



Estonian Information  
Technology College

# Filesystem Hierarchy

## *Operating systems 1800*

Edmund Laugasson  
edmund.laugasson@itcollege.ee

*There has been used materials from Margus Ernits, Katrin Loodus when creating current slides.*

Current document copying, distributing and/or modifying has been set out by one of the following licences by user's choice:

\* GNU Free Documentation Licence version 1.2 or newer

\* Creative Commons Attribution + ShareAlike licence 4.0 (CC BY-SA)

# Problems

- when renewing the server
  - new will be installed next to old one
    - cannot copy (new hardware is completely different and also OS will be renewed)
    - the program configurations should be migrated
  - also user data will be migrated
- there is a need to know precisely:
  - what is needed to migrate from old to new system
  - what is needed to install onto new server

# Problems 2

- restoring a machine
  - when hardware will be damaged and restoring onto same hardware will not succeed...
    - there will be new machine installed
    - there will be software installed
    - there will be software configured
    - there will be data restored
- therefore there will be precisely needed to know: what to install and restore

# How to keep system files

- how to keep system files so that:
  - the system is easy to restore
  - some of the files could be kept on network drives (usually home folders, in some cases even program files)
  - user files would be separately
  - locally installed programs would be distinguishable from those of the installed by manufacturer in the system
- UNIX-like OS (like Linux) this has been standardized
- **every** IT administrator must know the principles

# Filesystem Hierarchy

- UNIX-like have a standard *Filesystem Hierarchy Standard* (FHS)
  - describes the location of files and directories
  - the goal is to raise the compatibility between systems and software displacement simplification
  - makes
  - makes it easier for system administrators to work
- The standard is available on the address:
  - <http://refspecs.linuxfoundation.org/fhs.shtml>
- FHS development:
  - <https://wiki.linuxfoundation.org/en/FHS>
  - <https://wiki.linuxfoundation.org/lfs/fhs>

# ls -l /

- bin
- boot
- cdrom
- dev
- etc
- home
- `initrd.img`
- lib
- lib32
- lib64
- lost+found
- media
- mnt
- opt
- proc
- root
- run
- sbin
- srv
- sys
- `tmp`
- usr
- var
- `vmlinuz`

# FHS

- Linux system files are in folders
- *Filesystem Hierarchy Standard (FHS)* defines folders and files hierarchy
- */* is the system *root directory*
  - root directory is the beginning of tree-like filesystem
  - must be available when system is booting (e.g. /usr /opt and /var should not)
  - the system must provide tools for primary remediation

# / directory must contain

- **/bin** programs necessary for the system to load
- **/boot** bootloader files
- **/dev** device files
- **/etc** host specific configuration files
- **/home** users' home directories
- **/lib** shared libraries and kernel modules (in addition to the `/lib32`, `/lib64`)
- **/media** removable media mount point
- **/mnt** temporary file system mount point
- **/opt** additional software is installed to opt folder
- **/sbin** mainly system commands meant for root user
- **/srv** service data files (RedHat based distributions)
- **/tmp** temporary files
- **/usr** a hierarchy that is not needed at boot time, but often in the main part of the programs installed in the `usr` directory
- **/var** variable-sized data files



# /bin

- **/bin/** system programs that are needed to boot the system (mount, cp, cat, etc)
  - contains a files that are used by system boot scripts
  - contains a programs that both regular users and root user will be used
  - must contain: ***cat chgrp chmod chown cp date dd df dmesg echo false hostname kill ln login ls mkdir mknod more mount mv ps pwd rm rmdir sed sh stty su sync true umount uname***
  - can contain: ***csh ed tar cpio gzip gunzip zcat netstat ping***

# FHS

- **/boot/** *kernel, initrd*
  - in the system there can be multiple kernels that can be choosed when booting
- **/dev/** device files
  - must contain the command **MAKEDEV**
  - e.g. */dev/null* , */dev/urandom* etc
- **/etc/** configuration files
  - the configuration of system and services
  - binary files are not allowed here
- **/home/** users' home directories
  - usually on the separate partition

## FHS 2

- **/lib/** shared libraries
  - libraries are used by binary files that are located at `/bin/` and `/sbin/`.
- **/media/** here will be mounted removable media
  - e.g. CD/DVD/BD-ROM, USB removable storage, cameras
- **/mnt/** here will be mounted temporary file systems
  - in old Linux systems also optical storage etc

# FHS 3

- **/opt/** for additional programs, e.g. third party programs
- **/proc/** virtual file system to access kernel parameters
  - *cat /proc/cpuinfo*
  - *cat /proc/meminfo*
  - *cat /proc/asound/cards*
  - *cat /proc/partitions*
  - *cat /proc/filesystems*
  - *cat /proc/devices*
  - *cat /proc/interrupts*
  - *cat /proc/ioproports , cat /proc/iomem*
  - *cat /proc/version*
  - *cat /proc/uptime, jne*
  - through the virtual hierarchy there will be information about system and processes
  - *man proc*

# FHS 4

- **/root/** superuser home directory
- **/sbin/** system programs that are needed to boot the computer
  - e.g. init, ifup, fdisk, etc.
- **/srv/** service state files
  - actually this is not yet respected by distro creators. The reason: there is no consensus
- **/sys/** virtual file system to keep device status

# FHS 5

- **/tmp/** for temporary files and folders
- **/usr/** hierarchy for files that are not needed to boot the system
  - therefore there can be large program files kept over network by keeping **/bin** and **/sbin** sizes small
- **/usr/bin/** program files that are not needed to boot the system
  - e.g. mplayer
- **/usr/bin/X11/** GUI files

