



Загрузка системы

О себе



Никифоров Александр

Больше 10 лет работы с GNU/Linux.

Администратор инфраструктуры “облака” в Селектел.

Область интересов - автоматизация abaremetal deployment.

в сети: @burlunder

Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в Slack #linux-2022-12
или #general



Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу

Условные обозначения



Индивидуально



Время, необходимое
на активность



Пишем в чат



Говорим голосом



Документ



Ответьте себе или
задайте вопрос

Маршрут вебинара

Процесс загрузки

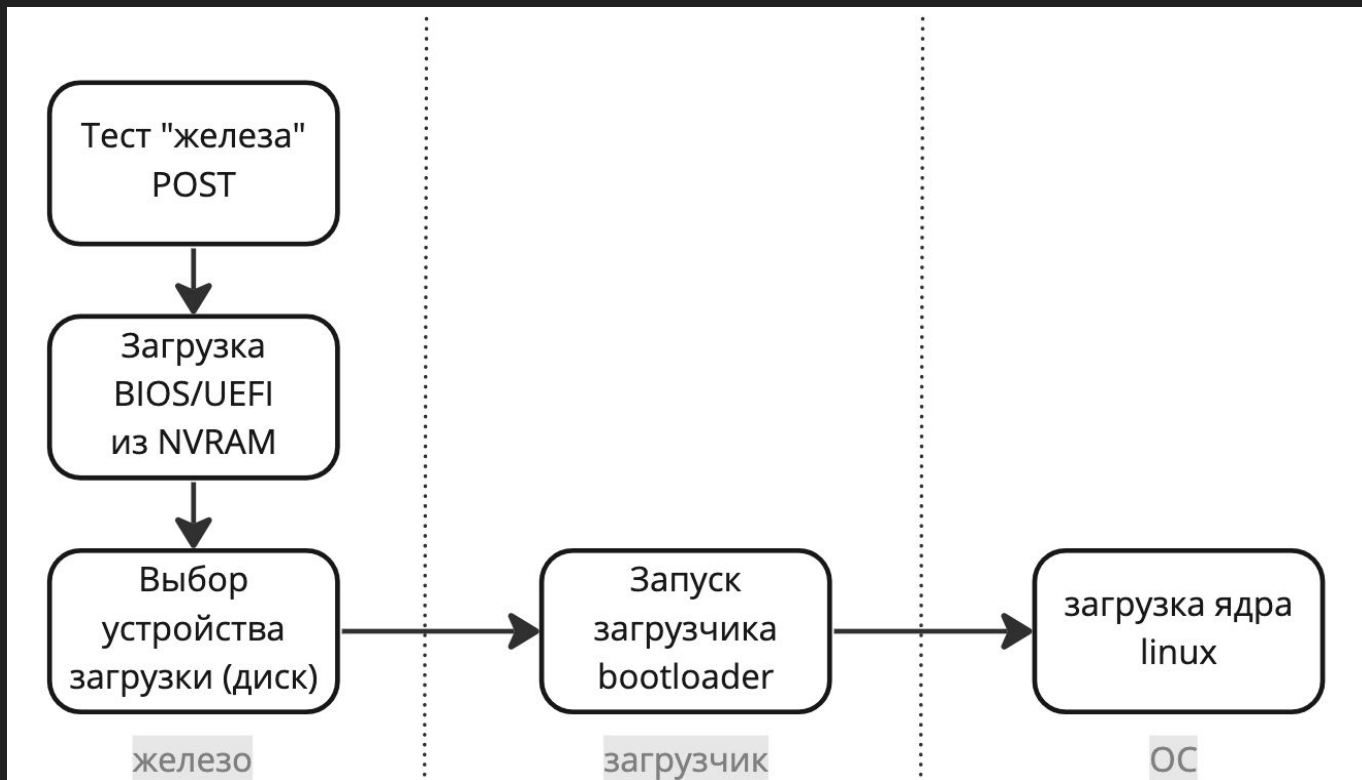
GRUB

параметры загрузки cmdline

initrd

Пример добавления модуля initrd

