

Stredná odborná škola elektrotechnická Žilina

Inštalácia a konfigurácia OS UBUNTU na server Fujitsu PRIMERGY TX100 S3

verzia 1.0

Radoslav Michal Kopera

3.X.2014

Táto práca vznikla na základe oboznamovania sa s OS UBUNTU na serveri FUJITSU TX100. Na pomoc mi bolo fórum www.ubuntu.cz a traja konzultanti z radov mojich známych, ktorí mali väčšie skúsenosti s riešením rôznych problémov v linuxe. Ďalšie zdroje informácií sú uvádzané priebežne v tejto práci. Táto práca neprešla jazykovou ani odbornou kontrolou.

Prípadného nadšenca kopírovania prosím aby sa nehanbil uviesť do zoznamu literatúry alebo na iné viditeľné miesto aj túto moju príručku (názov tejto práce a moje meno ako autora). Kopírovanie informácií bez uvedenia zdroja sa považuje za plagiát. Okopírovať niečo a spomenúť autora textu je česť pre autora.

OBSAH

1.	Nastavenie BIOS a HW RAID vo Fujitsu TX100 S3	1
2.	Príkazy pre UBUNTU	2
2.1.	Inštalácia grafického rozhrania.....	2
2.1.1.	GNOME	2
2.1.2.	MATE.....	2
2.2.	REŠTART PC	2
2.3.	VYPNUTIE PC	2
2.4.	Aktualizácie.....	3
2.5.	Kontrola kódu.....	3
2.6.	Veriza systému	3
2.7.	TCPDUMP – logovanie komunikácie ze LAN kartu.	3
3.	Inštalácia UBUNTU server 14.10 na RAID1	5
3.1.	Kontrola RAID1	25
3.2.	Výmena HDD v poli RAID1	25
3.2.1.	HW simulácia poruchy v poli RAID1	25
3.2.2.	Výmena poškodeného HDD v poli RAID1	26
3.2.2.1.	Formátovanie nového HDD.....	27
3.2.3.	Pripojenie nového disku do pol'a RAID1	28
3.2.4.	Kontrola RAIDu.....	30
3.2.5.	Obnova – oprava BOOTLOADERa	31
3.3.	Automatická kontrola RAID1	32
3.3.1.	Vytvorenie skriptov.....	32
3.3.2.	Časové spúšťanie kontroly RAID1	37
4.	Užívateľské účty.....	40
4.1.	Vytvorenie užívateľského účtu s právami – USER	40
4.2.	Vytvorenie užívateľského účtu s právami – ROOT	40
5.	Literatúra a prílohy	42
5.1.	Softwarové RAID pole - 1. díl	42
5.1.1.	Úvodem	42
5.1.2.	Příprava.....	43
5.1.3.	Vytvoření nového RAID disku	43
5.1.4.	Připojení RAIDu	43
5.1.5.	Manuální kontrola stavu	44
5.1.6.	Simulace poruchy disku	44
5.1.7.	Oprava poškozeného pole.....	45
5.1.8.	Poškození dat	46
5.1.9.	Automatické monitorování stavu	46
5.2.	Softwarové RAID pole - 2. díl	49
5.2.1.	Úvodem	49
5.2.2.	Příprava.....	49
5.2.3.	Přidání dalšího disku	49
5.2.4.	Instalace SW	50
5.2.5.	Příprava nového disku	50
5.2.6.	Vytvoření diskového pole.....	51
5.2.7.	Přesun dat a úprava zavaděče	52
5.2.8.	Úprava původního disku, kompletace pole	54
5.3.	Softwarové RAID pole - 3. díl	57

5.3.1.	Úvodem	57
5.3.2.	Příprava.....	57
5.3.3.	Zastavení RAIDu	57
5.3.4.	Vyčištění disků.....	58
5.3.5.	Závěrem.....	58
5.4.	RAID - degraded.....	59
5.4.1.	Degraded RAID	59
5.4.2.	RAID Maintenance	59
5.5.	Cron I.	61
5.5.1.	Crontab	61
5.5.2.	Formát souboru.....	61
5.6.	Jak na démona Cron.....	61
5.7.	CRON II.	65
5.7.1.	Cron – správca úloh	65
5.7.2.	Používame cron.....	66
5.7.3.	Cron tabuľky.....	67
5.7.4.	Niekoľko príkladov	68
5.7.5.	Kto bude mať prístup?.....	69
5.7.6.	Výstup z cronu	69
5.7.7.	Tipy	70
5.8.	Shutdown.....	71
5.8.1.	About shutdown.....	71
5.8.2.	Syntax	71
5.8.3.	Description.....	71
5.8.4.	Options	71
5.8.5.	Details.....	72
5.8.6.	Access Control.....	72
5.8.7.	Halting vs. Powering Off.....	73
5.8.8.	Files	73
5.8.9.	Examples	73

1. Nastavenie BIOS a HW RAID vo Fujitsu TX100 S3

Operačný systém Ubuntu server idem inštalovať na PC (server) Fujitsu PRIMERGY TX100 S3. Tento hardvér podporuje len OS Windows server 2008 a novší a Linux server RED HAT a SUSE. Po nainštalovaní Windows server 2008 ide hardvérový RAID vynikajúco. Po nainštalovaní ubuntu rôznych grafických rozhraní (XBUNTU, KBUNTU, LBUNTU.....) nefunguje RAID správne alebo vôbec. Softvérový RAID obsiahnutý v inštalácii UBUNTU server funguje veľmi dobre.

2. Príkazy pre UBUNTU

2.1. Inštalácia grafického rozhrania

2.1.1. GNOME

```
Sudo apt-get install ubuntu-desktop
```

2.1.2. MATE

```
sudo apt-add-repository ppa:ubuntu-mate-dev/ppa
```

```
sudo apt-add-repository ppa:ubuntu-mate-dev/trusty-mate
```

```
sudo apt-get update && sudo apt-get upgrade
```

```
sudo apt-get install --no-install-recommends ubuntu-mate-core ubuntu-mate-desktop
```

Inštalácia po potvrdení „Y“ trvala asi 60 min.

!!! MATE nemá nainštalovaný program GEDIT!!!

Buď si ho doinštaluj (ak si začiatočník ako ja, tak si ho doinštaluj) alebo použi iný editor.

2.2. REŠTART PC

```
sudo reboot
```

Alebo

```
shutdown -r now
```

```
shutdown -r 8:08 reštart sa realizuje v 8:08
```

2.3. VYPNUTIE PC

```
shutdown -P now vynutie spôsobí aj odpojenie napájania
```

```
shutdown now vynutie spôsobí vnutie systému bez odpojenia napájania. PC je možné cez CTRL+ALT+DEL reštartovať a systém sa načíta.
```

```
shutdown -P 9:10 vynutie spôsobí aj odpojenie napájania v čase 9:10.
```

2.4. Aktualizácie

```
Sudo apt-get update
```

2.5. Kontrola kódu

```
sudo bash -vx 5rm_faulty.sh > 5rm_faulty.log 2>&1
```

Po spustení skriptu „5rm_faulty.sh“ zapíše do logu „5rm_faulty.log“ výsledok vykonanej úlohy

2.6. Verzia systému

```
lsb_release -a
```

2.7. TCPDUMP – logovanie komunikácie ze LAN kartu.

Vyhľadaj uloženie programu TCPDUMP.
Například:

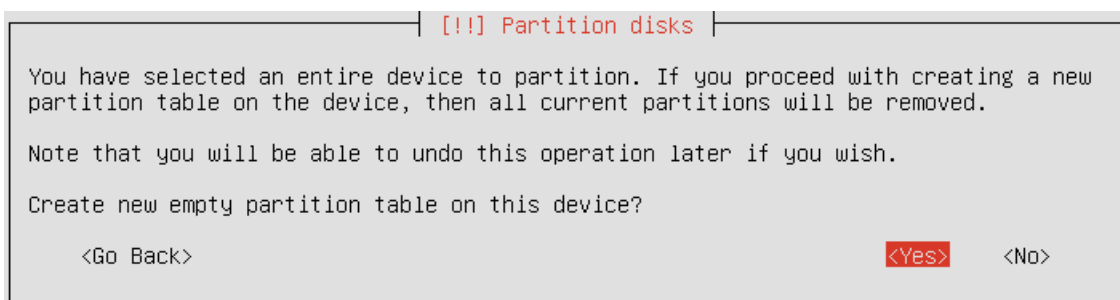
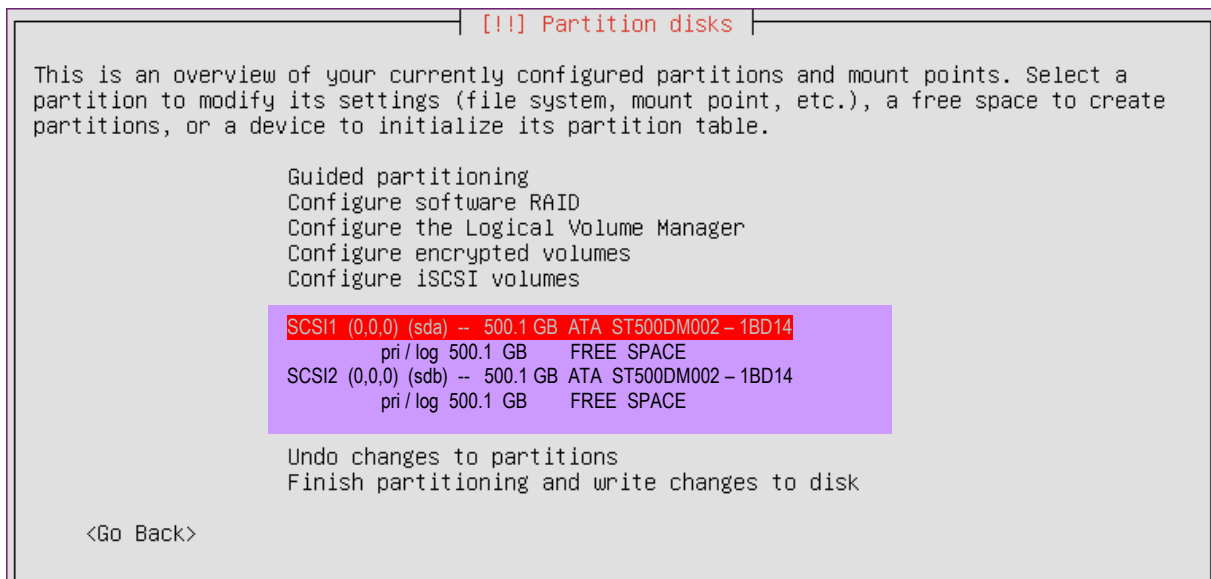
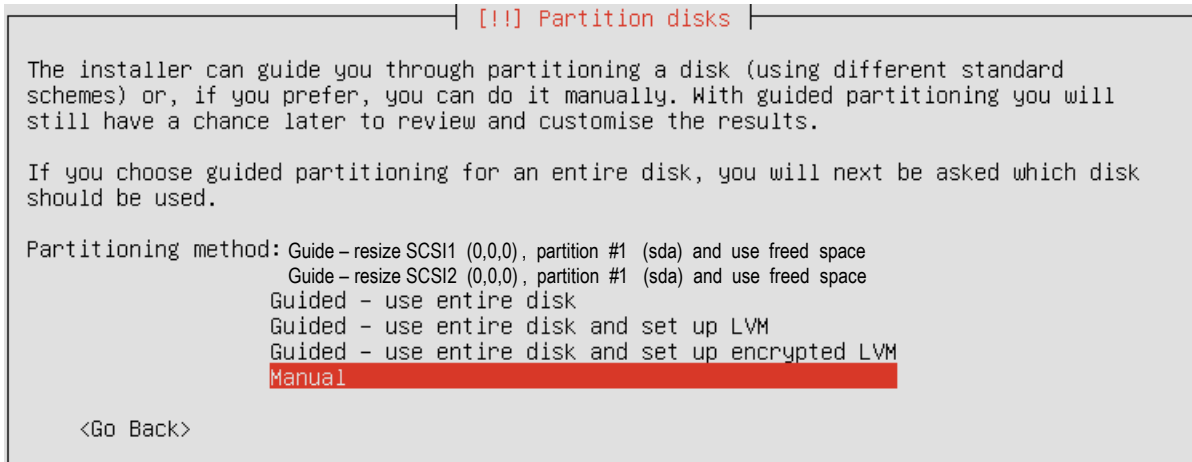
```
cd /usr/sbin
```

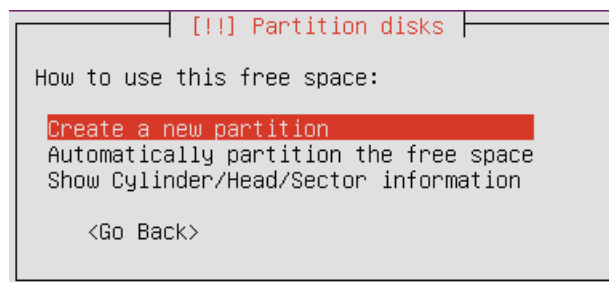
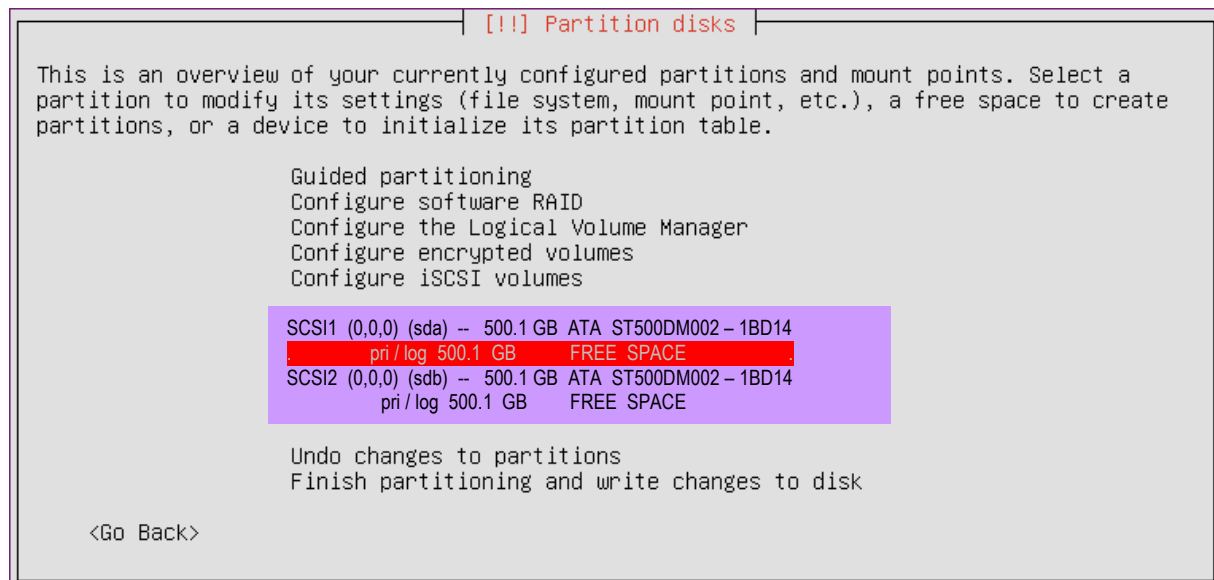
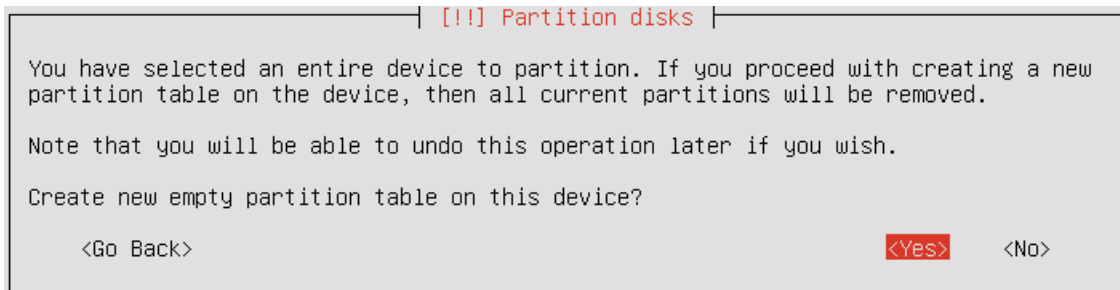
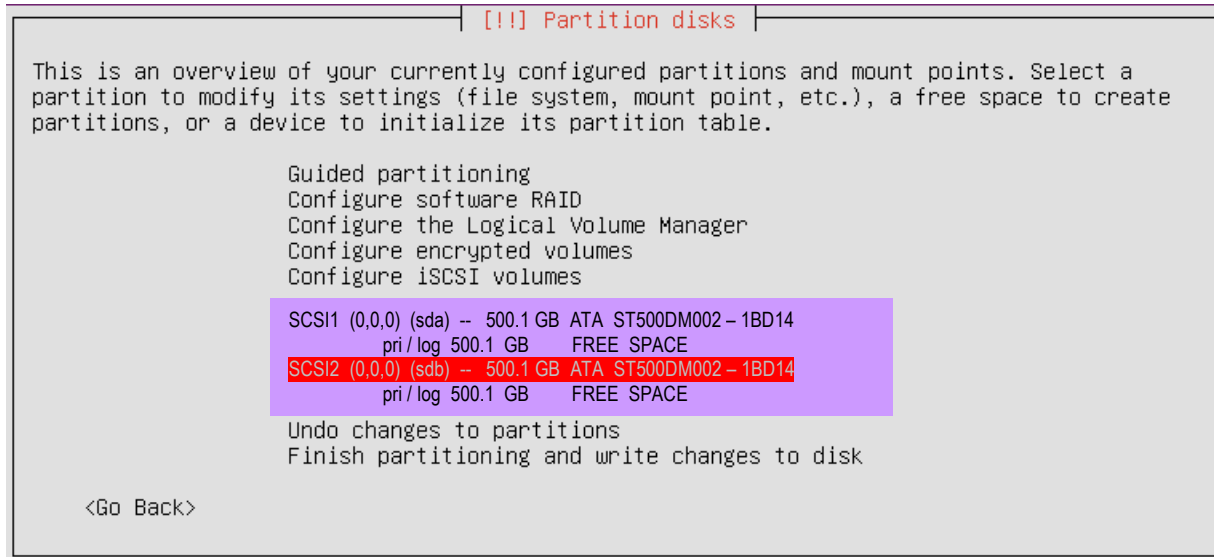
```
kopera@kopera:/usr/sbin$ sudo tcpdump
```

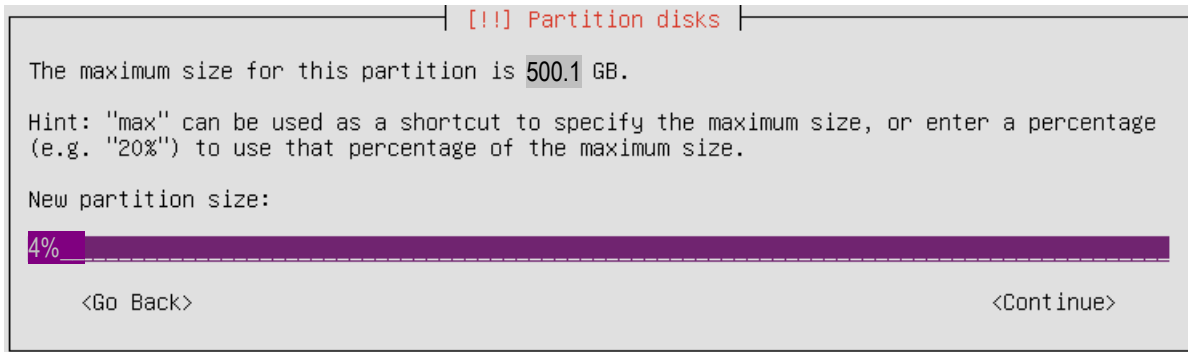

3. Inštalácia UBUNTU server 14.10 na RAID1

Svoje poznámky začnem až od konfigurácie softvérového RAID1. Obrázky sú robené na PC vo Virtual boxe. Tento návod je však odskúšaný na reálnom HW. Obrázky fotené z monitora neboli moc pekné preto som si pomohol VirtualBoxom. Obrázky som dodatočne upravoval pre môj HW, preto bude písmo troška inej veľkosti v niektorých obrázkoch.

