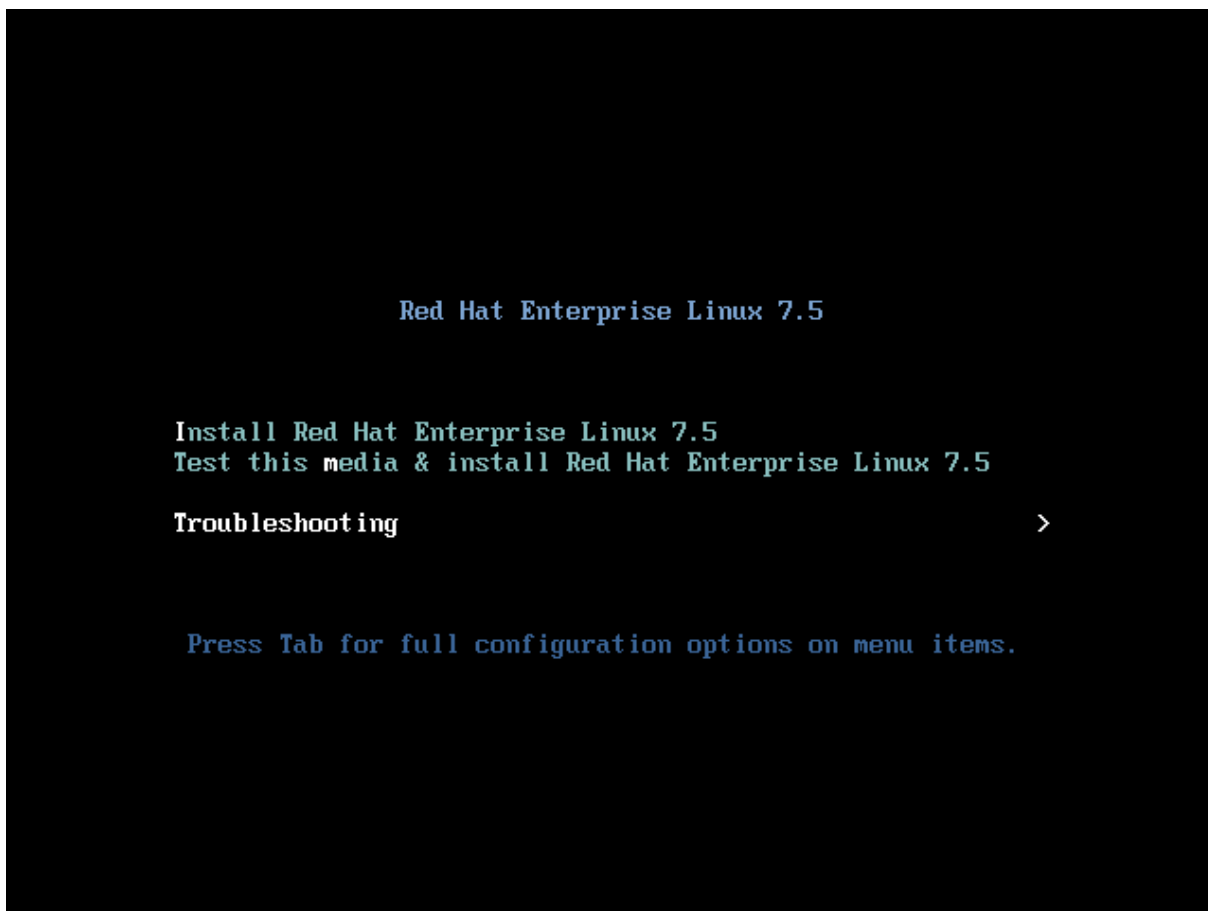


Troubleshooting bootloadera dla systemów Linux

W przypadku posiadania systemu plików XFS na Red Hat 7/CentOS 7 może być wymagane skorzystanie z nośnika instalacyjnego Red Hat 7/CentOS 7 w celu naprawy bootloadera.

Red Hat 7/CentOS 7

Aby przystąpić do naprawy, włóż nośnik instalacyjny CentOS oraz wystartuj z niego system. Następnie wybierz opcję **Troubleshooting** oraz **Rescue a Red Hat Enterprise Linux system** lub **Rescue a CentOS system**.



Zostaniesz zapytany o podjęcie czynności. Wybierz opcję zamontowania systemu znajdującego się na dysku poprzez Opcję **1) Continue**. To spowoduje zamontowanie wspomnianego systemu w punkcie `/mnt/sysimage`.

Ubuntu 12.04/14.04/16.04

Wystartuj nośnik instalacyjny dla jednej z powyższych wersji. Sugerujemy skorzystanie z wersji Ubuntu Desktop (lub Xubuntu Desktop) zamiast Ubuntu Server ze względu na to, że udostępnia nam pulpit graficzny, który jest wygodniejszy w obsłudze nadal posiadając dostęp do terminala. Przy starcie wybieramy **Try Xubuntu without installing**.