BIOS/MBR vs UEFI/GPT



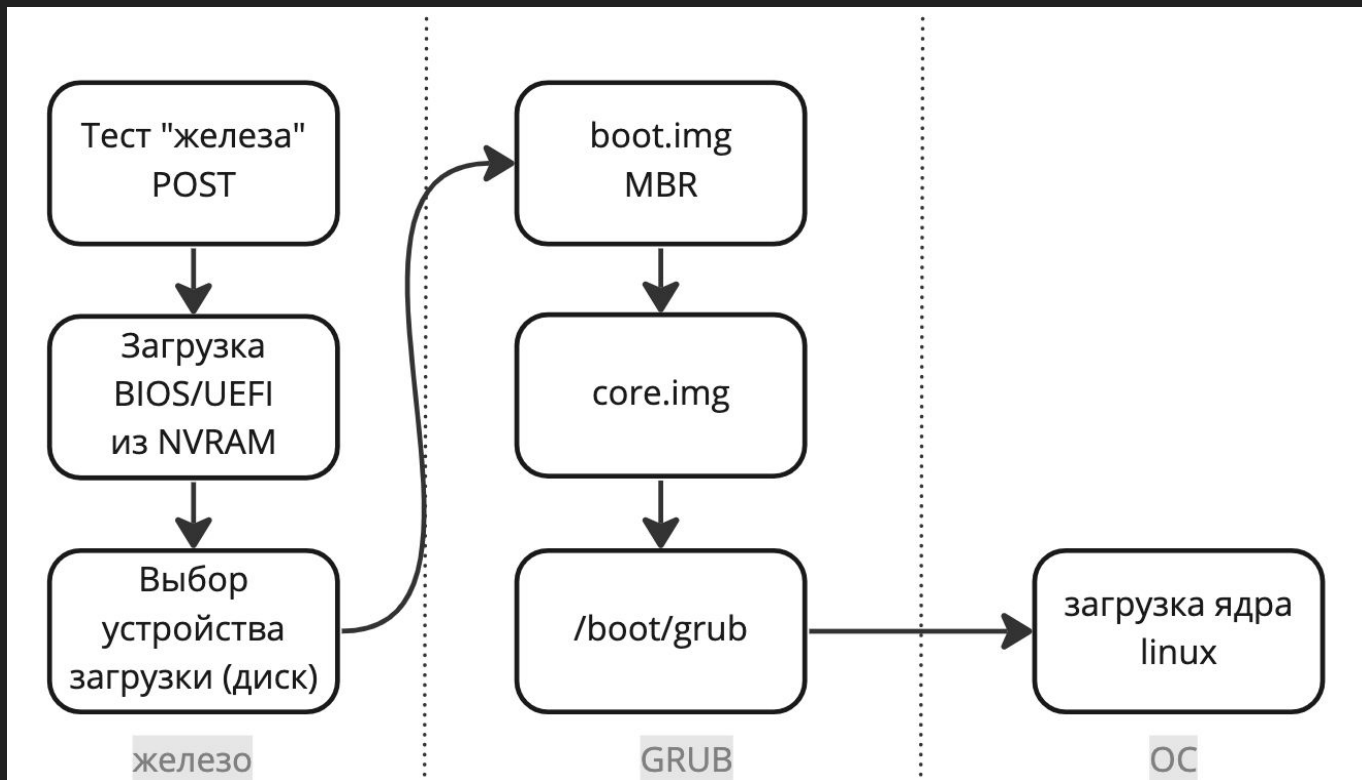
CentOS Linux 7 (Core), with Linux 3.10.0-229.20.1.el7.x86_64
CentOS Linux 7 (Core), with Linux 3.10.0-229.14.1.el7.x86_64
CentOS Linux 7 (Core), with Linux 3.10.0-229.11.1.el7.x86_64
CentOS Linux 7 (Core), with Linux 3.10.0-229.el7.x86_64
CentOS Linux 7 (Core), with Linux 0-rescue-b8499c73d54144fab276d90ae5
CentOS Linux 7 (Core), with Linux 3.10.0-229.20.1.el7.x86_64
CentOS Linux 7 (Core), with Linux 3.10.0-229.14.1.el7.x86_64
CentOS Linux 7 (Core), with Linux 3.10.0-229.el7.x86_64
CentOS Linux 7 (Core), with Linux 3.10.0-229.el7.x86_64
CentOS Linux 7 (Core), with Linux 0-rescue-b8499c73d54144fab276d90ae5