Ja mám vo svojom PC (serveri) 2x500GB HDD.







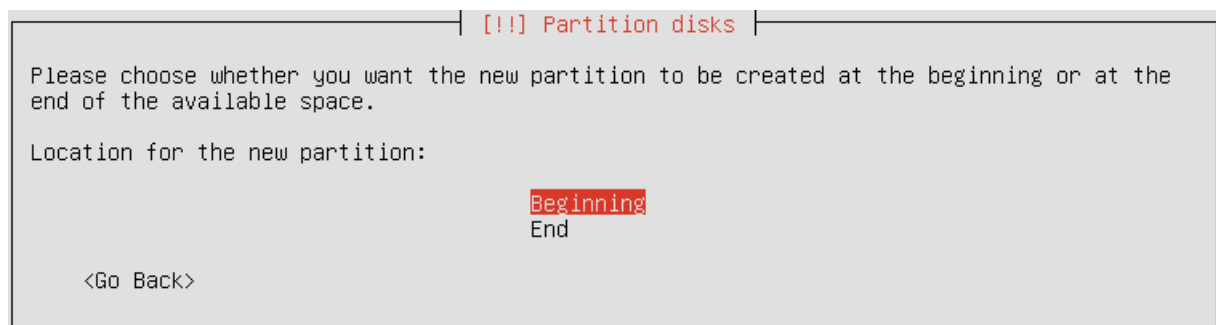
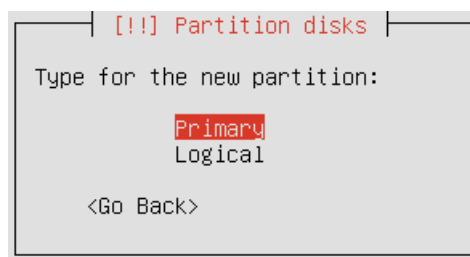
Poznámka: SWAP môžem zvoliť konkrétnu veľkosť alebo percentá z disku. Ak chceš inštalovať znova OS aby si si to preskúšal je nutné aby si SWAP zvolil inú veľkosť. Buď daj 9% alebo 11%. Inštalátor prepíše tabuľku rozdelenia diskov a inštalácia bude fungovať bez problémov. Ak zvolíš partície rovnakej veľkosti ako v predchádzajúcej inštalácii, inštalácia sa síce dokončí ale po reštarte začne vypisovať chybu:

System má snahu nabehnúť no potom na chvíľu vypíše:
 error: diskfilter writes are not supported.
 Press any key to continue ...

ak nestlačím nič, po chvíli sa to rozbehne ako keby som stlačil nejakú klávesu.

Potom sa zdá, že systém nabieha ale skončí to hláškou, ktorá sa opakuje dookola:

```
Incrementally starting RAID arrays...
mdadm: CREATE user root not found
mdadm: CREATE group disk not found
Incrementally started RAID arrays.
Incrementally starting RAID arrays...
mdadm: CREATE user root not found
mdadm: CREATE group disk not found
Incrementally started RAID arrays.
```



```

| [!] Partition disks |
You are editing partition #1 of SCSI3 (0,0,0) (sda). No existing file system was detected
in this partition.

Partition settings:

Use as:      Ext4 journaling file system

Mount point: /
Mount options: defaults
Label:      none
Reserved blocks: 5%
Typical usage: standard
Bootable flag: off

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>
```

```

| [!] Partition disks |
How to use this partition:

Ext4 journaling file system
Ext3 journaling file system
Ext2 file system
btrfs journaling file system
JFS journaling file system
XFS journaling file system
FAT16 file system
FAT32 file system
swap area
physical volume for encryption
physical volume for RAID
physical volume for LVM
do not use the partition

<Go Back>
```

```

| [!] Partition disks |
You are editing partition #1 of SCSI3 (0,0,0) (sda). No existing file system was detected
in this partition.

Partition settings:

Use as:      physical volume for RAID

Bootable flag: off

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>
```

[!] Partition disks

This is an overview of your currently configured partitions and mount points. Select a partition to modify its settings (file system, mount point, etc.), a free space to create partitions, or a device to initialize its partition table.

- Guided partitioning
- Configure software RAID
- Configure the Logical Volume Manager
- Configure encrypted volumes
- Configure iSCSI volumes

```
SCSI1 (0,0,0) (sda) -- 500.1 GB ATA ST500DM002-1BD14
#1 primary 20.0 GB K raid
. pri/log 480.1 GB FREE SPACE
SCSI2 (0,0,0) (sdb) -- 500.1 GB ATA ST500DM002-1BD14
pri/log 500.1 GB FREE SPACE
```

Undo changes to partitions
Finish partitioning and write changes to disk

<Go Back>

[!] Partition disks

How to use this free space:

- Create a new partition
- Automatically partition the free space
- Show Cylinder/Head/Sector information


<Go Back>

[!] Partition disks

The maximum size for this partition is **480.1** GB.

Hint: "max" can be used as a shortcut to specify the maximum size, or enter a percentage (e.g. "20%") to use that percentage of the maximum size.

New partition size:

480.1 GB 

<Go Back> <Continue>

[!] Partition disks

Type for the new partition:

- Primary
- Logical

<Go Back>


```

[!!] Partition disks

You are editing partition #2 of SCSI3 (0,0,0) (sda). No existing file system was detected
in this partition.

Partition settings:

Use as:      Ext4 journaling file system

Mount point: /
Mount options: defaults
Label:      none
Reserved blocks: 5%
Typical usage: standard
Bootable flag: off

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>
```

```

[!!] Partition disks

How to use this partition:

Ext4 journaling file system
Ext3 journaling file system
Ext2 file system
btrfs journaling file system
JFS journaling file system
XFS journaling file system
FAT16 file system
FAT32 file system
swap area
physical volume for encryption
physical volume for RAID
physical volume for LVM
do not use the partition

<Go Back>
```

```

[!!] Partition disks

You are editing partition #2 of SCSI3 (0,0,0) (sda). No existing file system was detected
in this partition.

Partition settings:

Use as:      physical volume for RAID

Bootable flag: off

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>
```

```

[!!!] Partition disks

You are editing partition #2 of SCSI3 (0,0,0) (sda). No existing file system was detected
in this partition.

Partition settings:

Use as:          physical volume for RAID

Bootable flag:  on

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>
```

```

[!!!] Partition disks

This is an overview of your currently configured partitions and mount points. Select a
partition to modify its settings (file system, mount point, etc.), a free space to create
partitions, or a device to initialize its partition table.

Guided partitioning
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes

SCSI1 (0,0,0) (sda) -- 500.1 GB ATA ST500DM002-1BD14
#1 primary 20.0 GB  K  raid
#2 primary 480.0 GB B  K  raid
SCSI2 (0,0,0) (sdb) -- 500.1 GB ATA ST500DM002-1BD14
pri / log 500.1 GB  FREE SPACE

Undo changes to partitions
Finish partitioning and write changes to disk

<Go Back>
```

```

[!!!] Partition disks

How to use this free space:

Create a new partition
Automatically partition the free space
Show Cylinder/Head/Sector information


<Go Back>
```

| **!!! Partition disks** |

The maximum size for this partition is **500.1** GB.

Hint: "max" can be used as a shortcut to specify the maximum size, or enter a percentage (e.g. "20%") to use that percentage of the maximum size.

New partition size:

4% 

<Go Back> <Continue>

| **!!! Partition disks** |

Type for the new partition:

Primary
Logical

<Go Back>

| **!!! Partition disks** |

Please choose whether you want the new partition to be created at the beginning or at the end of the available space.

Location for the new partition:

Beginning
End

<Go Back>

| **!!! Partition disks** |

You are editing partition #1 of SCSI4 (0,0,0) (sdb). No existing file system was detected in this partition.

Partition settings:

Use as: Ext4 journaling file system

Mount point: /
Mount options: defaults
Label: none
Reserved blocks: 5%
Typical usage: standard
Bootable flag: off

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>

```
  [!!!] Partition disks

How to use this partition:

Ext4 journaling file system
Ext3 journaling file system
Ext2 file system
btrfs journaling file system
JFS journaling file system
XFS journaling file system
FAT16 file system
FAT32 file system
swap area
physical volume for encryption
physical volume for RAID
physical volume for LVM
do not use the partition

<Go Back>
```

```
  [!!!] Partition disks

You are editing partition #1 of SCSI4 (0,0,0) (sdb). No existing file system was detected
in this partition.

Partition settings:

Use as:          physical volume for RAID

Bootable flag:  off

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>
```

```
  [!!!] Partition disks

This is an overview of your currently configured partitions and mount points. Select a
partition to modify its settings (file system, mount point, etc.), a free space to create
partitions, or a device to initialize its partition table.

Guided partitioning
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes

SCSI1 (0,0,0) (sda) -- 500.1 GB ATA ST500DM002-1BD14
  #1 primary 20.0 GB   K raid
  #2 primary 480.0 GB B K raid
SCSI2 (0,0,0) (sdb) -- 500.1 GB ATA ST500DM002-1BD14
  #1 primary 20.0 GB   K raid
  pri /log 480.1 GB   FREE SPACE

Undo changes to partitions
Finish partitioning and write changes to disk

<Go Back>
```

```
| [!!] Partition disks |
How to use this free space:
Create a new partition
Automatically partition the free space
Show Cylinder/Head/Sector information

<Go Back>
```

```
| [!!] Partition disks |
The maximum size for this partition is 480.1 GB.
Hint: "max" can be used as a shortcut to specify the maximum size, or enter a percentage
(e.g. "20%") to use that percentage of the maximum size.
New partition size:
480.1 GB
<Go Back> <Continue>
```

```
| [!!] Partition disks |
Type for the new partition:
Primary
Logical

<Go Back>
```

```
| [!!] Partition disks |
You are editing partition #2 of SCSI4 (0,0,0) (sdb). No existing file system was detected
in this partition.
Partition settings:
Use as: Ext4 journaling file system
Mount point: /
Mount options: defaults
Label: none
Reserved blocks: 5%
Typical usage: standard
Bootable flag: off

Copy data from another partition
Delete the partition
Done setting up the partition

<Go Back>
```

```
  [!!!] Partition disks |
|
| How to use this partition:
|
| Ext4 journaling file system
| Ext3 journaling file system
| Ext2 file system
| btrfs journaling file system
| JFS journaling file system
| XFS journaling file system
| FAT16 file system
| FAT32 file system
| swap area
| physical volume for encryption
| physical volume for RAID
| physical volume for LVM
| do not use the partition
|
| <Go Back>
```

```
  [!!!] Partition disks |
|
| You are editing partition #2 of SCSI4 (0,0,0) (sdb). No existing file system was detected
| in this partition.
|
| Partition settings:
|
|           Use as:           physical volume for RAID
|
| Bootable flag: off
|
| Copy data from another partition
| Delete the partition
| Done setting up the partition
|
| <Go Back>
```

```
  [!!!] Partition disks |
|
| You are editing partition #2 of SCSI4 (0,0,0) (sdb). No existing file system was detected
| in this partition.
|
| Partition settings:
|
|           Use as:           physical volume for RAID
|
|           Bootable flag: on
|
| Copy data from another partition
| Delete the partition
| Done setting up the partition
|
| <Go Back>
```

| [!!] Partition disks |

This is an overview of your currently configured partitions and mount points. Select a partition to modify its settings (file system, mount point, etc.), a free space to create partitions, or a device to initialize its partition table.

```

Guided partitioning
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes
    
```

```

SCSI1 (0,0,0) (sda) -- 500.1 GB ATA ST500DM002 - 1BD14
  #1 primary 20.0 GB K raid
  #2 primary 480.0 GB B K raid
SCSI2 (0,0,0) (sdb) -- 500.1 GB ATA ST500DM002 - 1BD14
  #1 primary 20.0 GB K raid
  #2 primary 480.0 GB B K raid
    
```

```

Undo changes to partitions
Finish partitioning and write changes to disk
    
```

<Go Back>

| [!!] Partition disks |

Before RAID can be configured, the changes have to be written to the storage devices. These changes cannot be undone.

When RAID is configured, no additional changes to the partitions in the disks containing physical volumes are allowed. Please convince yourself that you are satisfied with the current partitioning scheme in these disks.

The partition tables of the following devices are changed:

```

SCSI3 (0,0,0) (sda)
SCSI4 (0,0,0) (sdb)
    
```

Write the changes to the storage devices and configure RAID?

<Yes> <No>

| [!!] Partition disks |

This is the software RAID (or MD, "multiple device") configuration menu.

Please select one of the proposed actions to configure software RAID.

Software RAID configuration actions

```

Create MD device
Delete MD device
Finish
    
```

<Go Back>

| [!!] Partition disks |

Please choose the type of the software RAID device to be created.

Software RAID device type:

```

RAID0
RAID1
RAID5
RAID6
RAID10
    
```

<Go Back>

[!] Partition disks

The RAID1 array will consist of both active and spare devices. The active devices are those used, while the spare devices will only be used if one or more of the active devices fail. A minimum of 2 active devices is required.

NOTE: this setting cannot be changed later.

Number of active devices for the RAID1 array:

2

<Go Back> <Continue>

[!] Partition disks

Number of spare devices for the RAID1 array:

0

<Go Back> <Continue>

[!] Partition disks

You have chosen to create a RAID1 array with 2 active devices.

Please choose which partitions are active devices. You must select exactly 2 partitions.

Active devices for the RAID1 array:

[*]	/dev/sda1	(2152MB; raid)
[]	/dev/sda2	(19388MB; raid)
[*]	/dev/sdb1	(2152MB; raid)
[]	/dev/sdb2	(19388MB; raid)

<Go Back> <Continue>

[!] Partition disks

This is the software RAID (or MD, "multiple device") configuration menu.

Please select one of the proposed actions to configure software RAID.

Software RAID configuration actions

Create MD device
Delete MD device
Finish

<Go Back>

[!] Partition disks

Please choose the type of the software RAID device to be created.

Software RAID device type:

RAID0
RAID1
RAID5
RAID6
RAID10

<Go Back>

[!!] Partition disks

The RAID1 array will consist of both active and spare devices. The active devices are those used, while the spare devices will only be used if one or more of the active devices fail. A minimum of 2 active devices is required.

NOTE: this setting cannot be changed later.

Number of active devices for the RAID1 array:

2

<Go Back> <Continue>

[!!] Partition disks

Number of spare devices for the RAID1 array:

0

<Go Back> <Continue>

[!!] Partition disks

You have chosen to create a RAID1 array with 2 active devices.

Please choose which partitions are active devices. You must select exactly 2 partitions.

Active devices for the RAID1 array:

[*]	/dev/sda2	(19388MB; raid)
[*]	/dev/sdb2	(19388MB; raid)

<Go Back> <Continue>

[!!] Partition disks

This is the software RAID (or MD, "multiple device") configuration menu.

Please select one of the proposed actions to configure software RAID.

Software RAID configuration actions

- Create MD device
- Delete MD device
- Finish**

<Go Back>

[!!] Partition disks

This is an overview of your currently configured partitions and mount points. Select a partition to modify its settings (file system, mount point, etc.), a free space to create partitions, or a device to initialize its partition table.

Guided partitioning
 Configure software RAID
 Configure the Logical Volume Manager
 Configure encrypted volumes
 Configure iSCSI volumes

```

RAID1 device #0 -- 20.0 GB Software RAID device
#1 20.0 GB
RAID1 device #1 -- 480.0 GB Linux Software RAID Array
#1 480.0 GB ext3
SCSI1 (0,0,0) (sda) -- 500.1 GB ATA ST500DM002-1BD14
#1 primary 20.0 GB K raid
#2 primary 480.0 GB B K raid
SCSI2 (0,0,0) (sdb) -- 500.1 GB ATA ST500DM002-1BD14
#1 primary 20.0 GB K raid
#2 primary 480.0 GB B K raid
    
```

Undo changes to partitions
 Finish partitioning and write changes to disk

<Go Back>

[!!] Partition disks

You are editing partition #1 of RAID1 device #0. No existing file system was detected in this partition.

Partition settings:

Use as: do not use

Copy data from another partition
 Erase data on this partition
 Done setting up the partition

<Go Back>

[!!] Partition disks

How to use this partition:

Ext4 journaling file system
 Ext3 journaling file system
 Ext2 file system
 btrfs journaling file system
 JFS journaling file system
 XFS journaling file system
 FAT16 file system
 FAT32 file system
 swap area
 physical volume for encryption
 physical volume for LVM
 do not use the partition

<Go Back>

[!] Partition disks

You are editing partition #1 of RAID1 device #0. No existing file system was detected in this partition.

Partition settings:

Use as: swap area

Copy data from another partition
Erase data on this partition
Done setting up the partition

<Go Back>

[!] Partition disks

This is an overview of your currently configured partitions and mount points. Select a partition to modify its settings (file system, mount point, etc.), a free space to create partitions, or a device to initialize its partition table.

Guided partitioning
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes

RAID1 device #0	--	20.0 GB	Software RAID device
#1		20.0 GB	f swap swap
RAID1 device #1	--	480.0 GB	Linux Software RAID Array
#1		480.0 GB	ext3
SCSI1 (0,0,0) (sda)	--	500.1 GB	ATA ST500DM002-1BD14
#1	primary	20.0 GB	K raid
#2	primary	480.0 GB	B K raid
SCSI2 (0,0,0) (sdb)	--	500.1 GB	ATA ST500DM002-1BD14
#1	primary	20.0 GB	K raid
#2	primary	480.0 GB	B K raid

Undo changes to partitions
Finish partitioning and write changes to disk

<Go Back>

[!] Partition disks

You are editing partition #1 of RAID1 device #1. No existing file system was detected in this partition.