GNU GRUB version 2.02~beta2-36ubuntu3.20

```
*Try Xubuntu without installing
Install Xubuntu
OEM install (for manufacturers)
Check disc for defects
```

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the commands before booting or 'c' for a command-line. ESC to return previous menu.

Następnie po wystartowaniu systemu musimy się zorientować jaki jest nasz schemat partycji (nie mylić ze schematem partycjonowania (MBR/GPT)).

W naszym założeniu schemat partycjonowania to:

/dev/nvme0n1p1 – partycja EFI – punkt montowania /boot/efi (w przypadku GPT ta partycja jest wymagana, w przypadku MBR należy tę partycję pominąć)

/dev/nvme0n1p2 – partycja bootowania – punkt montowania /boot

/dev/nvme0n1p3 – główny system plików – punkt montowania /

/dev/nvme0n1p4 – partycja wymiany swap

W Ubuntu należy ręcznie to, co środowisko ratunkowe CentOS/Red Hata robi automatycznie, czyli odpowiednie zamontowanie partycji. W tym przypadku musimy zamontować główny system plików, następnie partycję /boot, a na końcu /boot/efi. Po tym należy umożliwić zamontowanemu systemowi korzystanie z aktualnych zasobów systemowych (zakładając, że punkt montowania dla przywróconego systemu to (/mnt/sysimage)).

```
# mkdir /mnt/mountpoint
# mount /dev/nvme0n1p3 /mnt/sysimage
# mount /dev/nvme0n1p2 /mnt/sysimage/boot
# mount /dev/nvme0n1p1 /mnt/sysimage/boot/efi
# mount -t proc none /mnt/sysimage/proc
# mount -rbind /sys /mnt/sysimage/sys
# mount -rbind /dev /mnt/sysimage/dev
```



Chroot i aktualizacja bootloadera

Następnym elementem, który musimy wykonać jest chroot (change root), co powoduje, że folder, który wskażemy jest traktowany jako nowy punkt montowania głównego systemu plików.

To się przydaje do obsługi systemu operacyjnego, który jest offline (nie wystartowano systemu np. z dysku, lecz z bootowalnego pendrive'a).

Po wykonaniu poprzednich czynności wystarczy wykonać pojedyncze polecenie:

```
# chroot /mnt/sysimage
```

Po wykonaniu chroota są trzy rzeczy do wykonania by mieć pewność, że system wystartuje:

- Poprawność pliku FSTAB,
- Instalacja GRUB oraz generacja listy opcji startu systemów operacyjnych,
- Regeneracja plików initramfs (tylko CentOS/Red Hat).

Aby zrozumieć dlaczego te trzy elementy, należy rozumieć w jaki sposób Linux uruchamia system operacyjny. W najczęstszej konfiguracji dzieli się to na kilka punktów:

1. Start z dysku, uruchamia się bootloader GRUB.
2. GRUB załaduje listę opcji startu systemów operacyjnych, w nim ten jest wybierany z listy ręcznie lub automatycznie kernel (jądro systemu operacyjnego) wraz z plikiem initramfs.
3. Start kernela (jednocześnie systemu operacyjnego).
4. Przepisanie zawartości pliku initramfs do pamięci RAM.
5. Zamontowanie partycji wedle wskazań w pliku FSTAB.

Jeśli jeden z tych punktów zawiedzie, uruchomienie systemu operacyjnego jest niemożliwe.

FSTAB

FSTAB (**f**ile **s**ystem **t**able) to tabela zawierająca listę partycji.

Najczęstszym problemem w przypadku tego pliku jest nieprawidłowe przypisanie woluminów do punktów montowania. Aby poprawić wpisy w tym pliku należy edytować plik **/etc/fstab**, przy czym prawidłowy wpis zawiera następującą składnię:

```
<partycja> <punkt montowania> <system plików> <parametry montowania> <dump> <fsck>
```

Dump to aktualnie niewykorzystywana funkcjonalność archiwizacji partycji, więc zawsze ustawia się wartość 0.

Fsck (file system check) to sprawdzanie poprawności partycji przy starcie, więc partycja root powinna posiadać wartość 1 (sprawdzanie partycji), inne wartość 2 (sprawdź po root). Wartość 0 powoduje brak testu.