Загрузчик GRUB

Use the ↑ and ↓ keys to change the selection.
Press 'e' to edit the selected item, or 'c' for a command prompt.

GRUB

- GNU
- Актуальный GRUB2 (ранее GRUB Legacy)
- Эталонная реализация загрузчика по спецификации [Multiboot](#)
- Может передавать управление другому загрузчику (multichain booting)
- Модульная архитектура, за счёт чего работает с любым оборудованием (lvm, raid, luks)
- Интерактивная консоль, можно передать ядру произвольные параметры загрузки (rescue режим)



Стадии загрузки (MBR)

- `boot.img`
располагается в области MBR, ограничен 512 байт
- `core.img`
 - между MBR и 1-м разделом диска, 32 Kb *
 - на файловой системе
- `/boot/grub`
 - все остальные модули
 - интерактивная консоль

Файлы в /boot

- **/boot** - по [LFH](#) содержит всё для процесса загрузки до пользовательского режима (user-mode)
- **/boot/grub** - конфигурационные файлы grub
- **/boot/initramfs-*** - образ initrd (init ramdisk)
- **/boot/vmlinuz-*** - ядро Linux
- **/boot/System.map-*** - экспортированные функции ядра. Требуется для компоновки ядра при его сборке

Файлы в `/boot/grub`

- **`/boot/grub/device.map`**
список дисков/разделов в формате GRUB
- **`/boot/grub/grub.cfg`**
основной конфигурационный файл при загрузке GRUB, генерируется автоматом
- **`/boot/grub/grubenv`**
служит для хранения промежуточных данных между перезагрузками

Изменение параметров загрузки

- **/etc/default/grub**
переменные для генерации **grub.cfg**
- **/etc/grub.d/40_custom**
секция скрипта для кастомных пунктов меню загрузки ядер (добавляется как есть в *grub.cfg*)

```
$ grub-mkconfig -o /boot/grub/grub.cfg
```

/etc/grub/default

GRUB_CMDLINE_LINUX - параметры ядра, передаваемые в cmdline

GRUB_DEFAULT - пункт меню по умолчанию

GRUB_TIMEOUT - время до автовыбора (в сек.)

GRUB_SAVEDEFAULT - автосохранение в grubenv последнего выбора

/etc/grub.d/40_custom

```
#!/bin/sh
exec tail -n +3 $
# This file provides an easy way to add custom menu entries. Simp
# menu entries you want to add after this comment. Be careful not
# the 'exec tail' line above.
menuentry 'OTUS Kernel' {
    insmod gzio
    insmod part_msdos
    set root='(hd1,msdos1)'
    linux /otus_kernel root=/dev/sdb1 ro quiet
    initrd /initrd_otus_kernel.img
}
```

Starting Security Auditing Service...

Mounting RPC Pipe File System...

[OK] Mounted RPC Pipe File System.

[OK] Started Tell Plymouth To Write Out Runtime Data.

[18.789668] type=1305 audit(1518461763.459:3): audit_pid=1371 old=0 auid=0

Starting Show Plymouth Power Off Screen...

Starting Restore /run/initramfs...

[OK] Removed slice system-getty.slice.

Stopping LVM2 PV scan on device 8:2...

[OK] Removed slice system-selinux\x2dpolicy\x2dmigrate\x2dlocal\x2dchange

Unmounting RPC Pipe File System...

Stopping Availability of block devices...