Partition settings:

Use as: do not use

Copy data from another partition
Erase data on this partition
Done setting up the partition

<Go Back>

```
[!!!] Partition disks

How to use this partition:

Ext4 journaling file system
Ext3 journaling file system
Ext2 file system
btrfs journaling file system
JFS journaling file system
XFS journaling file system
FAT16 file system
FAT32 file system
swap area
physical volume for encryption
physical volume for LVM
do not use the partition

<Go Back>
```

Vo väčšine návodov na internete, užívateľ používa EXT4. Po konzultácií so skúsenejším kolegom, ja používam EXT3.

```
[!!!] Partition disks

You are editing partition #1 of RAID1 device #1. No existing file system was detected in
this partition.

Partition settings:

Use as:          Ext3 journaling file system
Mount point:    none
Mount options:  defaults
Label:          none
Reserved blocks: 5%
Typical usage:  standard

Copy data from another partition
Erase data on this partition
Done setting up the partition

<Go Back>
```

```
[!!!] Partition disks

Mount point for this partition:

/ - the root file system
/boot - static files of the boot loader
/home - user home directories
/tmp - temporary files
/usr - static data
/var - variable data
/srv - data for services provided by this system
/opt - add-on application software packages
/usr/local - local hierarchy
Enter manually
Do not mount it

<Go Back>
```

```

[!!!] Partition disks

You are editing partition #1 of RAID1 device #1. No existing file system was detected in
this partition.

Partition settings:

      Use as:           Ext3 journaling file system

      Mount point:     /
      Mount options:   defaults
      Label:           none
      Reserved blocks: 5%
      Typical usage:   standard

      Copy data from another partition
      Erase data on this partition
      Done setting up the partition

<Go Back>
    
```

```

[!!!] Partition disks

This is an overview of your currently configured partitions and mount points. Select a
partition to modify its settings (file system, mount point, etc.), a free space to create
partitions, or a device to initialize its partition table.

      Guided partitioning
      Configure software RAID
      Configure the Logical Volume Manager
      Configure encrypted volumes
      Configure iSCSI volumes

      RAID1 device #0 -- 20.0 GB Software RAID device
      . #1          20.0 GB   f swap   swap .
      RAID1 device #1 -- 480.0 GB Linux Software RAID Array
      . #1          480.0 GB   ext3    / .
      SCSI1 (0,0,0) (sda) -- 500.1 GB ATA ST500DM002 - 1BD14
      . #1 primary 20.0 GB   K raid .
      . #2 primary 480.0 GB   B K raid .
      SCSI2 (0,0,0) (sdb) -- 500.1 GB ATA ST500DM002 - 1BD14
      . #1 primary 20.0 GB   K raid .
      . #2 primary 480.0 GB   B K raid .

      Undo changes to partitions
      Finish partitioning and write changes to disk

<Go Back>
    
```

```

[!!!] Partition disks

If you continue, the changes listed below will be written to the disks. Otherwise, you
will be able to make further changes manually.

The partition tables of the following devices are changed:
      RAID1 device #0
      RAID1 device #1

The following partitions are going to be formatted:
      partition #1 of RAID1 device #0 as swap
      partition #1 of RAID1 device #1 as ext3

Write the changes to disks?

      <Yes>                                     <No>
    
```

Táto hláška sa môže objaviť až po inštalácii celého systému

| **[!!] Partition disks** |

If your root filesystem is on a RAID, and a disk is missing at boot, it can either boot with the degraded array, or hold the system at a recovery shell.


Running a system with a degraded RAID could result in permanent data loss if it suffers another hardware fault.

If you do not have access to the server console to use the recovery shell, you might answer "yes" to enable the system to boot unattended.

Do you want to boot your system if your RAID becomes degraded?

<Go Back> **<Yes>** <No>

| **Installing the system...** |


83%


Configuring linux-image-3.11.0-12-generic

| **[!] Configure the package manager** |

If you need to use a HTTP proxy to access the outside world, enter the proxy information here. Otherwise, leave this blank.

The proxy information should be given in the standard form of "http://[[user] [:pass]@]host[:port]/".

HTTP proxy information (blank for none):



<Go Back> <Continue>

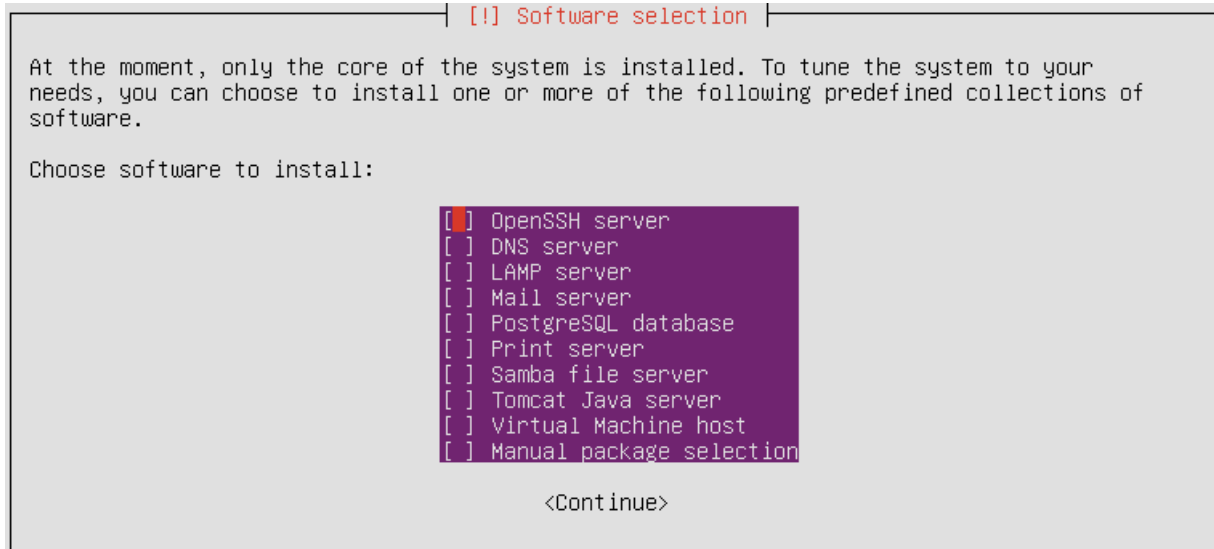
| **[!] Configuring tasksel** |

Applying updates on a frequent basis is an important part of keeping your system secure.

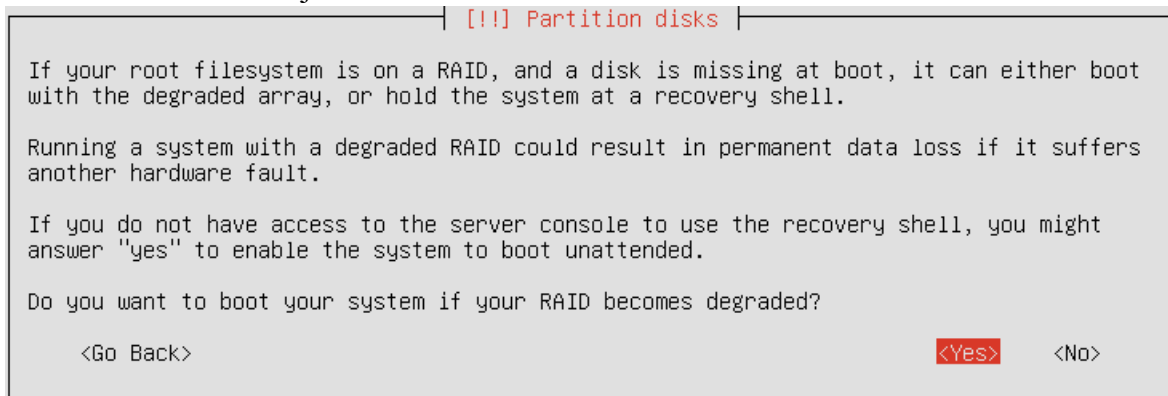
By default, updates need to be applied manually using package management tools. Alternatively, you can choose to have this system automatically download and install security updates, or you can choose to manage this system over the web as part of a group of systems using Canonical's Landscape service.

How do you want to manage upgrades on this system?

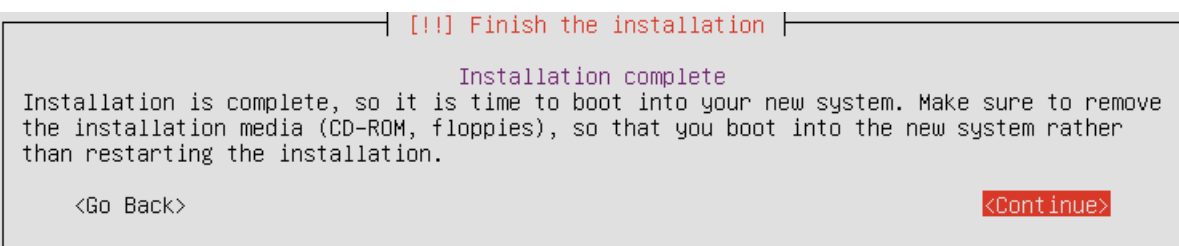
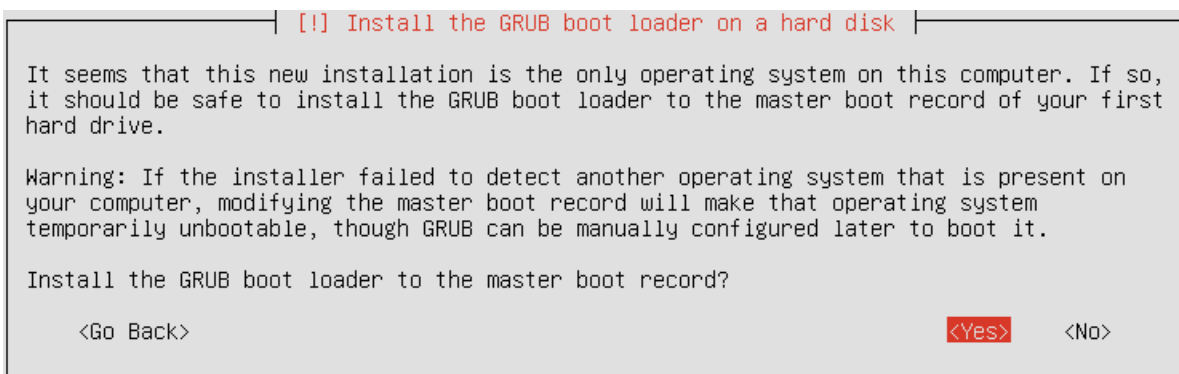
No automatic updates
Install security updates automatically
Manage system with Landscape



Táto hláška sa mohla objaviť už skôr.



Táto hláška sa nemusí objaviť.



3.1. Kontrola RAID1

```
kopera@ubuntu:~$ sudo mdadm --detail /dev/md0
[sudo] password for kopera:
/dev/md0:
    Version : 1.2
  Creation Time : Mon Oct 13 17:26:20 2014
    Raid Level : raid1
    Array Size : 19517440 (18.61 GiB 19.99 GB)
  Used Dev Size : 19517440 (18.61 GiB 19.99 GB)
    Raid Devices : 2
  Total Devices : 2
 Persistence : Superblock is persistent

 Update Time : Mon Oct 27 10:29:49 2014
   State : clean
 Active Devices : 2
 Working Devices : 2
 Failed Devices : 0
 Spare Devices : 0

    Name : ubuntu:0 (local to host ubuntu)
   UUID : 10fec234:9dbd7f67:e6d2b366:bef38426
  Events : 127

   Number   Major   Minor   RaidDevice State
     3         8         1         0   active sync   /dev/sda1
     2         8        17         1   active sync   /dev/sdb1
kopera@ubuntu:~$
```

Riadok „State:clean“ hovorí, že RAID1 je v poriadku

3.2. Výmena HDD v poli RAID1

To, čo popisujem je temer reálna výmena disku. Ja som totiž jeden disk preformátoval a zrušil som rozdelenie partícií aby som mal istotu, že RAID1 naozaj funguje a pri ozajstnom páde ma neprekvapilo, že to potom nebude možné, ak v PC budem mať nastavené a nainštalované programy podľa potreby.

3.2.1. HW simulácia poruchy v poli RAID1

- vypnem PC
- odpojím jeden disk (simulujem tým poškodenie HDD)
- zapnem PC a odskúšam funkčnosť systému (systém by mal fungovať bez náznaku poruchy)
- skontrolujem funkčnosť RAID1

„sudo mdadm --detail /dev/md0“ a


```
„sudo mdadm --detail /dev/md1“
```

```
kopera@ubuntu:~$ sudo mdadm --detail /dev/md0
[sudo] password for kopera:
/dev/md0:
    Version : 1.2
  Creation Time : Mon Oct 13 17:26:20 2014
    Raid Level : raid1
    Array Size : 19517440 (18.61 GiB 19.99 GB)
  Used Dev Size : 19517440 (18.61 GiB 19.99 GB)
    Raid Devices : 2
  Total Devices : 1
  Persistence : Superblock is persistent

    Update Time : Mon Oct 27 10:29:49 2014
      State : clean, degraded
  Active Devices : 1
 Working Devices : 1
 Failed Devices : 0
  Spare Devices : 0

    Name : ubuntu:0 (local to host ubuntu)
   UUID : 10fec234:9dbd7f67:e6d2b366:bef38426
  Events : 127

   Number   Major   Minor   RaidDevice State
    0         0         0         0        removed
    2         8         1         1    active sync   /dev/sda1
kopera@ubuntu:~$
```

Riadok „State:clean, degraded“ hovorí, že RAID1 nie je v poriadku.

Predposledný riadok hovorí, že je zapojený len jeden disk

```
„2         8         1         1    active sync   /dev/sda1“
```

pre porovnanie pozri predošlý výpis.

Podobne to bude vyzerat' aj pre príkaz „sudo mdadm --detail /dev/md1“

- vypnem PC
- na disku, ktorý je pripojený, zruším partície pomocou nejakého diskového manažéra, prípadne pre istotu vytvorím nejaké iné partície aby som zrušil pôvodnú tabuľku rozdelenia partícií. Napríklad časť naformátujem na FAT32 a druhú časť na NTFS
- ak systém nenačítalo, podarilo sa mi zmazať partície
- vymažem partície FAT32 a NTFS. Celý disk nechám bez partície „unallocated“ alebo iného názvu podľa zvyklostí toho, ktorého diskového manažéra
- zapojím disk s funkčným systémom a zapnem PC

3.2.2. Výmena poškodeného HDD v poli RAID1

- ak systém funguje, PC vypnem a zapojím aj druhý (nový) disk

3.2.2.1. Formátovanie nového HDD

- partition manažérom nastavím partície „nového disku“ podľa funkčného disku, použil som GParted (nainštalovala sa mi verzia 0.19.0) takto:
 - „nový“ disk som rozdelil na rovnaké časti ako disk starý na ktorom beží systém. V mojom prípade je to disk „/dev/sdb“ rozdelený na „/dev/sdb1“ veľkosť partície (primárna, ext3) 18,63 GB – SWAP v MB je to číslo 19076, zostatok (lebo aj druhý disk je rovnako veľký) (primárna, ext3) „/dev/sdb2“ ako hlavný pre systém a dokumenty. V oboch prípadoch som ich naformátoval na „ext3“. Zatláč na zelenú fajku na potvrdenie zmien a čakaj! Celý proces trvá niekoľko minút (asi 2 minúty). Skontroluj rozdelenie partícií podľa stĺpca „Size“. Partícia „/dev/sda1 = /dev/sdb1“ a „/dev/sda2 = /dev/sdb2“
 - naformátuj na novom disku oddiel SWAP na „linux-swap“, nenechávaj ho na ext3 alebo ext4.
 - Partíciám priradiť FLAGS takto: novému disku „/dev/sdb1“ priradiť cez „Manage Flags“ RAID. V stĺpci „Flags“ by sa mal objaviť text „raid“. „/dev/sdb2“ priradiť cez „Manage Flags“ RAID a BOOT. V stĺpci „Flags“ by sa mal objaviť text „boot, raid“. Podobne skontroluj, prípadne nastav aj „/dev/sda“

- skontrolujem príkazom „**sudo mdadm --detail /dev/md0**“ funkčnosť RAID1

```
kopera@ubuntu:~$ sudo mdadm --detail /dev/md0
[sudo] password for kopera:
/dev/md0:
```

```
Version : 1.2
Creation Time : Mon Oct 13 17:26:20 2014
Raid Level : raid1
Array Size : 19517440 (18.61 GiB 19.99 GB)
Used Dev Size : 19517440 (18.61 GiB 19.99 GB)
Raid Devices : 2
Total Devices : 1
Persistence : Superblock is persistent

Update Time : Mon Oct 27 10:29:49 2014
State : clean, degraded
Active Devices : 1
Working Devices : 1
Failed Devices : 0
Spare Devices : 0

Name : ubuntu:0 (local to host ubuntu)
UUID : 10fec234:9dbd7f67:e6d2b366:bef38426
Events : 127
```

Number	Major	Minor	RaidDevice	State	
3	8	1	0	active sync	/dev/sda1
2	0	0	2	removed	

- skontrolujem príkazom „**sudo mdadm --detail /dev/md1**“ funkčnosť RAID1

```
kopera@ubuntu:~$ sudo mdadm --detail /dev/md1
/dev/md1:
```

```
Version : 1.2
Creation Time : Mon Oct 13 17:26:24 2014
Raid Level : raid1
Array Size : 468719616 (447.01 GiB 479.97 GB)
Used Dev Size : 468719616 (447.01 GiB 479.97 GB)
```

```

Raid Devices : 2
Total Devices : 1
Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Tue Nov  4 11:58:16 2014
State : clean, degraded
Active Devices : 1
Working Devices : 1
Failed Devices : 0
Spare Devices : 0

Name : ubuntu:1 (local to host ubuntu)
UUID : a67c6961:1a6693c0:ca4a028f:9336e5d2
Events : 6978

Number   Major   Minor   RaidDevice State
  2         8         2         0   active sync   /dev/sda2
  2         0         0         2   removed

kopera@ubuntu:~$

```

3.2.3. Pripojenie nového disku do poľa RAID1

<http://forum.ubuntu.cz/index.php?topic=72814.0>

„`sudo mdadm /dev/md0 -a /dev/sdb1`“ – swap (19.99 GB)

„`sudo mdadm /dev/md1 -a /dev/sdb2`“ – systém a dokumenty (479.97GB)

malo by to skončiť takouto hláškou

```

kopera@ubuntu:~$ sudo mdadm /dev/md0 -a /dev/sdb1
mdadm: added /dev/sdb1
kopera@ubuntu:~$ sudo mdadm /dev/md1 -a /dev/sdb2
mdadm: added /dev/sdb2
kopera@ubuntu:~$

```

Ak sa pozrieš ,na LED kontrolu HDD, na prednom paneli disku mal by si vidieť ako PC pracuje s diskami. Ak si dáš urobiť výpis stavu RAID

`sudo mdadm --detail /dev/md0` alebo aj

`sudo mdadm --detail /dev/md1`

, mal by si uvidieť toto:

```

kopera@ubuntu:~$ sudo mdadm --detail /dev/md0
/dev/md0:
  Version : 1.2
  Creation Time : Mon Oct 13 17:26:20 2014
  Raid Level : raid1
  Array Size : 19517440 (18.61 GiB 19.99 GB)
  Used Dev Size : 19517440 (18.61 GiB 19.99 GB)
  Raid Devices : 2
  Total Devices : 2
  Persistence : Superblock is persistent

  Update Time : Tue Nov  4 14:14:17 2014
  State : clean, degraded, recovering
  Active Devices : 1
  Working Devices : 2

```

```
Failed Devices : 0
Spare Devices : 1
```

Rebuild Status : **99% complete**

```
Name : ubuntu:0 (local to host ubuntu)
UUID : 10fec234:9dbd7f67:e6d2b366:bef38426
Events : 202
```

```
Number Major Minor RaidDevice State
   3     8     1         0 active sync /dev/sda1
   2     8    17         1  spare rebuilding /dev/sdb1
kopera@ubuntu:~$ sudo mdadm --detail /dev/md1
/dev/md1:
```

```
Version : 1.2
Creation Time : Mon Oct 13 17:26:24 2014
Raid Level : raid1
Array Size : 468719616 (447.01 GiB 479.97 GB)
Used Dev Size : 468719616 (447.01 GiB 479.97 GB)
Raid Devices : 2
Total Devices : 2
Persistence : Superblock is persistent
```

Intent Bitmap : Internal

```
Update Time : Tue Nov 4 14:14:38 2014
State : clean, degraded, recovering
Active Devices : 1
Working Devices : 2
Failed Devices : 0
Spare Devices : 1
```

Rebuild Status : **0% complete**

```
Name : ubuntu:1 (local to host ubuntu)
UUID : a67c6961:1a6693c0:ca4a028f:9336e5d2
Events : 7908
```

```
Number Major Minor RaidDevice State
   2     8     2         0 active sync /dev/sda2
   3     8    18         1  spare rebuilding /dev/sdb2
kopera@ubuntu:~$
```

stav kopírovania je v riadku „**Rebuild Status: xx% complete**“

aktuálny priebeh kopírovania si môžeš sledovať cez príkaz

watch cat /proc/mdstat,

mal by vyzeráť podobne

```
Every 2,0s: cat /proc/mdstat
Tue Nov 4 14:24:50 2014
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5]
[raid4] [raid10]
md0 : active raid1 sdc1[2] sdb1[0]
625129216 blocks [2/1] [U_]
[>.....] recovery = 0.3% (2205312/625129216)
finish=51.2min speed=126375K/sec
unused devices: <none>
```

v mojom prípade to kopírovanie trvalo asi 1 hod.