Przykład poprawnego wpisu:

```
/dev/sda15 /boot/efi vfat defaults 0 0
```

Wedle powyższego schematu przykładowa poprawna konfiguracja FSTAB to:



<code>/dev/nvme0n1p3</code>	<code>/</code>	<code>xf</code>	<code>defaults</code>	<code>0 1</code>
<code>/dev/nvme0n1p2</code>	<code>/boot</code>	<code>ext2</code>	<code>defaults</code>	<code>0 2</code>
<code>/dev/nvme0n1p1</code>	<code>/boot/efi</code>	<code>vf</code>	<code>defaults</code>	<code>0 2</code>

Domyślne konfiguracje FSTAB na niektórych dystrybucjach mają 0 przy fsck dla wszystkich partycji. To nie powoduje braku możliwości startu systemu operacyjnego. Robi się to po to, by system startował szybciej, ponieważ fsck może trwać od kilku sekund do kilkunastu minut.

UWAGA: czasami zamiast oznaczeń partycji mogą się pojawiać oznaczenia związane z logicznymi woluminami LVM, unikalnymi identyfikatorami UUID czy też etykietami dysków (LABEL= lub PARTLABEL= w przypadku używania GPT).

W przypadku LVM sprawdź poprzez polecenie **lvdisplay** jakie posiadasz woluminy i przypisz je odpowiednio do punktów montowania.

Najczęstsze dwa schematy nazewnictwa to:

Ubuntu:

`/dev/mapper/nazwalv`

RHEL/CentOS:

`/dev/nazwavg/nazwalv`

Gdzie nazwavg o nazwa VG (volume group) i nazwalv to nazwa LV (logical volume), czyli woluminu logicznego.

W przypadku UUID i etykiet zalecam do przejrzania przykładów konfiguracji na Wiki Gentoo Linuksa:

<https://wiki.gentoo.org/wiki/Fstab>

GRUB:

GRUB jest aktualnie najczęściej wykorzystywanym bootloadere dla systemów Linux. Jest alternatywą dla starszych rozwiązań jak LILO i jest on w stanie uruchamiać inne systemy operacyjne takie jak *BSD, OS X czy Windows.

Jest on potrzebny do startu systemu i w jego kwestii należy tylko upewnić się, że bootloader jest zainstalowany, a lista systemów jest wygenerowana względem aktualnego stanu w komputerze. Przy generowaniu listy GRUB nie bierze pod uwagę systemów znajdujących się na nośnikach USB typu pendrive.

Aby zainstalować bootloader, należy wykonać polecenie względem schematu partycjonowania:

MBR:

`# grub-install /dev/nvme0n1`

GPT:

`# grub-install /dev/nvme0n1 --efi-directory=/boot/efi`

(`/dev/nvme0n1` to nazwa dysku, a nie partycji, może to też być np. `/dev/sda` zamiast `/dev/sda1`)

UWAGA: niektóre systemy nie mają polecenia `grub-install`. W takiej sytuacji należy napisać `grub2-install`. Jest to związane z nowszą wersją tego bootloadera. Co prawda, ona jest już we wszystkich



nowoczesnych systemach, ale niektóre dystrybucje chcąc zachować konwencję poleceń ze starszej wersji 1 posiadają wspomniane w przykładzie polecenie.

Polecenie **grub2-install** znajduje się w systemach RHEL/CentOS 7, **grub-install** jest w RHEL/CentOS 6 oraz Ubuntu.

Drugim elementem naprawy GRUBa jest wygenerowanie listy dostępnych systemów na maszynie. Aby to zrobić, należy wykonać konkretne polecenie względem konkretnej dystrybucji Linuksa:

Ubuntu:

```
# update-grub
```

RHEL/CentOS 6:

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

RHEL/CentOS 7:

```
# grub2-mkconfig -o /boot/grub/grub.cfg
```

Regeneracja plików initramfs (tylko CentOS/Red Hat)

Czasami jest potrzebna regeneracja plików initramfs. Wiąże się to z wykorzystaniem modułu dracut we wspomnianych systemach. Przy zmianach w systemie (takich jak np. migracja na inny sprzęt lub zmiana schematu partycjonowania) ten plik powinien być utworzony od nowa.

Aby to zrobić, należy po wykonaniu chroota sprawdzić jaką wersję jądra wykorzystujemy w systemie poprzez polecenie **ls -l /boot**, a następnie wykonanie polecenia:

```
# dracut -f /boot/initramfs-wersja-kernela wersja-kernela
```

Na przykład:

```
# dracut -f /boot/initramfs-3.10.0-514.26.2.e17.x86_64.img 3.10.0-514.26.2.e17.x86_64
```