[OK] Stopped target Slices.

[OK] Removed slice User and Session Slice.

[OK] Stopped target Encrypted Volumes.

[OK] Stopped target Swap.

Deactivating swap /dev/mapper/rootvg-swaplv01...

[OK] Stopped Apply Kernel Variables.

Stopping Apply Kernel Variables...

Stopping Load/Save Random Seed...

[OK] Stopped Setup Virtual Console.

[19.155324] type=1305 audit(1518461763.824:55): audit_pid=0 old=1371 auid=0

[19.220046] type=1131 audit(1518461763.889:56): pid=1 uid=0 auid=4294967295

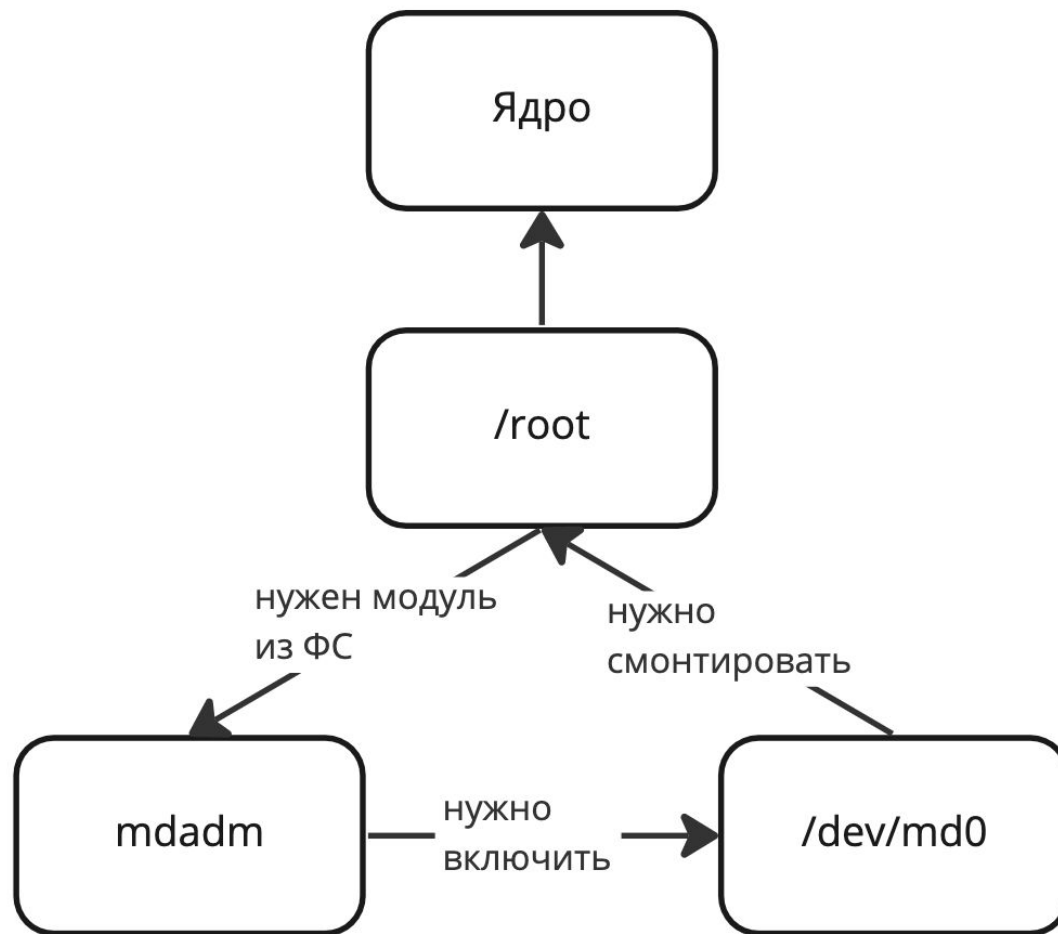
systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=succe

Загрузка ядра Linux

Ядро и /root (проблема курицы и яйца)