3.2.4. Kontrola RAIDu

Príkaz I.

```
sudo mdadm --detail /dev/md0 a
```

```
sudo mdadm --detail /dev/md1
```

```
kopera@ubuntu:~$ sudo mdadm --detail /dev/md0
[sudo] password for kopera:
/dev/md0:
    Version : 1.2
  Creation Time : Mon Oct 13 17:26:20 2014
    Raid Level : raid1
    Array Size : 19517440 (18.61 GiB 19.99 GB)
  Used Dev Size : 19517440 (18.61 GiB 19.99 GB)
  Raid Devices : 2
  Total Devices : 2
 Persistence : Superblock is persistent

 Update Time : Tue Nov  4 14:14:28 2014
   State : clean
 Active Devices : 2
 Working Devices : 2
 Failed Devices : 0
 Spare Devices : 0

    Name : ubuntu:0 (local to host ubuntu)
   UUID : 10fec234:9dbd7f67:e6d2b366:bef38426
  Events : 204

    Number Major Minor RaidDevice State
       3      8      1        0  active sync  /dev/sda1
       2      8     17        1  active sync  /dev/sdb1
kopera@ubuntu:~$ sudo mdadm --detail /dev/md1
/dev/md1:
    Version : 1.2
  Creation Time : Mon Oct 13 17:26:24 2014
    Raid Level : raid1
    Array Size : 468719616 (447.01 GiB 479.97 GB)
  Used Dev Size : 468719616 (447.01 GiB 479.97 GB)
  Raid Devices : 2
  Total Devices : 2
 Persistence : Superblock is persistent

 Intent Bitmap : Internal

 Update Time : Wed Nov  5 06:52:13 2014
   State : clean
 Active Devices : 2
 Working Devices : 2
 Failed Devices : 0
 Spare Devices : 0

    Name : ubuntu:1 (local to host ubuntu)
   UUID : a67c6961:1a6693c0:ca4a028f:9336e5d2
  Events : 9000

    Number Major Minor RaidDevice State
```

```

      2      8      2      0      active sync  /dev/sda2
      3      8     18      1      active sync  /dev/sdb2
kopera@ubuntu:~$

```

V obidvoch prípadoch je v riadku „**State: clean**“, čo svedčí o funkčnosti RAID1. Ak si pozrieš kapitolu 3.1 na strane 25, nájdeš tam rovnaký stav ako po výmene HDD, čo je dobrá známka a ja môžem pokračovať v dokončení konfigurácie celého RAID1. poznámka: nepovažujem sa ešte za odborníka na linux, ale zrejme swap nie je nutné mirrorovať. Pripojil som ho do RAID1 len na skúšku. Oddiel pre SWAP je v každom prípade nutný na obidvoch diskoch!

Príkaz II.

```
cat /proc/mdstat
```

ak je všetko v poriadku mal by si vidieť niečo podobné

```

kopera@ubuntu:~$ cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5]
[raid4] [raid10]
md1 : active raid1 sdb2[3] sda2[2]
      468719616 blocks super 1.2 [2/2] [UU]
      bitmap: 0/4 pages [0KB], 65536KB chunk

md0 : active raid1 sdb1[2] sda1[3]
      19517440 blocks super 1.2 [2/2] [UU]

unused devices: <none>
kopera@ubuntu:~$

```

3.2.5. Obnova – oprava BOOTLOADERa

Počítač reštartuj a v BIOSe nastav bootovanie raz z jedného disku potom z druhého. Ak počítač nebutuje z jediného z nich, je treba (v mojom prípade) obnoviť BOOTLOADER.

md1 – je RAID pre boot a systém

sdb2 – je časť novopripojeného disku, kde mám systém a dokumenty a malo by sa dať s neho aj nabudovať

preto

```
grub-install /dev/sdb
```

```

kopera@ubuntu:~$ sudo grub-install /dev/sdb
Installing for i386-pc platform.
Installation finished. No error reported.

```

```
update-grub
```

```

kopera@ubuntu:~$ sudo update-grub
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-3.16.0-23-generic
Found initrd image: /boot/initrd.img-3.16.0-23-generic

```

```

Found linux image: /boot/vmlinuz-3.16.0-20-generic
Found initrd image: /boot/initrd.img-3.16.0-20-generic
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
done
kopera@ubuntu:~$

```

Funkčnosť otestujem tak, že v BIOSe nastavím butovanie raz z jedného, potom z druhého disku. V obidvoch prípadoch musí systém fungovať bez problémov. Ak máš čas otestovať to aj hardvérovo, odpoj raz jeden potom druhý disk. Pozor, ale zakaždým musíš dať pripojiť odpojený disk do RAIDu. Teda musíš zopakovať kapitolu 3.2.3 na strane 28. Formátovať ho nemusíš. V tom prípade bude kopírovanie dáť oveľa rýchlejšie. Mne to na systéme bez nainštalovaných ďalších aplikácií trvalo asi 15 minút.

3.3. Automatická kontrola RAID1

Detailný popis tohoto postupu nájdeš v tomto dokumente v prílohe na strane 46.

3.3.1. Vytvorenie skriptov

Inštalácia programu na posielanie e-mailu.

```
sudo apt-get install sendmail
```

V „Home“ (môže byť aj iné miesto na systémovom disku) skript „rs_rmk1.sh“

(v pôvodnom dávode

<http://ubuntuhow2.blog.zive.cz/2009/01/softwareve-raid-pole-1-dil/>

alebo pozri dole nižšie prílohu na strane 42, sa hovorí, že by to malo ísť, mne to však nešlo, fungovalo to len z adresára HOME).

```
sudo gedit rs_rmk1.sh
```

otvorí sa nám okno, kde napíš alebo skopíruj tento kód, prípadne ho uprav podľa potreby. Pravdepodobne budeš písať **md0** namiesto **md1**.

```
#!/bin/bash
/sbin/mdadm --detail /dev/md1
```

Skript ulož a skontroluj otvorením, či príkaz nie je náhodne zmenený. Na ikonke by si mal vydieť zamknutý visiaci zámok.

Súbor nastavím ako spúšťač s právami „root“

```
sudo chmod 777 rs_rmkl.sh
```

– od tejto chvíle je spúšťačí

```
sudo chown root rs_rmkl.sh
```

– mne príkaz fungoval po reštarte aj keď
nemal práva ROOT ???

```
kopera@ubuntu:~$ sudo chmod 777 rs_rmkl.sh
[sudo] password for kopera:
kopera@ubuntu:~$
```

či je súbor spúšťačí a kto ho „vlastní“ overím príkazom

```
ls -l
kopera@ubuntu:~$ ls -l
total 80
-rwxr-xr-x 1 kopera kopera 575 Oct 29 12:25 1_raidmonitor.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 29 10:42 Desktop
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Documents
drwxr-xr-x 4 kopera kopera 4096 Oct 27 08:50 Downloads
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Music
drwxr-xr-x 2 kopera kopera 4096 Oct 21 13:29 Pictures
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Public
-rwxrwxrwx 1 root root 830 Nov 5 15:01 raidmonitor_rmkl.sh
-rwxrwxrwx 1 kopera kopera 825 Nov 5 15:01 raidmonitor_rmkl.sh~
-rwxrwxrwx 1 kopera kopera 810 Oct 29 16:58 raidmonitor.sh
-rwxrwxrwx 1 kopera kopera 809 Oct 29 16:58 raidmonitor.sh~
-rwxrwxrwx 1 root root 42 Nov 5 14:48 raid_status_rmkl.sh
-rw-rw-r-- 1 kopera kopera 42 Nov 5 13:22 raid_status_rmkl.sh~
-rwxrwxrwx 1 root kopera 59 Nov 5 13:01 raid_status.sh
-rwxrwxr-x 1 kopera kopera 53 Nov 5 13:01 raid_status.sh~
-rw-r--r-- 1 root root 794 Nov 5 15:23 rm_rmkl.sh
-rwxrwxrwx 1 root root 42 Nov 5 15:20 rs_rmkl.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Templates
-rw-rw-r-- 1 root kopera 6 Oct 29 15:50 test.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Videos
kopera@ubuntu:~$
```

je spustiteľný.

Overím to ak do terminálu napíšem

```
sudo ./rs_rmkl.sh
```

malo by sa mi objaviť presne to, ak by som napísal príkaz

```
sudo mdadm --detail /dev/md1
```

Otvorím /etc/sudoers

```
sudo gedit /etc/sudoers
```

na koniec pridám riadok

```
prihlasovacie-meno ALL=NOPASSWD:/cesta-k-súboru/rs_rmkl.sh
```

ten riadok zaistí, že od nás nebude zakaždým vyžadovať heslo

V mojom prípade vyzerá celý súbor takto:

```
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
```



```

#
# See the man page for details on how to write a sudoers file.
#
Defaults env_reset
Defaults mail_badpass
Defaults
    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/s
bin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin  ALL=(ALL)  ALL

# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d
kopera ALL=NOPASSWD:/home/rs_rmkl.sh

```

Teraz napíšem skript, ktorý bude v rovnakom adresáry ako **rs_rmkl.sh**. Bude využívať prvý skript a zisťovať, či je všetko v poriadku a navyše odošle e-mail s varovaním.

```
sudo gedit rm_rmkl.sh
```

Otvorí sa mi súbor a tam vložím (uprav si to podľa potreby) tento text – skript (príkaz)

```

#!/bin/bash
get_Rs_rmkl()
{
sudo ./rs_rmkl.sh | egrep -i "faulty|degraded|clean" | wc -l
}
send_mail()
{
FILENAME="/tmp/raid_report_`date +%d_%m_%Y_%T`.txt"
sudo ./rs_rmkl.sh > $FILENAME
sendEmail \
-f radoslav.kopera@centrum.sk \
-t koperator@pobox.sk \
-cc koperar@soseza.sk \
-u RAID_FUJITSU_TX100_kopera \
-m NASTALI PROBLEMY S VASIM DISKOVYM POLOM!! \
"

```

kontaktuj prosim uzivatela:

```

Michal Michalovic Michalov
Zilina - xxxxxxxxxx
Ulica - Stred 007
045 21
tel. xxxx 888 888; 041 / xxxx xxx
rmk_1477@centrum.sk
PC vo WC na UPC
Fujitsu-Siemens TS100 S3

```

```

" \
Byl vygenerovan report. \
V prilozke je vypis z `date +%d/%m/%Y_%T`. \
\
-s smtp.centrum.sk \
-o tls=auto \
-xu prihlasovacie_meno_pre_smtp_server \
-xp heslo_pre_smtp_server \
-a $FILENAME
}

```

```

RAID_STATUS=`get_Rs_rmkl`
if [ $RAID_STATUS -ne 0 ];then
send_mail
fi

```

ak ti bude fungovať odosielanie e-mailov potom z riadku

```
sudo ./rs_rmkl.sh | egrep -i "faulty|degraded|clean" | wc -l
```

odstráň „|clean“! Inak ti budú chodiť e-mailly neustále lebo RAID je v poriadku v pri spustení skriptu bude vždy splnená podmienka "faulty|degraded|clean". Potom ti príde e-mail len ak sa ti poškodí RAID. Vtedy nastane zhoda podmienky "faulty|degraded".

Či je súbor (skript) spúšťač overím príkazom

```
ls -l
```

```

kopera@ubuntu:~$ ls -l
total 80
-rwxr-xr-x 1 kopera kopera 575 Oct 29 12:25 1_raidmonitor.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 29 10:42 Desktop
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Documents
drwxr-xr-x 4 kopera kopera 4096 Oct 27 08:50 Downloads
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Music
drwxr-xr-x 2 kopera kopera 4096 Oct 21 13:29 Pictures
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Public
-rwxrwxrwx 1 root root 830 Nov 5 15:01 raidmonitor_rmkl.sh
-rwxrwxrwx 1 kopera kopera 825 Nov 5 15:01 raidmonitor_rmkl.sh~
-rwxrwxrwx 1 kopera kopera 810 Oct 29 16:58 raidmonitor.sh
-rwxrwxrwx 1 kopera kopera 809 Oct 29 16:58 raidmonitor.sh~
-rwxrwxrwx 1 root root 42 Nov 5 14:48 raid_status_rmkl.sh
-rw-rw-r-- 1 kopera kopera 42 Nov 5 13:22 raid_status_rmkl.sh~
-rwxrwxrwx 1 root kopera 59 Nov 5 13:01 raid_status.sh
-rwxrwxr-x 1 kopera kopera 53 Nov 5 13:01 raid_status.sh~
-rw-r--r-- 1 root root 794 Nov 5 15:23 rm_rmkl.sh
-rwxrwxrwx 1 root root 42 Nov 5 15:20 rs_rmkl.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Templates

```

```
-rw-rw-r-- 1 root kopera 6 Oct 29 15:50 test.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Videos
kopera@ubuntu:~$
```

skript nie je spúšťací. Ak by bol spúšťací na začiatku riadku by si našiel „-rwxrwxrwx“, ale ja mám „-rw-r--r--“.

Nastavím ho ako spúšťací príkazom

```
sudo chmod 777 rm_rmkl.sh
```

a opäť pozriem, či je spúšťací príkazom

```
ls -l
```

```
kopera@ubuntu:~$ ls -l
total 80
-rwxr-xr-x 1 kopera kopera 575 Oct 29 12:25 1_raidmonitor.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 29 10:42 Desktop
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Documents
drwxr-xr-x 4 kopera kopera 4096 Oct 27 08:50 Downloads
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Music
drwxr-xr-x 2 kopera kopera 4096 Oct 21 13:29 Pictures
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Public
-rwxrwxrwx 1 root root 830 Nov 5 15:01 raidmonitor_rmkl.sh
-rwxrwxrwx 1 kopera kopera 825 Nov 5 15:01 raidmonitor_rmkl.sh~
-rwxrwxrwx 1 kopera kopera 810 Oct 29 16:58 raidmonitor.sh
-rwxrwxrwx 1 kopera kopera 809 Oct 29 16:58 raidmonitor.sh~
-rwxrwxrwx 1 root root 42 Nov 5 14:48 raid_status_rmkl.sh
-rw-rw-r-- 1 kopera kopera 42 Nov 5 13:22 raid_status_rmkl.sh~
-rwxrwxrwx 1 root kopera 59 Nov 5 13:01 raid_status.sh
-rwxrwxr-x 1 kopera kopera 53 Nov 5 13:01 raid_status.sh~
-rwxrwxrwx 1 root root 794 Nov 5 15:23 rm_rmkl.sh
-rwxrwxrwx 1 root root 42 Nov 5 15:20 rs_rmkl.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Templates
-rw-rw-r-- 1 root kopera 6 Oct 29 15:50 test.sh
drwxr-xr-x 2 kopera kopera 4096 Oct 14 06:19 Videos
kopera@ubuntu:~$
```

Funkčnosť overím príkazom

```
sudo ./rm_rmkl.sh
```

```
kopera@ubuntu:~$ sudo ./rm_rmkl.sh
Nov 05 16:09:08 ubuntu sendEmail[7144]: Email was sent successfully!
kopera@ubuntu:~$
```

!!!

Ak by ti skript nešiel, ale si si istý, že je už overený a fungoval skús nakopírovať celý text cez „gedit“ alebo iný textový editor. Môže sa stať, že už nachystaný skript nebude z nejakého dôvodu funkčný

!!!

O niekoľko sekúnd mi prišiel e-mail s týmto výpisom

```

NASTALI          PROBLEMY          S          VASIM          DISKOVYM          POLOM!!
kontaktuj                prosim                uzivatela:
xxxxxxx                xxxxxx                xxxx
xxxxxxx                -                xxxxx
Pazite          -                Stred                645
011                11
1232 123 10; 041 / 000 000123
asd@asd.asd.sk
PC          pod                stolom                v                dielni
Fujitsu-Siemens                TS100                S3

```

Byl vygenerovan report. V prilozke je vypis z 05/11/2014_16:09:08.

V prílohe bol TXT súbor s týmto výpisom

```

/dev/md1:
  Version : 1.2
  Creation Time : Mon Oct 13 17:26:24 2014
  Raid Level : raid1
  Array Size : 468719616 (447.01 GiB 479.97 GB)
  Used Dev Size : 468719616 (447.01 GiB 479.97 GB)
  Raid Devices : 2
  Total Devices : 2
  Persistence : Superblock is persistent

  Intent Bitmap : Internal

  Update Time : Wed Nov 5 16:08:15 2014
  State : clean
  Active Devices : 2
  Working Devices : 2
  Failed Devices : 0
  Spare Devices : 0

  Name : ubuntu:1 (local to host ubuntu)
  UUID : a67c6961:1a6693c0:ca4a028f:9336e5d2
  Events : 9000

  Number Major Minor RaidDevice State
    2      8      2      0      active sync  /dev/sda2
    3      8     18      1      active sync  /dev/sdb2

```

3.3.2. Časové spúšťanie kontroly RAID1

Na pravidelné spúšťanie som nainštaloval program CRON. Podrobnejší popis a manipuláciu pozri v prílohe na strane 61. Ja som na skúšku zadefinoval najprv spúšťanie programu každú minútu. Počítač som reštartoval ale neprihlasoval som sa. Fungovalo to vinikajúco. Ja som mal nainštalované UBUNTU SERVER 14.10.

Nastavenie je takéto:

V termináli (príkazovom riadku) napíšem príkaz

```
sudo crontab -e
```

na samý spodok toho okna som napísal

```
*/1 * * * * (cd /home/kopera; ./rm_rmk1.sh)
```

!!! medzi hviezdičkami je medzera !!!

stlačil som „CTRL+X“ a potom „Y“ potom „ENTER“

výpis po úspešnej konfigurácii:

```
kopera@ubuntu:~$ sudo crontab -e
crontab: installing new crontab
kopera@ubuntu:~$
```

prípadne podobný výpis.

Je nutné ešte reštartovať CRON.

```
sudo invoke-rc.d cron restart
```

```
kopera@ubuntu:~$ sudo invoke-rc.d cron restart
cron stop/waiting
cron start/running, process 4453
kopera@ubuntu:~$
```

Výpis úloh môžeš urobiť pre potreby kontroly príkazom

```
sudo crontab -l
```

Ak chceš na skúšku aby to posielalo každých 5 minút napíš:

```
*/5 * * * * (cd /home/kopera; ./rm_rmk1.sh)
```

namiesto

```
*/1 * * * * (cd /home/kopera; ./rm_rmk1.sh)
```

Detaily pozri v prílohe.

