование файловых систем;
- `device` – управление устройствами;
- `slice` – распределение системных ресурсов;
- `timer` – активация юнитов по расписанию.

...

Пример юнит-файла



```
# /lib/systemd/system/networking.service
```

[Unit]

```
Description=Raise network interfaces  
DefaultDependencies=no  
Requires=ifupdown-pre.service  
Wants=network.target  
After=local-fs.target network-pre.target apparmor.service ...  
Before=network.target shutdown.target network-online.target  
Conflicts=shutdown.target
```

[Install]

```
WantedBy=multi-user.target  
WantedBy=network-online.target
```

[Service]

```
Type=oneshot  
EnvironmentFile=-/etc/default/networking  
ExecStart=/sbin/ifup -a --read-environment  
ExecStop=/sbin/ifdown -a --read-environment --exclude=lo  
RemainAfterExit=true  
TimeoutStartSec=5min
```

Системный демон `init`



Демон `init` – первый системный демон (всегда имеет `pid 1`), с которого начинается порождение всех остальных процессов в пользовательском пространстве.

Процесс `init (/sbin/init)` запускается ядром ОС после монтирования корневой ФС.

Кроме инициализации системы, `init` выполняет функцию родителя для всех "осиротевших" процессов (получает их код возврата).

Многопользовательский режим

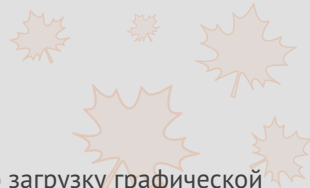


Состояние системы, в котором она готова для входа обычных пользователей.

В systemd этому режиму соответствует специальный юнит `multi-user.target`.

Все юниты, требуемые для входа и работы пользователей, содержат в юнит-файле строчку `WantedBy=multi-user.target` в секции `Install`.

Системные консоли



Переход в многопользовательский режим не предполагает обязательную загрузку графической подсистемы.

Работу с системой пользователи могут осуществлять с помощью системных консолей.

В каждой системной консоли запущена программа `login`, которая ожидает ввода логина и пароля пользователя и запускает командную оболочку от имени пользователя в случае успешной аутентификации.

Вид системной консоли



```
Debian GNU/Linux 11 laptop tty1

laptop login: user
Password:
Linux laptop 4.19.0-18-amd64 #1 SMP Debian 4.19.208-1 (2021-09-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Nov 20 15:19:22 MSK 2023 on tty1
You have no mail.
user@laptop:~$
```

Пользовательские сессии systemd



Контроль за сессиями пользователей осуществляется системным демоном `systemd-logind`.

- ▶ Каждому залогиненному пользователю соответствует юнит типа `slice` для гибкого распределения системных ресурсов.
- ▶ Каждой пользовательской сессии – юнит типа `scope` для упрощения контроля и управления.

Взаимодействовать с менеджером сессий можно с помощью программы `loginctl`.

Графический режим



Для полноценной работы пользователей в графическом окружении, система должна перейти в графический режим.

В systemd этому режиму соответствует специальный юнит `graphical.target`.

Для перехода в графический режим требуется запуск `display-manager.service`.

Менеджер графических сессий



Задачи display-manager:

- ▶ запуск X-сервера;
- ▶ графический аналог программы login;
- ▶ контроль графических пользовательских сессий.

Примеры дисплейных менеджеров: LightDM, GDM, SDDM, ...

X Window System