На примере RAID

- ядру требуется корневая файловая система / (root)
- /root нужно смонтировать с диска /dev/md0
- для включения /dev/md0 нужен модуль mdadm
- модуль mdadm находится на файловой системе в /lib/modules



initrd (initramdisk, initramfs)

Минимальный образ ФС

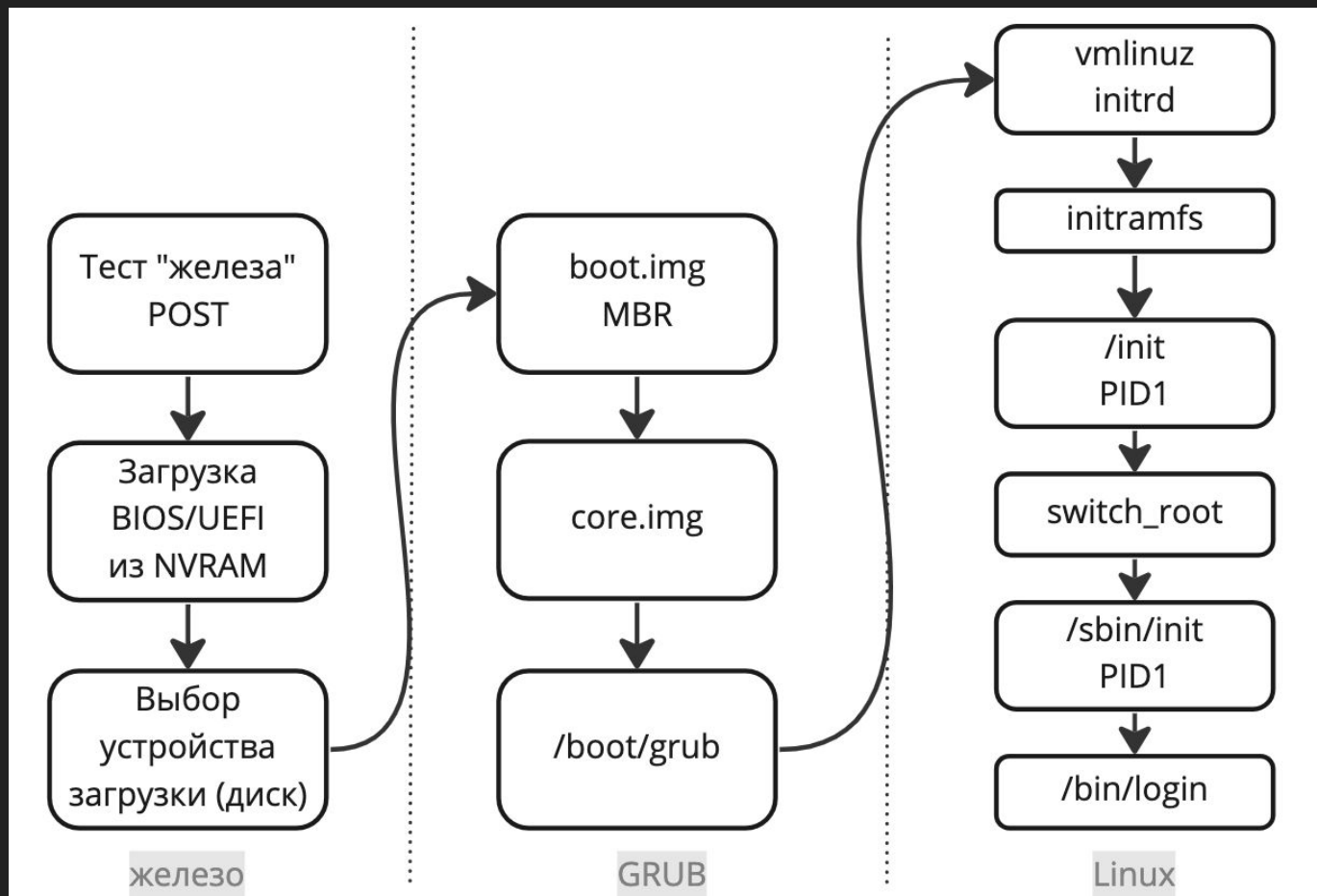
- архив `срjо`, который распаковывается в `rootfs`
- содержит модули, необходимые для загрузки
- скрипт `/init` нужен для поиска устройства, которое будет дальше `root`

initrd (initramdisk, initramfs)