Doterajšie skúsenosti s CRON.

Ak si naplánuješ rôzne úlohy a v pláne sa majú vykonávať viaceré úlohy, daj si pozor aby sa v rovnakú minútu nevykonali viac než jedna úloha. Spustí vždy len jednu z nich – samozrejmä tú prvú, ktorá sa má vykonať a potom ďalšie úlohy jednoducho preskočí.

Ak sa má prvá úloha vykonávať napríklad každých 9 minút a druhá každých 7 minút vždy berie vykonávanie úlohy od nultej minúty v hodine. V mojom prípade sa

vykoná prvá úloha, ostatné sa potom vykonávajú v danom čas, teda druhá úloha sa vykoná v 7 minúte dajej hodiny. Potom sa v 9-tej minúte vykoná úloha prvá, 14-tej druhá, 18-tej prvá a tak ďalej až kým nepríde nultá minúta hodiny. Program neberie do úvahy, že sa prvá úloha spustila v 59-tej minúte. Spustí ju opäť v 0-tej minúte lebo je prvá a ostatné preskočí. Ak sa druhá úloha vykonala v 54-tej minúte, nevykoná sa v 1-vej minúte ale až v 7-mej.

4 0 * * * * tento zápis znamená, že sa úloha bude spúšťať každý deň 4 minúty po polnoci

*/4 * * * * tento zápis znamená, že sa úloha bude spúšťať každé 4 minúty

* */4 * * * * aj tento zápis znamená, že sa úloha bude spúšťať každé 4 minúty

Ak si zadefinuješ, že sa má spúšťať nejaký skript po 69 minútách a po 78 minútách tak obidva sa budú spúšťať každú hodinu. Pozor ale obidva sa spustia jeden po druhom nie len jeden ako je to popísané v druhom odstavci.

4. Užívateľské účty

4.1. Vytvorenie užívateľského účtu s právami – USER

Vytvorím užívateľský účet „rmk“ s heslom „aaa“

Do terminálu (príkazového riadka) napíš:

```
sudo adduser rmk
```

2x si vypíta si heslo pre nového užívateľa

```
aaa  
aaa
```

Potom vypíš podľa potreby údaje o užívateľovi.

Full Name:

Room Number:

Work Phone:

Home Phone:

Other:

Is the information correct? [Y/n]

!!! užívateľ sa nemôže dostať do práv ROOT !!!

4.2. Vytvorenie užívateľského účtu s právami – ROOT

<http://forum.ubuntu.cz/index.php?topic=73039.msg526906#msg526906>

19.XI.2014

Ak chcem vytvoriť nového užívateľa a má patriť do skupiny odkiaľ sa dá prihlásiť aj správami ROOT,

napíšem to takto:

najprv sa prepnem do ROOT módu

```
"kopera@ubuntu: ~$ sudo su"
```

potom napíšem príkaz na vytvorenie nového užívateľa patriaceho do skupiny "sudo"

```
"root@ubuntu:/home/kopera# useradd rmk7 -G sudo"
```

vytvoril som užívateľa "rmk7" patriaceho do skupiny "Administrator".

Ak sa pozriem cez "System Setting / User Accounts / rmk7"
V riadku "Account Type" uvidím "Administrator"

P.S.

Fungovalo mi to aj s prepínačom "-g"

5. Literatúra a prílohy

Rôzne po internete som ponachádzal zaujímavé články. Niektoré boli tak perfektne spracované, že by bola škoda uviesť len odkaz. V prípade vymazania zo servera by sa čitateľ nedozvedel aj iné zaujímavosti, ktoré som z danej literatúry nepoužil. Preto v tejto časti okrem odkazu, kde je možné článok nájsť aj s prípadnými ďalšími komentármi, dávam k dispozícii aj kópiu článkov, ktoré som použil ako začínajúci užívateľ.

5.1. Softwarové RAID pole - 1. díl

9. 1. 2009, afk_cz

<http://ubuntuhow2.blog.zive.cz/2009/01/softwareve-raid-pole-1-dil/>



Běžný domácí uživatel většinou pozná cenu svých dat, až když o ně přijde. Dnes se po díváme na jednu z mnoha možností bezpečnější práce s daty, konkrétně na vytvoření tzv. RAIDu.

V 1. dílu budeme přidávat do systému další disky a vytvářet diskové pole pouze na nich.

5.1.1. Úvodem

Pokud patříte k těm, kteří toto slovo slyší poprvé, pak vězte, že RAID znamená Redundant Array of Independent Disks, neboli vícenásobné diskové pole nezávislých disků. Komplexní popis v češtině naleznete například na [Wikipedii](#).

Zkráceně lze říci, že data jsou uložena na několika pevných discích a je jen otázkou nastavení, zda se více disků tváří jako jeden, nebo se jednoduše kapacita posčítá do jednoho velkého apod. My se budeme zabývat pro nás nejlevnějším a z mého pohledu nejefektivnějším řešením, a to **softwarovým RAIDem**. Nejlevnějším proto, protože odpadá nákup nějakého hardwarového řadiče v řádu tisíců Kč, nejefektivnějšího proto, protože můžeme celé pole pohodlně přenést do jiného počítače bez nutnosti de/montáže řadiče a navíc se nemusíme bát poruchy řadiče samotného. Narozdíl od hardwarového řadiče je u software řešení využíváno v dnešní době již zanedbatelné procento procesorového výkonu. Softwarové řešení má ale ještě jedno plus, které u hardwarového nenajdete - můžete do raidu přidávat i zařízení běžící na různých řadičích. Málokdo toto asi využije, ale představte si diskové pole, kdy do něj máte zapojeny najednou SATA, IDE, USB a E-SATA disky, nebo dokonce disky někde na síti či internetu. Těžko si představit takový HW řadič, který by toto zvládl.

Ještě před samotným návodem bych rád uvedl na pravou míru, že u drtivé většiny základních desek, u kterých výrobce uvádí, že obsahují RAID 0,1,0+1 apod., se nejedná o pravý HW RAID, ale tzv. Fake-RAID (falešný RAID), který spočívá v tom, že BIOS počítače ve spolupráci s ovladači a řadičem (neobsahujícím RAID čip) vytvoří diskové pole. Čili jde v podstatě opět o SW RAID. Dost ale teorie, podíváme se na konkrétní řešení.

5.1.2. Příprava

V našem modelovém případě budeme uvažovat takto: máme již operační systém nainstalovaný a chystáme se přidat 2 stejně velké SATA disky, kde budeme mít pouze jeden velký oddíl na ukládání pro nás cenných dat. Volba padla na zrcadlení, čili RAID 1 -> oba disky budou obsahovat totožná data a na venek budou tedy vystupovat jako jeden. V případě pádu jednoho z disků můžeme udělat za chodu výměnu (díky SATA hot-plug, rozhodně nedělat u IDE/PATA disků!) a nechat data z funkčního disku opět sesynchronizovat na nový.

Uvažujeme tedy, že systémový disk bude v našem případě jako `/dev/sda`, další disky po přidání do PC budou vidět jako `/dev/sdb` a `/dev/sdc`. Toto lze zjistit například pomocí programu [gparted](#). V Ubuntu ho najdeme pod *Systém->Správa->Editor diskových oddílů*, nebo pro jednoduchý výpis v terminálu zadáme

```
sudo fdisk -l
```

Teď když víme, o které disky v systému jde, můžeme začít s prací.

5.1.3. Vytvoření nového RAID disku

Z repozitářů nainstalujeme balík `mdadm`

```
sudo apt-get install mdadm
```

Spustíme `gparted` a na obou discích vytvoříme stejně velký primární oddíl a naformátujeme je také stejně, doporučuji na [Ext3](#). Poté nastavíme oběma oddílům příznak *Raid*, čímž systému řekneme, že bude tento oddíl použit v raidu (pravé tlačítko myši na vytvořeném oddílu->Nastavit příznaky->Raid). Ukončíme `gparted`.

Vytvoříme RAID 1 - v terminálu zadáme

```
sudo mdadm -v --create /dev/md0 -l 1 -n 2 /dev/sdb1 /dev/sdc1
```

<code>-v</code>	...	upovídaný	výpis
<code>--create</code>	<code>/dev/md0</code>	...	vytvoří virtuální raidový disk md0
<code>-l</code>	<code>1</code>	...	RAID 1
<code>-n 2</code>	...	počet zařízení	

Měla by ihned začít synchronizace obou disků. Průběh si můžeme prohlédnout v terminálu

```
watch cat /proc/mdstat
```

Po ukončení synchronizace máme v systému nově nenaformátovaný disk `/dev/md0`. Provedeme formát na `ext3` buďto v `gparted`, nebo v terminálu pomocí

```
sudo mke2fs -j /dev/md0
```

5.1.4. Připojení RAIDu

Nový disk můžeme nyní kamkoliv připojit, například do `/media/raid`

...vytvoříme adresář

```
sudo mkdir /media/raid
```

a pripojíme do něj disk

```
sudo mount /dev/md0 /media/raid
```

Změníme vlastníka adresáře na nás

```
sudo chown náš-login:naše-skupina /media/raid
```

Nyní již můžeme náš nový RAID 1 používat. Zautomatizujeme si pripojování RAIDu po každém startu počítače, v terminálu

```
sudo gedit /etc/fstab
```

a přidáme řádek

```
/dev/md0          /media/raid      defaults          0          0
```

5.1.5. Manuální kontrola stavu

Diskové pole lze kontrolovat například v terminálu

```
sudo mdadm --detail /dev/md0
```

ve výpisu by se nám mělo objevit, pokud jsou disky v pořádku, něco takového

```
/dev/md0:
  Version : 00.90.03
  Creation Time : Wed Sep  3 22:28:14 2008
  Raid Level : raid1
  Array Size : 625129216 (596.17 GiB 640.13 GB)
  Used Dev Size : 625129216 (596.17 GiB 640.13 GB)
  Raid Devices : 2
  Total Devices : 2
  Preferred Minor : 0
  Persistence : Superblock is persistent

  Update Time : Thu Jan  8 07:42:01 2009
  State : clean
  Active Devices : 2 (...aktivní disky RAIDu)
  Working Devices : 2 (...z toho funkčních)
  Failed Devices : 0 (...z toho nefunkčních)
  Spare Devices : 0 (...počet záložních)

  UUID : 90136de2:2366fce0:01f9e43d:ac30fbff
  Events : 0.26

   Number   Major   Minor   RaidDevice State
    -----   -----   -----   -
    0         8       17         0   active sync  /dev/sdb1
    1         8       33         1   active sync  /dev/sdc1
```

5.1.6. Simulace poruchy disku

Pokud jsou disky v poli zdravé, můžeme vyzkoušet fingovaný pád /dev/sdc1, abychom si otestovali, jak to může vypadat někdy v reálu.

```
sudo mdadm --manage --set-faulty /dev/md0 /dev/sdc1
```

Ověříme status pole

```
sudo mdadm --detail /dev/md0
```

a dostáváme už hlášení

```

/dev/md0:
Version : 00.90.03
Creation Time : Wed Sep  3 22:28:14 2008
Raid Level : raid1
Array Size : 625129216 (596.17 GiB 640.13 GB)
Used Dev Size : 625129216 (596.17 GiB 640.13 GB)
Raid Devices : 2
Total Devices : 2
Preferred Minor : 0
Persistence : Superblock is persistent

Update Time : Thu Jan  8 21:10:06 2009
State : clean, degraded
Active Devices : 1
Working Devices : 1
Failed Devices : 1
Spare Devices : 0

UUID : 90136de2:2366fce0:01f9e43d:ac30fbff
Events : 0.28

Number   Major    Minor    RaidDevice State
0         8        17       0         active sync   /dev/sdb1
1         0         0        1         removed
2         8        33       -         faulty spare /dev/sdc1

```

5.1.7. Oprava poškozeného pole

Nejprve je třeba z diskového pole odebrat poškozený disk

```
sudo mdadm /dev/md0 -r /dev/sdc1
```

RAID nám tedy běží v degradovaném režimu pouze s jedním diskem. Jakoby vyměníme disk za nový a je tedy třeba ho znovu přidat do RAIDu

```
sudo mdadm /dev/md0 -a /dev/sdc1
```

Dostáváme zprávu o znovupřidaném disku a začíná automaticky probíhat synchronizace ze zdravého disku na nově přidaný. Na průběh se můžeme podívat přes

```
watch cat /proc/mdstat
```

kde dostaneme něco takového

```

Every 2,0s: cat /proc/mdstat                                     Thu
Jan  8 21:24:50 2009
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4]
[raid10]
md0 : active raid1 sdc1[2] sdb1[0]
625129216 blocks [2/1] [U_]
[>.....] recovery = 0.3% (2205312/625129216) finish=108.2min
speed=95883K/sec

```

unused devices: <none>

... jednoduché, že ?

5.1.8. Poškození dat

Bohužel pokud není poškozen superbloc, RAID sám o sobě nepozná a ani nemůže poznat, zda jsou nějaká data poškozena. Nebyl tak ani navržen. Nehraje vůbec roli, zda se jedná o softwarový či hardwarový RAID. Pokud tedy neselže samotný zápis na disk, nelze nic poznat. RAID nehlídá integritu dat, RAID nám umožní zachovat data při pádu disku.

5.1.9. Automatické monitorování stavu

V této části si stručně popíšeme, jak stav pole nechat monitorovat automaticky systém a v případě pádu si o tom navíc necháme poslat informační email.

Pokud nám na počítači běží již nakonfigurovaný emailový server, můžeme pro odesílání stavu pole použít například

```
sudo mdadm --monitor --mail=muj@mail.cz --delay=1800 /dev/md0
```

Pokud emailový server nemáme, stačí si nainstalovat jednoduchý malý prográmeček pro posílání e-mailů z terminálu

```
sudo apt-get install sendemail
```

a vyzkoušíme sami sobě poslat e-mail s předmětem Raid_Report, do kopie přidáme nějakou druhou emailovou adresu, tělo zprávy bude Test, k odeslání použijeme [SMTP server](#), například Vašeho poskytovatele internetu. Osobně jsem použil smtp.upcmail.cz od mého poskytovatele a nebyla vyžadována autentizace. Naproti tomu iol.cz toto vyžaduje.

```
sendEmail -f muj_email@muj_email.cz -t muj_email@muj_email.cz -cc
muj_email_2@muj_email_2.cz
-u Raid_Report -m Test -s smtp.vasposkytovatelpripojeni.cz -o tls=auto
```

...vše je jeden příkaz, musel jsem ho rozdělit

Doporučuji použít SMTP server buďto poskytovatele připojení, nebo nějaký veřejný, kde máme například emailovou schránku. V druhém případě ale dostaneme hlášku, že je vyžadována autentizace. Tu lze zautomatizovat pomocí dalšího parametru `-xu <vaše-jméno>` a pro heslo `-xp <vaše-heslo>`. Není to ale ideální, protože potom budeme muset do skriptu uložit i heslo v čitelné podobě, což není bezpečné.

Až zvolíme funkční volbu a email je úspěšně odeslán, můžeme přejít k napsání skriptu. Ten nám zjistí stav pole, v případě poruchy uloží výpis do duboru s aktuálním datem a časem, navíc vše odešle emailem s přílohou.

Nejprve vytvoříme někde na systémovém disku nový skript, který nám zjistí stav pole.

```
sudo gedit raid_status.sh
```

a vložíme do něj následující kód

```
#!/bin/bash
/sbin/mdadm --detail /dev/md0
```

Po uložení soubor nastavíme jako spouštěcí s příslušnými právy

```
sudo chown root raid_status.sh
```

Otevřeme /etc/sudoers

```
sudo gedit /etc/sudoers
```

a přidáme na konec následující řádek

```
přihlašovací-jméno ALL=NOPASSWD:/cesta-k-souboru/raid_status.sh
```

Výše uvedené nám zajistí, že při spuštění skriptu po nás nebude vyžadováno pokaždé heslo.

Teď si napíšeme skript, který budeme mít ve stejném adresáři jako raid_status.sh. Bude využívat první skript a zjišťovat, zda je vše v pořádku či nikoliv + odešle email s varováním.

```
sudo gedit raidmonitor.sh
```

Do něj vložíme a upravíme dle potřeby odesílatele, adresáta, předmět zprávy, samotnou zprávu atd.

```
#!/bin/bash
get_RAID_status()
{
sudo ./raid_status.sh | egrep -i "faulty|degraded" | wc -l
}
send_mail()
{
FILENAME="/tmp/raid_report_`date +%d_%m_%Y_%T`.txt"
sudo ./raid_status.sh > $FILENAME
sendEmail \
-f EMAIL-ODESÍLATELE \
-t EMAIL-ADRESÁTA \
-cc KOPIE-EMAILU-PRO-KOHO-NEPOVINNÉ \
-u PŘEDMĚT-ZPRÁVY \
-m NASTALY PROBLEMY S VASIM DISKOVYM POLEM!! \
Byl vygenerovan report. \
V príloze je vypis z `date +%d/%m/%Y_%T`. \
\
-s VÁŠ-SMTP-SERVER \
-o tls=auto \
-a $FILENAME
}

RAID_STATUS=`get_RAID_status`
if [ $RAID_STATUS -ne 0 ];then
send_mail
fi
```

Stručně řečeno: Spustí se zjištění stavu diskového pole a hledá se z výpisu slovo *faulty* nebo *degraded*. Pokud je alespoň jedno nalezeno, vytvoří se v /tmp soubor raid_report.txt, který obsahuje v názvu aktuální datum a čas. Je následně vytvořen a odeslán email, ke kterému je přiložen onen výpis. Pokud není nalezena chyba diskového pole, neděje se nic.

Funkčnost skriptu můžeme otestovat tak, že například hledanou hodnotu *faulty* změním na *clean*. Jakmile spustíme `raidmonitor.sh` z terminálu..

```
./raidmonitor.sh
```

mělo by se vše provést. Pokud nám dorazí email tak, jak si představujeme, je čas na nastavení opakování kontrol automaticky, i když třeba počítač běží a nejsme přihlášení. Použijeme pro toto [Cron](#)

(<http://cs.wikipedia.org/wiki/Cron>)

```
sudo crontab -e
```

na konec přidáme nový řádek

```
0 20 * * * (cd /CESTA-K-ADRESÁŘI-SE-SKRIPTEM; ./raidmonitor.sh)
```

stiskneme CTRL + X, dáme A a ENTER. Měli bychom dostat hlášku: *crontab: installing new crontab*.

Naše nastavení znamená, že každý den ve 20:00 se spustí náš skript. Formát času je *minuta hodina den měsíc rok*, přičemž * znamená vše/každý. Způsob nastavení času je velice pěkně popsán [zde](#) a [zde](#). Osobně jsem otestoval plánovač tak, že mi každou minutu poslal mail- nastaveno pomocí `*/1 *`

(<http://interval.cz/clanky/jak-na-demonu-cron/>)

a (<http://www.linuxexpres.cz/praxe/cron-spravca-uloh>)

Zrestartujeme službu Cron

```
sudo invoke-rc.d cron restart
```

Funguje-li hlášení na email korektně, nesmíme zapomenout změnit hodnotu *clean* zpátky na *faulty* - viz výše.

V [druhém dílu](#) se podíváme na vytvoření RAIDu z již běžícího systému přidáním dalšího disku.

5.2. Softwarové RAID pole - 2. díl

5. 2. 2009, afk_cz

<http://ubuntuhow2.blog.zive.cz/2009/02/softwareve-raid-pole-2-dil/>



2. díl o softwarových diskových polích je věnován vytvoření softwarového RAID pole z již běžícího systému, a to přidáním dalšího disku.