## FHS 6

- **/usr/include/** C compiler header files
- **/usr/lib/** program shared libraries
  - for programs in */usr/bin/* and */usr/sbin/*
- **/usr/local** for sysadmins to install software locally and use globally
- **/usr/sbin/** programs needed for system management but not needed to boot
- **/usr/share/** shared files. Do not depend on architecture (e.g. IA-32, x86-64)
- **/usr/src/** source-code, e.g. kernel

# FHS 7

- **/var/** files with variable size
  - system log files
  - e-mail data
  - printer spooler
  - backup files
  - service files (www etc)
- **/var/lock/** resources lock files
- **/var/log/** different log files
- **/var/mail/** user mail boxes



# FHS 8

- **/var/run/** the current system information services and their users since the last boot
- **/var/spool/** queue - waiting for data processing
  - e.g. printer queues, unread e-mail
- **/var/tmp/** temporary files that can not be deleted when booting the system

# but MS Windows?

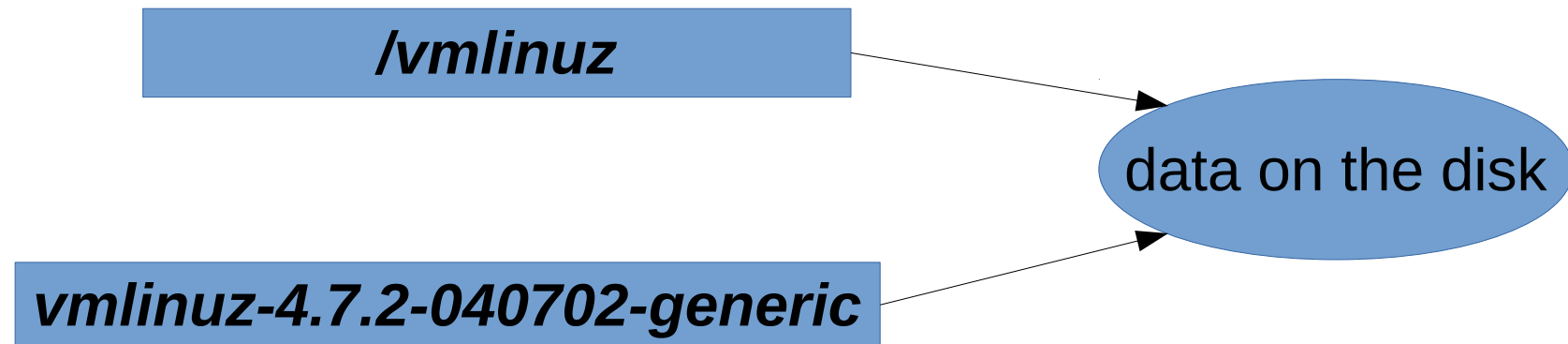
- %windir%
- Program Files
- Users
- %windir%\System32 - 32bit libraries and programs
- %windir%\SysWOW64 - 64bit libraries and programs
- <http://technet.microsoft.com/en-us/library/bb457124.aspx>
- [http://msdn.microsoft.com/en-us/library/windows/desktop/aa384187\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa384187(v=vs.85).aspx)

# Links

- sometimes there is needed that same file/folder would be located in multiple locations at the same file system

e.g. ***ls -l /vmlinuz***

```
lrwxrwxrwx 1 root root 33 aug 22 08:03 /vmlinuz -> boot/vmlinuz-4.7.2-040702-generic
```



# Links 2

- to one file there can refer many links on different places
- programs can access to file/folder through the link
- UNIX-like systems may have two types of links
  - Hard Link
    - reference to data
    - every link is equal
    - can only be done within a single file system
  - Symbolic Link
    - the name link
    - it can refer to a file on another file system

# Links 3

- **hard link**
  - In [parameters] filename linkname
- **symbolic link**
  - In -s filename linkname
- examples
  - In -s /media/disk/user “/home/user/usb disk”
  - will create a link “usb disk” that refer e.g. to external disk that will be mounted under /media folder

# Broken links

- when you delete a file that is referred to by a link
- if you delete a file, the name referred to by link, then the link is broken
- link remains in the system, but the reference of the file is deleted
- `find -L . -type l`
  - helps you find broken symbolic links
- if you delete the name of the link, the file will not be deleted
- hard link - since the link is equivalent to a file, the data is not lost
  - <http://unix.stackexchange.com/questions/50179/what-happens-when-you-delete-a-hard-link>

# MS Windows NTFS links

- MS Windows NTFS allows also links
  - hard link alternative:
    - `mklink /h link fail`
  - symbolic link (since MS Windows Vista NTFS)
    - `mklink [parameters] link file`
  - <https://jpsoft.com/help/mklink.htm>

# References

- Filesystem Hierarchy Standard (FHS)  
<http://refspecs.linuxfoundation.org/fhs.shtml> - choose the newest version
- Linux Filesystem Tree (Ubuntu)  
<https://help.ubuntu.com/community/LinuxFilesystemTreeOverview>
- Filesystem Hierarchy Standard  
[http://en.wikipedia.org/wiki/Filesystem\\_Hierarchy\\_Standard](http://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard)
- Virtual file system /proc  
<http://tldp.org/LDP/Linux-Filesystem-Hierarchy/html/proc.html>
- An introduction to Linux filesystems
  - <https://opensource.com/life/16/10/introduction-linux-filesystems>



Questions?

Thank you for your attention!