Минимальный образ ФС

- архив `срjо`, который распаковывается в `rootfs`
- содержит модули, необходимые для загрузки
- скрипт `/init` нужен для поиска устройства, которое будет дальше `root`

1. `срjо` распаковывается в **initramfs**
2. **/init**
3. ищется устройство **real root**
4. **switch_root** (`pivot_root`)
5. **/sbin/init**



Использование initrd

список модулей dracut

```
lsinitrd -m /boot/initramfs-$(uname -r).img
```

разборка

```
zcat /boot/initrd-$(uname -r).img | cpio -i
```

сборка

```
find. -print0 |  
cpio -o --null --format=newc |  
gzip -q -9 > /boot/initrd-$(uname -r).img
```

Dracut

<https://www.kernel.org/pub/linux/utils/boot/dracut/dracut.html>

Используется в CentOS для управления **initrd**

- модификация
- просмотр содержимого initrd
- добавление своих скриптов на разные этапы загрузки

Dracut. модули (функции-хуки)

- **cmdline** - самое начало загрузки initd
- **pre-udev** - перед запуском udev-подсистемы
- **pre-trigger** - в процессе запуска udev, возможность ним взаимодействовать
- **pre-mount** - перед монтированием файловых систем
- **mount** - смонтировать root-filesystem
- **pre-pivot** - после монтирования, перед pivot_root
- **cleanup** - перед pivot_root, "подчистка за собой"

Dracut. пример модуля

```
#!/bin/bash
```

```
check() {  
    return 0  
}
```

```
depends() {  
    return 0  
}
```

```
install() {  
    inst_hook cleanup 00 "${moddir}/test.sh"  
}
```

Dracut. пример модуля

```
#!/bin/bash
exec 0<>/dev/console 1<>/dev/console 2<>/dev/console
cat <<'msgend'
```

```
< I'm dracut module >
```

```
-----
```

```
\
```

```
\
```

```
.--.
```

```
|o_o|
```

```
|:_/|
```

```
// \ \
```

```
(| |)
```

```
/'\_ _/\'
```

```
\__)=(__/
```

```
msgend
```

```
sleep 10
```

```
echo " continuing...."
```

BIOS vs UEFI

UEFI

Придумали в IBM для своей же архитектуры Power в конце 80-х. По сути та же маленькая ОС со своей спецификацией

- стартует в защищённом режиме
- знает что такое ФС и GPT, может сразу грузиться с дисков
- может грузиться с дисков > 2TB (GPT)
- модульная архитектура, можно использовать свои драйвера и приложения
- встроенный менеджер загрузки

ESP (EFI system partition)

- при загрузке система ищет специальный раздел с меткой ESP (EF00)
- требует разметки диска GPT
- файловая система FAT
- спецификация UEFI определяет структуру файлов на ESP
EFI\BOOT\BOOTX64.EFI

UEFI Secure Boot

- Призван защищать от буткитов
- Технология защищает от выполнения неподписанного кода не только на этапе загрузки, но и на этапе выполнения ОС, например, как в Windows, так и в Linux проверяются подписи драйверов/модулей ядра. Таким образом, вредоносный код в режиме ядра выполнить будет нельзя.

Но это справедливо только, если нет физического доступа к компьютеру, т.к., в большинстве случаев, при физическом доступе ключи можно заменить на свои.

**Заполните, пожалуйста,
опрос о занятии
по ссылке в чате
<https://otus.ru/polls/56950/>**

Спасибо за внимание!

Приходите на следующие вебинары



Никифоров Александр

системный администратор

@burlunder