5.2.1. Úvodem

Návod navazuje na [předchozí díl](#), kdy jsme přidávali k systémovému disku další disky, systémový ale zůstal mimo diskové pole. Článek obsahoval trochu té teorie, srovnání, doporučení apod. Dnes se vrhneme přímo na věc.

5.2.2. Příprava

V dnešním modelovém případě budeme uvažovat takto: máme již operační systém nainstalovaný a chystáme se přidat další, stejně velký, nebo kapacitně větší, SATA disk. Bude nás zajímat zrcadlení, čili RAID 1. Náš systémový disk dnes bude jako `/dev/sda`, další, přidávaný disk, budeme vidět jako `/dev/sdb`. Jak zjistit značení disků jste se mohli dočíst v minulém dílu. Systémový disk je rozdělen následovně: `/dev/sda1` -je připojen do `/`, tedy jako kořenový adresář, `/dev/sda2` -je odkládací oddíl (swap), `/dev/sda3` -je např. jako úložiště dat připojený do `/media/storage`. Všechny 3 oddíly chceme mít v RAID 1, tzn. budeme vytvářet `/dev/md0` (z `/dev/sda1` a `/dev/sdb1`), `/dev/md1` (z `/dev/sda2` a `/dev/sdb2`), `/dev/md2` (z `/dev/sda3` a `/dev/sdb3`). Ani jeden z disků dříve nebyl součástí diskového pole (pokud byl, pak je třeba projít odstraněním zbytků nastavení disku, čemuž se budeme věnovat v dalším dílu tohoto seriálu).

5.2.3. Přidání dalšího disku

Připojíme další disk do PC a po nastartování bychom měli vidět v terminálu po zadání `sudo fdisk -l` něco takového

```
Disk /dev/sda: 500,1 GB, 500 107 862 016 bajtů
hlav: 255, sektorů na stopu: 63, cylindrů: 60 801
Jednotky = cylindry po 16065 * 512 = 8 225 280 bajtech
Identifikátor disku: 0xaaaaaaaa
Zařízení   Zavádět   Začátek       Konec         Bloky         Id       Systém
/dev/sda1   *         1             6079          48829536      83      Linux
/dev/sda2           6080          6444          2931862+      82      Linux
swap/Solaris
/dev/sda3           6445          60801         436622602+    83      Linux
```

```
Disk /dev/sdb: 500,1 GB, 500 107 862 016 bajtu
hlav: 255, sektorů na stopu: 63, cylindrů: 60 801
Jednotky = cylindry po 16065 * 512 = 8 225 280 bajtech
Identifikátor disku: 0x0006b48a
```

Disk `/dev/sdb` neobsahuje tabulku rozdělení

5.2.4. Instalace SW

Z repozitářů nainstalujeme balíky mdadm a initramfs-tools

```
sudo apt-get install mdadm initramfs-tools
```

Pokud se objeví dialog pro nastavení Citadel-serveru, postačí vše jen odenterovat. Jedná se o doporučený balík pro mdadm, který v podstatě nebudeme potřebovat a můžeme ho ihned po instalaci odebrat pomocí *sudo apt-get remove citadel-server*.

Dále si rovnou nahrajeme ručně několik modulů, abychom nemuseli restartovat systém.

```
sudo modprobe md
sudo modprobe linear
sudo modprobe multipath
sudo modprobe raid0
sudo modprobe raid1
sudo modprobe raid5
sudo modprobe raid6
sudo modprobe raid10
```

Ověříme si správné načtení modulů

```
cat /proc/mdstat
```

a pokud je vše OK, vidíme

```
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4]
[raid10]
unused devices: none
```

5.2.5. Příprava nového disku

Budeme postupovat tak, že připravíme nový disk */dev/sdb*, zkopírujeme na něj obsah */dev/sda*, spustíme pole v degradovaném režimu s 1 diskem (*/dev/sdb*) a ve finále do pole přidáme i náš systémový */dev/sda*.

Provedeme přesnou kopii rozdělení systémového disku na nový

```
sudo sfdisk -d /dev/sda | sudo sfdisk /dev/sdb
```

a pomocí fdisku ověříme rozdělení

```
sudo fdisk -l
```

měli bychom dostat

```
Disk /dev/sda: 500,1 GB, 500 107 862 016 bajtu
hlav: 255, sektorů na stopu: 63, cylindru: 60 801
Jednotky = cylindry po 16065 * 512 = 8 225 280 bajtech
Identifikátor disku: 0xaaaaaaaa
```

Zařízení	Zavádět	Začátek	Konec	Bloky	Id	System
/dev/sda1	*	1	6079	48829536	83	Linux
/dev/sda2		6080	6444	2931862+	82	Linux
swap/Solaris						
/dev/sda3		6445	60801	436622602+	83	Linux

```
Disk /dev/sdb: 500,1 GB, 500 107 862 016 bajtu
hlav: 255, sektorů na stopu: 63, cylindrů: 60 801
Jednotky = cylindry po 16065 * 512 = 8 225 280 bajtech
Identifikátor disku: 0x0006b48a
```

Zařízení	Zavádět	Začátek	Konec	Bloky	Id	System
/dev/sdb1	*	1	6079	48829536	83	Linux
/dev/sdb2		6080	6444	2931862+	82	Linux
swap/Solaris						
/dev/sdb3		6445	60801	436622602+	83	Linux

Systemu musíme říci, že oddíly disku *sdb* budou patřit do diskového pole

```
sudo fdisk /dev/sdb
```

zvolíme **t**, oddíl **1**, id **fd** (raid)

zvolíme **t**, oddíl **2**, id **fd**

zvolíme **t**, oddíl **3**, id **fd**

uložíme změny pomocí **w**.

Opět ověříme stav */dev/sdb* disku

```
sudo fdisk -l /dev/sdb
```

a vidíme u jednotlivých oddílů ID *fd* a *RAID*

```
Disk /dev/sdb: 500,1 GB, 500 107 862 016 bajtů
hlav: 255, sektorů na stopu: 63, cylindrů: 60 801
Jednotky = cylindry po 16065 * 512 = 8 225 280 bajtech
Identifikátor disku: 0xaaaaaaaaa
```

Zařízení	Zavádět	Začátek	Konec	Bloky	Id	System
/dev/sdb1	*	1	6079	48829536	fd	Linux RAID
samorozpoznatelný						
/dev/sdb2		6080	6444	2931862+	fd	Linux RAID
samorozpoznatelný						
/dev/sdb3		6445	60801	436622602+	fd	Linux RAID
samorozpoznatelný						

5.2.6. Vytvoření diskového pole

Vytvoříme degradované RAID 1 pole s počtem disků 2, přičemž první disk pole bude zatím chybět (tam později přijde náš systémový)- v terminálu zadáme

```
sudo mdadm -v --create /dev/md0 -l 1 -n 2 missing /dev/sdb1
sudo mdadm -v --create /dev/md1 -l 1 -n 2 missing /dev/sdb2
sudo mdadm -v --create /dev/md2 -l 1 -n 2 missing /dev/sdb3
```

Ověříme stav diskového pole

```
cat /proc/mdstat
```

zobrazí se nám

```
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4]
[raid10]
md2 : active raid1 sdb3[1]
```

```

436622528 blocks [2/1] [_U]

md1 : active raid1 sdb2[1]
      2931776 blocks [2/1] [_U]

md0 : active raid1 sdb1[1]
      48829440 blocks [2/1] [_U]

unused devices: none

```

Vytvoříme souborové systémy pro */dev/mdx*

```

sudo mkfs.ext3 /dev/md0
sudo mkswap /dev/md1
sudo mkfs.ext3 /dev/md2

```

Zazálohujeme si konfigurační soubor pro raid

```

sudo cp /etc/mdadm/mdadm.conf /etc/mdadm/mdadm.conf.puvodni

```

a jako root proskenujeme degradované diskové pole + uložíme výpis na konec konf. souboru (tento krok nejde pod Ubuntu 8.10 dělat pouze pomocí sudo, je třeba se přihlásit jako root)

```

su      ...zadáme heslo pro root
mdadm -examine -scan >> /etc/mdadm/mdadm.conf
exit    ...odhlásíme správce root

```

Prohlédneme si modifikovaný *mdadm.conf*

```

cat /etc/mdadm/mdadm.conf

```

Na konci výpisu vidíme něco podobného tomuto

```

ARRAY /dev/md0 level=raid1 num-devices=2
UUID=d793860a:222e5933:799f5a3b:588da548
ARRAY /dev/md1 level=raid1 num-devices=2
UUID=15722f79:48cd3024:799f5a3b:588da548
ARRAY /dev/md2 level=raid1 num-devices=2
UUID=1492d1bd:226a5ac8:799f5a3b:588da548

```

5.2.7. Přesun dat a úprava zavaděče

Připojíme si zatím prázdné diskové pole *md0* a *md2* do 2 libovolných adresářů. Odkládací prostor (swap), který budeme zrcadlit v *md1*, není třeba kopírovat.

```

sudo mkdir /media/{md0,md2}
sudo mount /dev/md0 /media/md0
sudo mount /dev/md2 /media/md2

```

Ověříme správné připojení oddílů

```

mount | grep md

```

dostáváme

```

/dev/md0 on /media/md0 type ext3 (rw)
/dev/md2 on /media/md2 type ext3 (rw)

```

Upravíme tabulku připojení disků

```
sudo gedit /etc/fstab
```

a nahradíme původní `/dev/sda1` za `/dev/md0`, dále `/dev/sda2` za `/dev/md1` a nakonec `/dev/sda3` za `/dev/md2`. U mne to vypadá takto (pozn.: `/dev/sda3` jsem měl jako úložiště připojený do `/media/storage`)

```
proc                /proc                proc                defaults            0                0
/dev/md0            /                    ext3                relatime,errors=remount-ro 0                1
/dev/md1            none                 swap                sw                  0                0
/dev/md2            /media/storage       ext3                auto,user,exec,rw, sync 0                0
/dev/scd1           /media/cdrom0        udf,iso9660        user,noauto,exec,utf8 0                0
```

Upravíme tabulku aktuálního připojení disků

```
sudo gedit /etc/mtab
```

a změníme původní `/dev/sda1` na `/dev/md0`, dále `/dev/sda3` na `/dev/md2`. Výpis bude obdobný jako u `fstab` výše.

Nesmíme zapomenout na zavaděč systému GRUB

```
sudo gedit /boot/grub/menu.lst
```

najdeme řádek `default 0` a vložíme pod něj nový

```
fallback 1
```

což nám zajistí v případě selhání startu systému automatickou volbu další položky v pořadí. Nastavíme také startování systému z nového disku v degradovaném poli. Najedeme na konec souboru, kde máme zmíněné odstavce, a zkopírujeme první pod sebe. V něm pak nahradíme `root=/dev/sda1` (popřípadě tam můžeme mít `root=UUID=xxxxxxxxxxxx`) novým `/dev/md0`. `Root (hd0,0)` pak nahradíme za `hd1,0` ... to je v podstatě 2. disk `/dev/sdb`, kde nám již běží RAID.

```
title                Ubuntu 8.10, kernel 2.6.27-11-generic RAID (hd1)
root                 (hd1,0)
kernel               /boot/vmlinuz-2.6.27-11-generic root=/dev/md0 ro nosplash
initrd               /boot/initrd.img-2.6.27-11-generic
savedefault
```

```
title                Ubuntu 8.10, kernel 2.6.27-11-generic BEZ RAIDU
root                 (hd0,0)
kernel               /boot/vmlinuz-2.6.27-11-generic root=/dev/sda1 ro nosplash
initrd               /boot/initrd.img-2.6.27-11-generic
savedefault
```

Tento krok nám zajistil možnost nastartovat jako předtím, kdyby se nepovedlo nastartovat z RAIDu.

Zaktualizujeme ramdisk

```
sudo update-initramfs -u
```

Zatím ještě NERESTARTUJEME, dokud nezkopírujeme data na diskové pole. Způsobů je mnoho, můžeme použít `tar`, `cp`, `rsync` ...

```
sudo rsync -avx --delete / /media/md0 ... zkopíruje systém do md0, ze kterého
budeme startovat
sudo rsync -avx -delete /media/storage /media/md2 ... nakopíruje /dev/sda3
(připojený do /media/storage) do md2
```

Před restartem už nám chybí poslední krok, konkrétně instalace Grubu i na druhý disk */dev/sdb*. To nám zajistí možnost nastartovat systém z druhého disku v případě pádu toho prvního.

```
sudo grub
```

a píšeme

```
root (hd0,0)
setup (hd0)
root (hd1,0)
setup (hd1)
quit
```

Zrestartujeme systém. V grubu při startu vybereme první volbu - na konci s *RAID (hd1)*.

Jak ověříme, jestli nám systém už naběhl s degradovaným diskovým polem z druhého disku ?
Postačí jednoduše zadat

```
df -h
```

a ve výstupu bychom měli vidět */dev/md0* připojeného do */* a také */dev/md2* připojeného do */media/storage*. Neměl by se tam nikde již vyskytovat */dev/sda*.

Pro jistotu znovu ověříme stav diskového pole

```
cat /proc/mdstat
```

zobrazí se nám stav degradovaného pole

```
md2 : active raid1 sdb3[1]
      436622528 blocks [2/1] [_U]

md1 : active raid1 sdb2[1]
      2931776 blocks [2/1] [_U]

md0 : active raid1 sdb1[1]
      48829440 blocks [2/1] [_U]

unused devices: none
```

5.2.8. Úprava původního disku, kompletace pole

Bližší se do finále. Zbývá nastavit příznaky původního systémového disku */dev/sda*, přidat ho do pole a udělat synchronizaci dat.

```
sudo fdisk /dev/sda
```

zvolíme **t**, oddíl **1**, id **fd** (raid)

zvolíme **t**, oddíl **2**, id **fd**

zvolíme **t**, oddíl **3**, id **fd**

uložíme změny pomocí **w**.

Ověříme zadáním

```
sudo fdisk -l
```

ve výpisu vidíme u všech oddílů */dev/sda* *Linux RAID samorozpoznatelný*.

Přidáme nyní disk */dev/sda* do pole

```
sudo mdadm --add /dev/md0 /dev/sda1
sudo mdadm --add /dev/md1 /dev/sda2
sudo mdadm --add /dev/md2 /dev/sda3
```

Okamžitě by měla začít synchronizace obou disků, aktuální stav uvidíme opět přes

```
cat /proc/mdstat
```

Upravíme opět *mdadm.conf* a zaktualizujeme ho

```
sudo cp /etc/mdadm/mdadm.conf.puvodni /etc/mdadm/mdadm.conf
su      ...zadáme heslo pro root
mdadm -examine -scan >> /etc/mdadm/mdadm.conf
exit    ...odhlásíme správce root
```

Po dokončení synchronizace se stav diskového pole

```
cat /proc/mdstat
```

změní takto

```
md2 : active raid1 sda3[0] sdb3[1]
436622528 blocks [2/2] [UU]
md1 : active raid1 sda2[0] sdb2[1]
2931776 blocks [2/2] [UU]
md0 : active raid1 sda1[0] sdb1[1]
48829440 blocks [2/2] [UU]
```

RAID máme tedy hotový.

Posledním krokem bude úprava zavaděče Grub tak, aby byl schopen startovat do RAIDu z obou disků. Momentálně v něm totiž pořád máme jen start do RAIDu z disku *hd1* a musíme mít pojistku pro případ pádu jednoho z disků. Zeditujeme tedy nabídku

```
sudo gedit /boot/grub/menu.lst
```

zkopírujeme první odstavec pro RAID z *hd1* a přepíšeme ho na *hd0*. Výsledkem je přítomnost obou variant.

```
title          Ubuntu 8.10, kernel 2.6.27-11-generic RAID (hd0)
root           (hd0,0)
kernel        /boot/vmlinuz-2.6.27-11-generic root=/dev/md0 ro nosplash
initrd        /boot/initrd.img-2.6.27-11-generic
savedefault
title          Ubuntu 8.10, kernel 2.6.27-11-generic RAID (hd1)
root           (hd1,0)
kernel        /boot/vmlinuz-2.6.27-11-generic root=/dev/md0 ro nosplash
initrd        /boot/initrd.img-2.6.27-11-generic
savedefault
```

```
title          Ubuntu 8.10, kernel 2.6.27-11-generic BEZ RAIDU
root           (hd0,0)
kernel        /boot/vmlinuz-2.6.27-11-generic root=/dev/sda1 ro nosplash
initrd        /boot/initrd.img-2.6.27-11-generic
savedefault
```

Než soubor uložíme a zavřeme, najdeme a upravíme ještě řádek s `# kopt=root=/dev/sd...` a upravíme ho takto

```
# kopt=root=/dev/md0 ro
```

‘#’ **nemazeme**. Řádek sice zůstává zakomentovaný, ale je použit např. při automatické aktualizaci `menu.lst`, což potřebujeme.

Zaktualizujeme ramdisk

```
sudo update-initramfs -u
```

Zrestartujeme systém. Nejprve s volbou RAIDu z `hd0` a podruhé z `hd1`. Obě varianty by měly být funkční.

5.3. Softwarové RAID pole - 3. díl

1. 6. 2009, afk_cz

<http://ubuntuhow2.blog.zive.cz/2009/06/software-raid-pole-3-dil/>



Vítejte u 3. a zároveň posledního dílu pojednávajícího o softwarových diskových polích. Tento je věnován odstranění softwarového RAID pole.

5.3.1. Úvodem

Návod ukončuje 3-dílnou sérii. Popisuje, jak postupovat při odstranění SW raidu tak, aby zůstala zachována data, ale disk už nevystupoval navenek jako raidový. Pokud bychom ho totiž vzali a chtěli dát do jiného raidu, nemusí se nám to bez regulérního odebrání totiž podařit bez komplikací.

5.3.2. Příprava

V dnešním modelovém případě budeme uvažovat takto: máme softwarový raid v provozu a na discích máme data, o která bychom rádi přijít. 1 disk chceme zachovat jako systémový, další disk/disky chceme použít jako samostatné popř. do nového raidu. Náš raid vidíme jako `/dev/md0,1,2`. Požadujeme zachování `/dev/sda1,2,3` jako primárního disku se systémem, odkládacím prostorem a úložištěm. Odebíráme `/dev/sdb` z raidu. Zařízení `/dev/md0` je připojeno do `/`, tedy jako kořenový adresář, `/dev/md1` je odkládací oddíl (swap), `/dev/md2` -je úložiště dat připojené do `/media/storage`. Všechny 3 oddíly jsou momentálně v běžícím RAID 1.

pozn.: pokud chcete odebírat disk z RAIDu 0, musíte nejprve všechna data odkopírovat na jiný disk, jinak o ně přijdete!