Poniżej przedstawiamy wykonanie w praktyce.



```

sh-4.2# chroot /mnt/sysimage
bash-4.2# update-env
bash: update-env: command not found
bash-4.2# ls /boot
config-3.10.0-862.14.4.el7.x86_64          symvers-3.10.0-862.14.4.el7.x86_64.gz
config-3.10.0-957.1.3.el7.x86_64        symvers-3.10.0-957.1.3.el7.x86_64.gz
config-3.10.0-957.5.1.el7.x86_64       symvers-3.10.0-957.5.1.el7.x86_64.gz
config-4.1.12-124.24.1.el7uek.x86_64   symvers-4.1.12-124.24.1.el7uek.x86_64.gz
config-4.14.35-1844.1.3.el7uek.x86_64  symvers-4.14.35-1844.1.3.el7uek.x86_64.gz
config-4.14.35-1844.2.5.el7uek.x86_64  symvers-4.14.35-1844.2.5.el7uek.x86_64.gz
efi                                       System.map-3.10.0-862.14.4.el7.x86_64
grub2                                     System.map-3.10.0-957.1.3.el7.x86_64
initramfs-0-rescue-29eddeb0459d457392eba9d51bdf81f9.img System.map-3.10.0-957.5.1.el7.x86_64
initramfs-3.10.0-862.14.4.el7.x86_64.img System.map-4.1.12-124.24.1.el7uek.x86_64
initramfs-3.10.0-862.14.4.el7.x86_64kdump.img System.map-4.14.35-1844.1.3.el7uek.x86_64
initramfs-3.10.0-957.1.3.el7.x86_64.img System.map-4.14.35-1844.2.5.el7uek.x86_64
initramfs-3.10.0-957.5.1.el7.x86_64kdump.img vmlinuz-0-rescue-29eddeb0459d457392eba9d51bdf81f9
initramfs-4.1.12-124.24.1.el7uek.x86_64.img vmlinuz-3.10.0-862.14.4.el7.x86_64
initramfs-4.1.12-124.24.1.el7uek.x86_64kdump.img vmlinuz-3.10.0-957.1.3.el7.x86_64
initramfs-4.14.35-1844.1.3.el7uek.x86_64.img vmlinuz-3.10.0-957.5.1.el7.x86_64
initramfs-4.14.35-1844.1.3.el7uek.x86_64kdump.img vmlinuz-4.1.12-124.24.1.el7uek.x86_64
initramfs-4.14.35-1844.2.5.el7uek.x86_64.img vmlinuz-4.14.35-1844.1.3.el7uek.x86_64
initramfs-4.14.35-1844.2.5.el7uek.x86_64kdump.img vmlinuz-4.14.35-1844.2.5.el7uek.x86_64
bash-4.2# cd /boot
bash-4.2# pwd
/boot
bash-4.2# dracut -f /boot/initramfs-
initramfs-0-rescue-29eddeb0459d457392eba9d51bdf81f9.img initramfs-4.1.12-124.24.1.el7uek.x86_64.img
initramfs-3.10.0-862.14.4.el7.x86_64.img             initramfs-4.1.12-124.24.1.el7uek.x86_64kdump.img
initramfs-3.10.0-862.14.4.el7.x86_64kdump.img       initramfs-4.14.35-1844.1.3.el7uek.x86_64.img
initramfs-3.10.0-957.1.3.el7.x86_64.img             initramfs-4.14.35-1844.1.3.el7uek.x86_64kdump.img
initramfs-3.10.0-957.1.3.el7.x86_64kdump.img       initramfs-4.14.35-1844.2.5.el7uek.x86_64.img
initramfs-3.10.0-957.5.1.el7.x86_64.img             initramfs-4.14.35-1844.2.5.el7uek.x86_64kdump.img
bash-4.2# dracut -f /boot/initramfs-
initramfs-0-rescue-29eddeb0459d457392eba9d51bdf81f9.img initramfs-4.1.12-124.24.1.el7uek.x86_64.img
initramfs-3.10.0-862.14.4.el7.x86_64.img             initramfs-4.1.12-124.24.1.el7uek.x86_64kdump.img
initramfs-3.10.0-862.14.4.el7.x86_64kdump.img       initramfs-4.14.35-1844.1.3.el7uek.x86_64.img
initramfs-3.10.0-957.1.3.el7.x86_64.img             initramfs-4.14.35-1844.1.3.el7uek.x86_64kdump.img
initramfs-3.10.0-957.1.3.el7.x86_64kdump.img       initramfs-4.14.35-1844.2.5.el7uek.x86_64.img
initramfs-3.10.0-957.5.1.el7.x86_64.img             initramfs-4.14.35-1844.2.5.el7uek.x86_64kdump.img
bash-4.2# dracut -f /boot/initramfs-4.14.35-1844.2.5.el7uek.x86_64.img 4.14.35-1844.2.5.el7uek.x86_64
/etc/sysconfig/network-scripts/ifcfg-lan: line 10: /tmp: Is a directory
/etc/sysconfig/network-scripts/ifcfg-lo: line 10: /tmp: Is a directory

bash-4.2#
bash-4.2#
[anaconda] 1:main* 2:shell 3:log 4:storage-log 5:program-log Switch tab: Alt+Tab | Help: F1

```

Po wykonaniu powyższych czynności system uruchomi się poprawnie po restarcie.