5.3.3. Zastavení RAIDu

Nejprve projistotu ověříme stav diskového pole, aby nám náhodou ve finále nezůstal jen poškozený disk.

```
cat /proc/mdstat
nebo
sudo mdadm --detail /dev/md0
sudo mdadm --detail /dev/md1
sudo mdadm --detail /dev/md2
```

Jestli je vše v pořádku, můžeme zkusit zastavit RAID

```
sudo mdadm --stop /dev/md0
sudo mdadm --stop /dev/md1
sudo mdadm --stop /dev/md2
```

Pokud se zadaří, přejdeme k další kapitole. Pokud neuspějeme u některých zařízeních, je to způsobeno pravděpodobně tím, že je máme někde připojeny a jsou používány (chyba *device or resource busy*

nebo *still in use* apod.) - bude třeba je před zastavením raidu odpojit. Pro `/dev/md0` by to bylo následovně

```
sudo umount /dev/md0
```

Nejde-li zařízení odpojit, můžeme vyzkoušet ‘lazy umount’ (pro `/dev/md0`) - pozn.: nedoporučuje se takto odpojovat USB disky

```
sudo umount -l /dev/md0
```

5.3.4. Vyčištění disků

Nejprve vynulujeme superbloky na každém disku

```
sudo mdadm --zero-superblock /dev/sda
sudo mdadm --zero-superblock /dev/sdb
```

Je třeba vrátit změny v konfiguračních souborech, které jsme udělali v [minulém dílu](#). Smažeme celý obsah `/etc/mdadm` nebo jen přejmenujeme `/etc/mdadm/mdadm.conf` (každý doporučuje něco jiného, já mazal celý adresář), dále zeditujeme `/etc/mtab` a změníme `/dev/md0` na `/dev/sda1`, `/dev/md1` na `/dev/sda2`, dále `/dev/md2` na `/dev/sda3`. To samé uděláme v `/etc/fstab`. Zeditujeme `/boot/grub/menu.lst` a smažeme řádek *fallback 1*. Pokud jsme si v `menu.lst` nechali původní odstavec s UUID disku nebo `/dev/sda`, pak postačí smazat celý odstavec obsahující `/dev/md`. V řádku `# kopt=root=/dev/md0 ro` změníme `md0` na `sda1`. Pozn.: jestliže máme zálohu celého `menu.lst`, pak nám to ušetří námahu 😊 Pokud najdeme `/etc/raidtab`, můžeme ho také smazat.

Pokud jsme na nic nezapomněli, po vypnutí počítače můžeme odpojit požadované disky a zkusíme nastartovat systém. Jestliže narazíme na problém, pořád máme data na odpojeném disku, po ruce jistě i LiveCD apod.

5.3.5. Závěrem

Doufám, že Vám má série článků pomohla při práci se SW Raidem. Pokud máte nějaké připomínky nebo náměty, neváhejte je napsat do diskuze pod článkem a já případně doplním návod.

5.4. RAID - degraded

5.4.1. Degraded RAID

At some point in the life of the computer a disk failure event may occur. When this happens, using Software RAID, the operating system will place the array into what is known as a *degraded* state.

If the array has become degraded, due to the chance of data corruption, by default Ubuntu Server Edition will boot to *initramfs* after thirty seconds. Once the *initramfs* has booted there is a fifteen second prompt giving you the option to go ahead and boot the system, or attempt manual recover. Booting to the *initramfs* prompt may or may not be the desired behavior, especially if the machine is in a remote location. Booting to a degraded array can be configured several ways:

- The `dpkg-reconfigure mdadm` utility can be used to configure the default behavior, and during the process you will be queried about additional settings related to the array. □ucha s monitoring, email alerts, etc. To reconfigure `mdadm` enter the following:

```
sudo dpkg-reconfigure mdadm
```

- The `dpkg-reconfigure mdadm` process will change the `/etc/initramfs-tools/conf.d/mdadm` configuration file. The file has the advantage of being able to pre-configure the system's behavior, and can also be manually edited:
- `BOOT_DEGRADED=true`

Note

The configuration file can be overridden by using a Kernel argument.

- Using a Kernel argument will allow the system to boot to a degraded array as well:
 - When the server is booting press Shift to open the Grub menu.
 - Press e to edit your kernel command options.
 - Press the down arrow to highlight the kernel line.
 - Add "`bootdegraded=true`" (without the quotes) to the end of the line.
 - Press Ctrlx to boot the system.

Once the system has booted you can either repair the array see ? for details, or copy important data to another machine due to major hardware failure.

5.4.2. RAID Maintenance

The `mdadm` utility can be used to view the status of an array, add disks to an array, remove disks, etc:

- To view the status of an array, from a terminal prompt enter:

```
sudo mdadm -D /dev/md0
```

The `-D` tells mdadm to display *detailed* information about the `/dev/md0` device. Replace `/dev/md0` with the appropriate RAID device.

- To view the status of a disk in an array:

```
sudo mdadm -E /dev/sda1
```

The output is very similar to the `mdadm -D` command, adjust `/dev/sda1` for each disk.

- If a disk fails and needs to be removed from an array enter:

```
sudo mdadm --remove /dev/md0 /dev/sda1
```

Change `/dev/md0` and `/dev/sda1` to the appropriate RAID device and disk.

- Similarly, to add a new disk:

```
sudo mdadm --add /dev/md0 /dev/sda1
```

Sometimes a disk can change to a *faulty* state even though there is nothing physically wrong with the drive. It is usually worthwhile to remove the drive from the array then re-add it. This will cause the drive to re-sync with the array. If the drive will not sync with the array, it is a good indication of hardware failure.

The `/proc/mdstat` file also contains useful information about the system's RAID devices:

```
cat /proc/mdstatPersonalities : [linear] [multipath
] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10] md0 : active raid1
sda1[0] sdb1[1] 10016384 blocks [2/2] [UU] unused devices: <none>
```

The following command is great for watching the status of a syncing drive:

```
watch -n1 cat /proc/mdstat
```

Press `Ctrl+c` to stop the watch command.

If you do need to replace a faulty drive, after the drive has been replaced and synced, grub will need to be installed. To install grub on the new drive, enter the following:

```
sudo grub-install /dev/md0
```

Replace `/dev/md0` with the appropriate array device name.

5.5. Cron I.

Cron je [softwarový démon](#), který v [operačních systémech](#) automatizovaně spouští v určitý čas nějaký příkaz resp. proces (skript, program apod.). Jedná se vlastně o specializovaný systémový proces, který v operačním systému slouží jakožto [plánovač úloh](#), jenž umožňuje opakované spuštění periodicky se opakujících procesů (např. noční běhy dávkových úloh při hromadném zpracování dat) apod.

5.5.1. Crontab

Příkazy a časy, ve kterých má Cron tyto příkazy provést, jsou definovány v souboru `crontab` resp. `crontab.txt` (celosystémová definice je zpravidla umístěna v adresáři `/etc/crontab` resp. `\cronw\crontab.txt`). Ke zjednodušení práce s definicí může existovat nástroj `crontab`.

5.5.2. Formát souboru

Soubor `crontab` obsahuje na každé řádce jednu definici úlohy; každá řádka obsahuje několik položek oddělených mezerami nebo tabulátory. První část řádky definuje časy spuštění, zbytek řádky je pak příkaz, který se má spustit.

Například následující definice zajistí spuštění programu `/usr/bin/updatestats` každý den jednu minutu po půlnoci:

```
1 0 * * * /usr/bin/updatestats
```

Tato definice zajistí spuštění skriptu `c:\scripts\active\weathermap.cmd` každých 5 minut:

```
*/5 * * * * C:\scripts\active\weathermap.cmd
```

Pět polí na začátku definuje čas:

1. Minuta (0–59)
2. Hodina (0–23)
3. Den v měsíci (1–31)
4. Měsíc (1–12)
5. Den v týdnu (0 = neděle, 1 = pondělí, ..., 6 = sobota)

Každé pole může obsahovat hvězdičku, což znamená, že se na danou hodnotu nebere ohled, resp. že se příkaz provádí vždy (např. hvězdička v položce hodina znamená, že se příkaz provádí každou hodinu). Jinak může obsahovat konkrétní číslo (7), seznam oddělený čárkou (15, 30, 45), rozsah oddělený pomlčkou (1–5), některé implementace Cronu umožňují složitější syntaxe (např. lomítko označující násobky, např. `*/5` může znamenat „každých pět minut“).

5.6. Jak na démona Cron

21. 04. 2002 | [Jiří Kocman](#) | [Editory a IDE](#) | [Komentáře: 6](#)

<http://interval.cz/clanky/jak-na-demonu-cron/>

Cron je Linux/Unix systémový nástroj, který spouští různé programy v předem definovanou dobu a intervalu (obdoba naplánovaných úloh ve Windows). Každý trochu vyspělejší webový projekt, stojící na výše jmenované platformě, se bez tohoto systémového démona neobejde. Popíši vám jak ho zkrátit pro vlastní potřeby.

Démon **cron** používají převážně Linux/Unix administrátoři ke spouštění programů a skriptů, které pomáhají udržovat operační systém. Například mazáním pomocných souborů, které po sobě některé nepořádné aplikace zachovávají, nebo prováděním pravidelného zálohování databází či celých disků.

Samozřejmě tento démon může velice dobře pomoci i tvůrcům internetových projektů. Jeho pomocí můžete v intervalech spouštět například generování databázově velmi náročných stránek, nebo můžete získávat aktuální informace z různých zdrojů na internetu, třeba [aktuální kurzovní lístek](#), o kterém jsem psal dříve. Můžete také například promazávat různá nepotřebná data v databázi, nebo... Nechávám vaši fantazii a potřebám.

Pokud používáte webhostingových služeb, pravděpodobně nebudete mít ke cronu přímý přístup přes příkazový řádek, takže vás nemusí zajímat jeho konfigurace. Pokud systém používáte sami pro sebe, nebudete se muset jeho konfigurací také příliš zabývat, jelikož jako root k němu budete mít přístup vždy. Pokud ale plánujete cron dovolit více uživatelům, kteří se budou moci k systému nalogovat k příkazové řádce, bude dobré nastavit uživatele, kteří s ním budou a nebudou moci pracovat. Standardně k němu mají přístup všichni uživatelé.

Chcete-li přístup omezit jen na určité uživatele, budete muset nastavit jména těchto uživatelů do souborů *cron.allow* a *cron.deny*. Oba soubory byste měli najít v */usr/lib/cron* nebo */etc/cron.d* (záleží na konfiguraci jednotlivých systémů). Pokud chcete povolit přístup jen uživateli `jirka` a `pavel`, napíšete do souboru *cron.allow* jejich uživatelská jména. Každé na jeden řádek. Tím všem ostatním uživatelům přístup ke cronu znemožníte. Pokud byste chtěli výše uvedeným uživatelům přístup zakázat a ostatním ho povolit, uvedete jejich uživatelská jména do souboru *cron.deny*. Stejně jako u předchozího souboru se píše jeden uživatel na jeden řádek. Samozřejmě uvedená uživatelská jména by měla v systému existovat (měly by být v souboru */etc/passwd*). Pokud tyto soubory neexistují, bude mít ke cronu přístup pouze *superuser* (root).

Když už víme, co budeme chtít pomocí cronu spouštět, musíme určit i čas spuštění. Cron nelze použít k jednorázovému spouštění aplikací či skriptů - cron spouští definované aplikace vždy v předem definovaném intervalu. Nutno podotknout, že tento interval je minimálně 1 minuta. Pokud potřebujete spouštět program častěji, bude asi vhodnější nechat program běžet permanentně a program na určitou dobu pauzovat.

Ke spouštění aplikací pomocí cronu jsou dvě možnosti. Prvá spočívá v nakopírování aplikace do jednoho z adresářů */etc/cron.daily*, */etc/cron.hourly*, */etc/cron.weekly* a */etc/cron.monthly*. Již z názvů těchto adresářů je jasné, jak často se budou aplikace v těchto adresářích spouštět. Pokud ale potřebujete přesnější časování, nebo jiný interval, budete muset přistoupit k utilitě **crontab**. Tato utilita slouží ke správě seznamů úloh, které cron vykonává. Každý uživatel má vlastní seznam úloh.

Pokud zavoláte crontab s parametrem `-e`, otevře se přednastavený textový editor a v něm bude obsah seznamu úloh. Pokud v seznamu nemáte žádný záznam, bude samozřejmě prázdný. Záznam do seznamu úloh má šest parametrů oddělených mezerami nebo tabelátory:

1. minuta
2. hodina
3. den v měsíci
4. měsíc
5. den v týdnu (0 - neděle, 1 - pondělí... 6 - sobota)
6. cesta k programu, který má cron spustit

Pokud místo některého z prvních pěti parametrů napíšete `*` (hvězdičku), znamená to, že parametr není definován. Příklad několika zápisů do seznamu úloh:

```
0 1 * * * /www/cgi-bin/skript.cgi
5 1 * * * /backup/make_backup
* 2 * * * wget http://www.server.cz/stahuj.php
0 0 1 * * /backup/delete_www_log
0,5,10,15,20,25,30,35,40,45,50,55 * * * *
  /www/cgi-bin/generuj_sessions.cgi
```

První řádek spouští *skript.cgi* každý den v 1:00 ráno. Druhý řádek cronu říká, aby spouštěl program *make_backup* denně v 1:05 ráno. Třetí řádek bude volat program *wget*, který bude stahovat obsah zadané internetové adresy - spustí se každou minutu od 2:00 do 2:59. (I když máme definovanou hodinu, nedefinovali jsme minutu, tudíž se cron bude pokoušet spouštět úlohu každou minutu v danou hodinu.)

Čtvrtý řádek tohoto příkladu bude spouštět program, který zazálohuje a následně vymaže log web serveru. Svou činnost bude provádět vždy na začátku nového měsíce. Poslední řádek bude spouštět skript každých pět minut.

Pravděpodobně budete chtít použít cron ke spouštění PHP skriptů. Před tím ale bude dobré zjistit, zda je na serveru PHP zkompileováno jako CGI verze, zda existuje spustitelný program **php**. Pokud ne, budete muset spouštět PHP skript pomocí programu *wget*, který jej zavolá obdobně jako internetový prohlížeč. Tento způsob má však několik nevýhod. Mezi hlavní nevýhodu patří možný timeout nastavený pro provádění PHP skriptů, takže takto volaný skript nemusí dokončit svou práci včas. Druhá nevýhoda je, že skript zbytečně vytěžuje webový server a tím snižuje i výkon celého serveru, spuštění a následné provádění PHP skriptu z příkazové řádky je mnohem rychlejší.

Pokud tedy na serveru máte nainstalováno PHP i jako CGI verzi, máte dvě možnosti, jak PHP skript spustit. První možnost, která nepotřebuje zásah do skriptu, je zavolání PHP interpretu a předání parametru, který bude obsahovat cestu k PHP skriptu. Příklad zápisu:

```
0 1 * * * /bin/php -f /www/server/skript.php
```

Druhá možnost spočívá v úpravě PHP skriptu. Skript se pak stává samostatně spustitelným programem obdobně jako CGI verze. Skript bude možná potřebovat více úprav, ale ta zásadní je v uvedení cesty k interpretu PHP do prvního řádku zdrojového kódu a přidělení execute práv tomuto skriptu.

```
#!/bin/php  
phpinfo();
```

Takovýto skript pak cronem zavoláte následujícím způsobem:

```
0 1 * * * /www/server/skript.php
```

V obou případech je nutné pamatovat, že cesty k případným vloženým či otevíraným souborům musí být uváděny absolutně. Jinak by mohlo docházet k chybám, protože interpret PHP by nenašel požadované soubory. Výhodou spouštění PHP skriptů pomocí dvou posledně uvedených metod je menší zátěž systému.

Autor komentáře: Pavel Štemberk

Datum vložení: 16. Prosinec 2009, 20:02:29

Děkuji za popis cronu. Jen pro úplnost bych dodal použití lomítka; tj. potřebuji-li něco spuštět např. každých 5 minut, do editoru napíši */5 * * * * /www/cgi-bin/generuj_sessions.cgi



Autor komentáře: Jan Biskupk

Datum vložení: 31. Říjen 2011, 10:42:09

Zdravím, potreboval bych poradit. Chci na serveru spouštět automaticky script. Pochopil jsem, že na to mám jít přes cron, ale nevím, jak dostat příkaz do crontabu.



Autor komentáře: maw

Datum vložení: 03. Listopad 2011, 16:22:55

Zkus crontab -e (v příkazovém řádku) - alespoň na linuxu to takto funguje na win nevím.



Autor komentáře: Pavla

Datum vložení: 15. Listopad 2011, 14:36:05

Měla jsem script napsaný tak, že při ručním spuštění vypisoval co dělá, po spuštění v cronu se začaly vytvářet logy....jak má být správně napsané logování tak, aby se ten log přepisoval?
30 9 * * * root wget <http://cesta/mujscript.php> > /dev/null 2>&1



Autor komentáre: Eddie

Datum vložení: 17. Leden 2012, 12:49:21

zedituj crontab prikazem "crontab -e" , predtim si ale nastav vychozi editor jelikoz se ti editace otevre ve vi, tzn. pokud pises napr. ve vim pust nasledujici command "export EDITOR=vim"

Co se tyce samotneho crontabu tak zadavat velice jednoduse tzn. napr. pokud chci kazdou 15min a 45min v hodine vyvolat urcity script tak takhle

```
15 * * * * /home/eddie/Scripts/ping.sh
45 * * * * /home/eddie/Scripts/ping.sh
```



Autor komentáre: Racky

Datum vložení: 08. Říjen 2012, 13:35:49

Ahoj.

Dneska jsem začal objevovat crontab a pomocí příkazu "crontab -e" jsem zapsal toto:

```
* * * * * wget localhost/my.php
```

ale nic se neděje. Skript má vyrobit soubor a každou minutu do něj zapsat čas, což se neděje, takže usuzuju, že cron nefunguje. Co s tím? Musím někde něco restartovat, nebo tak nějak

5.7. CRON II.

<http://www.linuxexpres.cz/praxe/cron-spravca-uloh>

5.7.1. Cron – správca úloh

Ak používate Linux (Unix) a náhodou ste narazili na tajomné slovo „cron“, možno sa poškrabete po hlave a spýtate sa – čo to je? Cron je názov programu, ktorý užívateľom Linuxu a tým pádom aj Unixu (čo už nebudem ďalej zdôrazňovať) poskytne možnosť spúšťať skripty alebo iba príkazy či skupinu príkazov automaticky a pravidelne v stanovenom čase.

Pondělí, 15. říjen 2007 | Autor [Juraj Šípoš](#) | [1](#) | známka **1,80**
[bash](#), [shell](#), [slovensky](#), [systém](#)

Cron – mocný, ba až kráľovský nástroj pre administráciu systému, poslúži vhodne rôznymi spôsobmi napr. aj ako možnosť zálohovania, keď v určitom čase, povedzme vždy o 17:00 každý deň alebo o 19:00 každú nedeľu, sa spustí program (skript), ktorý zálohuje systém, konfiguračné súbory alebo www stránky. Pravdaže, zálohovať možno hocičo. Cron takisto využijete na hocičo. Ak nesurfujete po internete, ale iba čítate poštu, možno v takom prípade bude jednoduchšie si prácu s e-mailom zautomatizovať pomocou programu cron – pošta sa stiahne vždy v stanovenom čase do vášho počítača. Rovnako pomocou démonu cron si môžete pravidelne zasielať SMS správu na mobilný telefón, treba mať však pevnú linku alebo systém, ktorý bude v čase, keď sa posiela e-mail, vždy aktívny.

Cron sa spustí pri štarte systému a potom ostane prítomný; pravidelne sa „zobúda“ (každú minútu) a kontroluje svoje „crontabs“ alebo „cron tables“ (cron tabuľky) – súbory, v ktorých sú zadefinované úlohy aj s časom pre ich spustenie. Cron ako démon je súčasťou systému a po inštalácii akéhokoľvek systému typu unix bude prítomný. Ak náhodou neviete, démon je rezidentný program, ktorý beží na pozadí – stačí ho teda spustiť iba raz. Keďže hovoríme o démonoch, v súvislosti s termínom „cron“ používame aj termín „crond“.

Crond sa na väčšine Linuxoch nainštaluje automaticky. Pravdaže, nemusí to byť vždy pravidlom. Máte povedzme tak trochu neštandardný Linux, alebo sa niekto hral s vašimi konfiguračnými súbormi. V takom prípade je potrebné zistiť, či crond beží, čo docielite nasledovným príkazom:

```
$ ps aux | grep crond
Mali by ste mali vidieť text podobný:
root    740   0.0   0.8  1568   684 ?        S    16:49   0:00 crond
root    941   0.0   0.7  1700   584 pts/0    S    17:29   0:00 grep crond
```

V tomto prípade crond beží. Ak nebeží, najlepšie je ho spustiť spúšťacími skriptami, ktoré systém aktivuje pri svojom štarte. Stačí iba pridať riadok crond do jedného z nich. Red Hat/Fedora Linux má inak súbor crond v adresári /etc/rc.d/init.d . Cron sa teda spustí pri každom štarte systému. Crond sa nachádza v adresári /usr/sbin/ . Ak cron nie je zadefinovaný v niektorom inicializačnom skripte, stačí zadať do príkazového riadku ako užívateľ root príkaz crond .

Rôzne konfiguračné súbory démonov (httpd.conf atď.) vyžadujú, aby bol démon reštartovaný, čo plní účel načítania konfiguračného súboru. Cron však automaticky načíta zmeny v súboroch. Treba však zohľadniť aj skutočnosť, že autorov je viac, preto ani cron nie je jeden. Z tohto dôvodu hovoríme aj „Vixie Cron“ (autorom je Paul Vixie), ktorý má pravidlá iné ako cron od iného autora. Vixie Cron je však štandardný cron, preto, ak používate Linux, na vašom systéme bude s najväčšou pravdepodobnosťou Vixie Cron.

5.7.2. Používame cron

Red Hat/Fedora Linux a aj iné distribúcie majú v adresári /etc podadresáre cron.daily (cron každý deň), cron.hourly (cron každú hodinu), cron.weekly (cron každý týždeň), cron.monthly (cron každý mesiac). Red Hat/Fedora sa v tomto trochu líšia od iných Linuxov. Keď nahliadnete do týchto adresárov, uvidíte v nich súbory, ktoré sa týkajú automatického spustenia programov. Ak chcete mať ešte väčšie pole pôsobnosti, môžete editovať súbor /etc/crontab, ktorého syntax na systéme RH/F je:

```
SHELL=/bin/bash
```

```

PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/
# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly

```

Prvú časť skriptu hádam netreba vysvetľovať. „Shell“ je Bash, Sh, Csh a PATH je cesta pre cron. Ak napríklad používate program, ktorý sa nachádza v adresári /usr/local/bin, pridajte túto cestu do hornej časti skriptu. MAILTO slúži ako e-mailová schránka pre výstup – ak nejaký program, ktorý cron spustil, má aj výstup (napr. hľadá súbory s povolenými všetkými právami), výstup bude poslaný do tejto e-mailovej schránky. HOME je domovský adresár. Položky v súboroch cron pozostávajú z polí podobne, ako je to v súbore /etc/passwd, ale separátor je medzerník. Máme sedem polí, ktoré predstavujú:

minute	minúta, kedy bude aplikácia, príkaz, skupina príkazov spustená; hodnoty sú 0-59
hour	hodina, kedy bude aplikácia, príkaz, skupina príkazov spustená; hodnoty sú 0-23, 0 je polnoc
dom	deň v mesiaci, kedy bude aplikácia, príkaz, skupina príkazov spustená; hodnoty číselne korešponujú s mesiacom, napr. 23 je každý mesiac dvadsiateho tretieho
month	mesiac, kedy bude aplikácia, príkaz, skupina príkazov spustená; hodnoty sú 0-12
dow	deň týždňa, kedy bude aplikácia, príkaz, skupina príkazov spustená; hodnoty sú 0-7, pričom 0 a 7 sú nedeľou
user	užívateľ, ktorý spúšťa príkazy
cmd	príkaz, ktorý chce užívateľ spustiť

V adresároch /etc, teda /etc/cron.hourly atď. sú globálne systémové hodnoty. Ich aplikovanie sa týka celého systému a užívateľ môže do týchto adresárov doinštalovať rôzne skripty. Napríklad príkaz tmpwatch, pre ktorý je v RH/F zadefinovaná cron úloha v adresári /etc/cron.daily, odstráni súbory, ku ktorým nikto neprístupoval istý počet hodín. Tmpwatch je dobré aplikovať na /tmp adresár. Stačí teda prekopírovať súbor s príkazmi do niektorého z adresárov /etc/cron.daily, /etc/cron.monthly a ten sa bude spúšťať podľa potreby vždy každú hodinu, každý deň, každý týždeň alebo každý mesiac. Ak skript prekopírujete do niektorého z týchto adresárov, môžete naň zabudnúť, pretože jeho vykonávanie už bude automatické. Linux takto uľahčuje užívateľom prácu a trochu času pri stanovení častých úloh.

5.7.3. Cron tabuľky

Cron tabuľky sú určené pre všetkých užívateľov a nemusia byť globálne. Ako globálne sa môžu javiť, ak si ich zadefinuje užívateľ root, ale hociktorý užívateľ môže mať právo editovať, mazať či tvoriť svoje cron tabuľky, teda úlohy. Linux je viac užívateľský operačný systém, a preto aj každý užívateľ môže mať vlastné súbory crontab, ktoré si môže vytvoriť, editovať atď. príkazom crontab. Aj keď ide o textový súbor, nemali by sme ho editovať priamo. Súbor crontab je často uložený v adresári /var/spool/cron/crontabs/užívateľ (na systémoch Unix/Slackware/*BSD), /var/spool/cron/užívateľ (na systémoch Red Hat/Fedora), alebo /var/cron/tabs/užívateľ (na systéme SUSE Linux). Meno užívateľa vychádza z jeho definície v súbore /etc/passwd. Súbor crontab sa samozrejme môže nachádzať aj inde – závisí to od distribúcie, ktorú používate.

Ak chcete editovať súbor crontab, zadajte príkaz `crontab -e`, ktorým spustíte editor. Ak používate Bash shell a máte nastavený iný, nie editor `vi`, a chcete použiť editor `vi`, zadajte do príkazového riadku príkaz `export EDITOR=vi`. Namiesto `vi` môžete uviesť akýkoľvek iný editor, v ktorom ste zvyknutí pracovať. Pre súbor `crontab` platia rovnaké pravidlá ako pre súbor `/etc/crontab`, avšak s výnimkou, že nemusíte špecifikovať premennú `MAILTO`. Ak ste zadefinovali požiadavky pre automatické spúšťanie a prácu s editorom ste ukončili, prebehne kontrola syntaxe a vy budete mať možnosť opraviť chyby. Súbor môžete napísať aj v hocijakom editore a potom ho aktivovať príkazom `crontab` tak, aby ste načítali vami vytvorený súbor, t.j. `crontab moj_subor.cron`.

Príkazom `crontab -l` si prezriete súčasne nastavenie a príkazom `crontab -r` odstránite nastavenie v aktuálne používanom `crontab` súbore. Ak máte vyššie práva, ostatným užívateľom môžete editovať, mazať súbory `crontab` príkazom `crontab -u meno_užívateľa`.

5.7.4. Niekoľko príkladov

Zadáte teda príkaz `crontab -e` a ak používate editor `vi`, stlačte klávesu `[Ins]`, aby ste mohli písať. Po editácii stlačte klávesu `[Esc]` a potom `:wq` (dvojbodka a „w“ a „q“ bez úvodzoviek). Najprv napíšete definície pre určenie času a potom príkaz. Povedzme napríklad, že máme súbor na zálohovanie ramdisku, v ktorom máme uložené `www` stránky. Ramdisk sa nachádza v adresári `/mnt/ram0`, vytvoríme si teda skript `backupram` a zapíšeme doň tieto riadky:

```
#!/bin/sh
/bin/tar cvfzp /ram0.tgz /mnt/ram0
```

Z príkazov vyplýva, že sa spustí program `tar`, ktorý zálohuje celý adresár `/mnt/ram0` a uloží ho do koreňového adresára / pod názvom `ram0.tgz`. Naším skriptom je teraz `backupram`, ktorý uložíme povedzme do adresára `/root/bin`. Preto, keď otvoríme editor s príkazom `crontab -e`, zapíšeme doň tento riadok:

```
* /15 * * * * /root/bin/backupram
```

Skript `backupram` bude spustený každých 15 minút, `*` je nešpecifikovaná hodnota. To, čo nás musí zaujímať, je časový údaj, nie príkaz samotný, lebo ten si môžete zmeniť podľa ľubovôle – namiesto `/root/bin/backupram` môžete uviesť aj iný skript, resp. hocičo. Používanie skriptov je užitočné, ak máte viac príkazov, sériu príkazov atď. Ak uvediete príkaz/skript, je vhodné uviesť jeho absolútnu cestu (`/root/bin/backupram`, nie iba `backupram`).

```
17 8 * * * /root/bin/backupram
```

Príkaz bude spustený každý deň o 8:17 ráno.

```
44 4 1 * * /root/bin/backupram
```

Príkaz je spustený o 4:44 ráno vždy prvého každého mesiaca.

Vixie Cron prijíma aj zoznamy v poliach, ktoré si najlepšie znázorníme na príklade:

```
45 11 * * 1,2,3,4,5 backup.sh
```

Príkaz spustí Bash skript `backup.sh` o 11:45 v pondelok, utorok, stredu, štvrtok a piatok.

Cron prijíma aj „krokové“ hodnoty ako */2 v poli dom bude znamenať, že sa spustí každé dva dni. Podobne hodnota */5 v poli „hodina“ bude znamenať, že sa príkaz spustí každú piatu hodinu. Z tohto vyplýva, že nasledovné príkazy sa budú chovať rovnako:

```
* 12 10-16/2 * * backup.sh
* 12 10,12,14,16 * * backup.sh
```

Zoznamy možno kombinovať jeden s druhým ako aj s „krokovými“ hodnotami:

```
* 12 1-15,17,20-25 * * príkaz
```

V adresári /home/username/domain-www/ máte skript, ktorý chcete spustiť každé ráno a 4:30. Cron úlohu vytvoríte nasledovne:

```
crontab -e
30 4 * * * /home/username/domain-www/script_name
Esc
:wq #(ukončenie editora „vi“)
```

Do konfiguračných súborov crontab môžete umiestniť aj poznámky, ale musí byť pred nimi znak #.

5.7.5. Kto bude mať prístup?

Cron má zabudovanú funkciu pre právo zasiahnuť do jeho útrob, teda kto môže a kto nemôže definovať a meniť nastavenie. V adresári /etc sa nachádza súbor /etc/cron.allow a /etc/cron.deny. Tieto súbory plnia podobný účel ako „allow/deny“ pre iné demony (hosts.allow atď.). Ak sa nenachádzajú v adresári /etc, treba ich vytvoriť. Ak nechcete niektorému užívateľovi (užívateľom) povoliť, aby používal cron, stačí do súboru cron.deny uviesť jeho meno. Podobne – ak chcete povoliť užívateľovi používať cron, do súboru /etc/cron.allow treba uviesť jeho meno. Ak chcete všetkým užívateľom zakázať používanie cronu, do súboru /etc/cron.deny treba uviesť: ALL napr. príkazom echo ALL >>/etc/cron.deny alebo hociktorým editorom.

Ak súbory /etc/cron.allow a /etc/cron.deny neexistujú, potom je používanie časového démonu cron neobmedzené (každý užívateľ ho môže používať).

5.7.6. Výstup z cronu

Výstup z démonu cron je poslaný na e-mailovú adresu jeho vlastníka, resp. adresu špecifikovanú v položke MAILTO, kde môžete uviesť aj inú e-mailovú adresu:

```
príkaz | mail -s "predmet mailu" užívateľ
```

Užívateľ nemusí mať účet na danom počítači, preto namiesto „užívateľa“ môžete uviesť aj e-mailovú adresu s účtom užívateľa, ktorému chcete poslať výstup z cronu. Niekedy nemusíte chcieť, aby bol výstup zaslaný e-mailom, preto môžete presmerovať výstup do log súboru alebo do /dev/null: príkaz >> súbor.log . >> znamená, že výstup nevymaže predchádzajúci zápis, ale zapíše sa na jeho koniec (append), > vymaže predchádzajúci zápis (write). Ak si však neviete rady a niektoré otázky stále nie sú jasné, môžete použiť grafické konfigurátory pre crontab, ktorých je niekoľko.

5.7.7. Tipy

Uved'me si príklad – chceme, aby nám cron vypisoval, kedy má kto narodeniny. Otec sa narodil 20.8., mama 2.11., Peter 24.2., Zuzana 8.3., brat 26.9. Otvoríme crontab príkazom `crontab -e` a zapíšeme doň nasledovné riadky. Výstup bude poslaný na zadanú e-mailovú adresu:

```
#(Minúta, hodina, deň v mesiaci, mesiac).
0 8 20 08 * echo Narodeniny má otec | mail tvoj@email
0 8 02 11 * echo Narodeniny má mama | mail tvoj@email
0 8 24 02 * echo Narodeniny má Peter | mail tvoj@email
0 8 08 03 * echo Narodeniny má Zuzana | mail tvoj@email
0 8 26 09 * echo Narodeniny má brat | mail tvoj@email
```

Beriete tabletku a občas sa stáva, že zabudnete. Dnes služby v sieti mobilných operátorov ponúkajú možnosť poslať e-mail aj ako SMS správu na mobilný telefón. Zapísaním nasledovného riadku

```
0 20 * * * echo "Nezabudni na liek!" | mail mobil@tuniekde.sk
```

docielite, že v 0 minút vždy o 20 hodine večer ktorýkoľvek mesiac či deň si pomocou e-mailu pošlete SMS správu na mobilný telefón.

Dobrym tipom môže byť aj pozrieť si manuálové stránky zadaním príkazu `man cron`, `man crontab` alebo `man 5 crontab`.

5.8. Shutdown

<http://www.computerhope.com/unix/ushutdow.htm>

5.8.1. About shutdown

[Shut down](#) the [system](#).

5.8.2. Syntax

```
shutdown [-akrhPHfFnc] [-t sec] time [message]
```

5.8.3. Description

The **shutdown** command brings the system down in a secure way. All [logged-in](#) users are notified that the system is going down, and [login](#) operations are blocked. It is possible to shut the system down immediately, or after a specified delay.

All processes are first notified that the system is going down by the [signal](#) **SIGTERM**. This gives programs like [vi](#) time to save the file being edited, [mail](#) and [news](#) processing programs a chance to exit cleanly, etc.

shutdown does its job by signalling the [init](#) process, asking it to change the [runlevel](#). Runlevel 0 is used to [halt](#) the system, runlevel 6 is used to [reboot](#) the system, and runlevel 1 is used to put the system into a state where administrative tasks can be performed (single-user mode). Runlevel 1 is the default, unless the **-h** or **-r** options are specified.

Your system may have a different set of options for the **shutdown** command; check your documentation for details.

5.8.4. Options

- a** Control access to the **shutdown** command using the control access file `/etc/shutdown.allow`. See [Access Control](#) below for more information.
- k** Do not shut down, but send the warning messages as if the shutdown were real.
- r** [Reboot](#) after shutdown.
- h** Instructs the system to shut down and then halt.
- P** Instructs the system to shut down and then power down.
- H** If **-h** is also specified, this option instructs the system to drop into boot monitor on systems that support it.
- f** Skip [fsck](#) after reboot.
- F** Force **fsck** after reboot.
- Don't call **init** to do the shutdown of processes; instruct **shutdown** to do that itself.
- n** The use of this option is discouraged, and its results are not always predictable.
- c** Cancel a pending shutdown. (This does not apply to "**shutdown now**", which does not wait before shutting down.) With this option, it is not possible to give the *time*

argument, but you can still specify an explanatory message that will be sent to all users.

-t *sec* Tell **init** to wait *sec* seconds between sending processes the warning and the [kill](#) signal, before changing to another runlevel.

The *time* argument specifies when to perform the shutdown operation.

time can be formatted in different ways:

time First, it can be an absolute time in the format *hh:mm*, in which *hh* is the hour (1 or 2 digits, from **0** to **23**) and *mm* is the minute of the hour (in two digits).

Second, it can be in the format *+m*, in which *m* is the number of minutes to wait.

In addition, the word **now** is the same as specifying **+0**; it shuts the system down immediately.

message A message to be sent to all users, along with the standard shutdown notification.

5.8.5. Details

If a **shutdown** is scheduled for the future, it will create the advisory file **/etc/nologin** which causes programs such as **login** not to allow new user logins. This file is created five minutes before the shutdown sequence starts. **shutdown** removes this file if it is stopped before it can signal **init** (i.e. it is cancelled or something goes wrong). It also removes it before calling **init** to change the runlevel.

The **-f** flag means "reboot fast". This only creates an advisory file **/fastboot** which can be tested by the system when it comes up again. The system boot rc file ("rc" stands for "runcom", which is short for "run commands") can test if this file is present, and decide not to run **fsck** since the system has been shut down in the proper way. After that, the boot process should remove **/fastboot**.

The **-F** flag means "force [fsck](#)". This only creates an advisory file **/forcefsck** which can be tested by the system when it comes up again. The boot rc file can test if this file is present, and decide to run **fsck** with a special "force" flag so that even properly unmounted [file systems](#) get checked. After that, the boot process should remove **/forcefsck**.

The **-n** flag causes **shutdown** not to call **init**, but to kill all running processes itself. **shutdown** will then turn off [quota](#), accounting, and [swapping](#) and [unmount](#) all file systems.

5.8.6. Access Control

shutdown can be called from **init** when the [CTRL-ALT-DEL](#) key combination is pressed, if there exists an appropriate entry in **/etc/inittab**. This would mean that everyone who has physical access to the console keyboard can shut the system down, however.

To prevent this, **shutdown** can check to see if an authorized user is logged in. If **shutdown** is called with the **-a** argument (add this to the invocation of **shutdown** in **/etc/inittab**), it checks to see if the file **/etc/shutdown.allow** is present. It then compares the login names in that file with the list of people that are logged in on a virtual console (from **/var/run/utmp**). Only if

one of those authorized users or root is logged in, it will proceed. Otherwise it will write the message

```
shutdown: no authorized users logged in
```

to the (physical) system console. The format of **/etc/shutdown.allow** is one user name per line. Empty lines and comment lines (prefixed by a #) are allowed. Currently there is a limit of 32 users in this file.

Note that if **/etc/shutdown.allow** is not present, the **-a** argument is ignored.

5.8.7. Halting vs. Powering Off

The **-H** option just sets the **init** [environment variable](#) **INIT_HALT** to **HALT**, and the **-P** option just sets that variable to **POWEROFF**. The shutdown script that calls [halt](#) as the last thing in the shutdown sequence should check these environment variables and call **halt** with the right options for these options to actually have any effect.

5.8.8. Files

```
/fastboot  
/etc/inittab  
/etc/init.d/halt  
/etc/init.d/reboot  
/etc/shutdown.allow
```

5.8.9. Examples

```
shutdown 8:00
```

Schedule the system to shut down at 8 A.M.

```
shutdown 20:00
```

Schedule the system to shut down at 8 P.M.

```
shutdown +15 "Upgrading hardware, downtime should be minimal"
```

Schedule the system to shut down in fifteen minutes. Along with the normal message alerting users that the system is shutting down, they will be given the descriptive message about a hardware upgrade.

```
shutdown now
```

Bring down the system immediately.

```
shutdown -r now
```

Bring down the system immediately, and automatically reboot it.

```
shutdown -P now
```


Bring down the system immediately, and automatically power off the system.