

# Linux

Struktura adresářů

Zařízení

Práva

Zavaděč OS

# Příkazový řádek

- Přihlášení do systému
- Spouštění příkazů
- Používání virtuálních konzolí
- Odhlášení ze systému
- Vypnutí systému
- Práce s příkazovým řádkem
- Nápoměda
- Adresářový strom
- Práce s adresáři
- Práce se soubory
- Přístupová práva
- Práce s procesy
- Archivace a komprimace
- Skriptování

# Přihlášení do systému

- Běžný uživatel
  - Vlastní uživatelský účet (/etc/passwd , /etc/shadow)
  - domovský adresář a specifická nastavení (/home/username)
- Administrátor systému
  - Uživatel s označením root (/root)
  - Prakticky neomezená práva
- Systémové služby
  - systémové služby vystupují jako uživatelé
  - je nezbytné, aby měly určitá oprávnění (omezení)
  - Nelze využít k přihlášení
- Výzva login:, zadává se uživatelské jméno. Po potvrzení klávesou [Enter]
- Výzva password: zadává se heslo, které se nezobrazuje na obrazovku.

*root patří do skupiny root, jeho domovský adresář se jmenuje root a je v rootu*

# Spouštění příkazů

- **Příkazový řádek** zobrazuje výzvu (**prompt**) a (blikající) kurzor.
  - Příkazy se vkládají napsáním jejich názvu a stiskem klávesy „Enter“.
  - Interpret příkazů (**bash**) spustí příslušný program.
  - Je možno zadávat volby a parametry příkazu
- Prompt (*user@host:dir\$*)
  - lze změnit nastavením proměnné shellu PS1
  - Uživatel **user** na počítači **host** v adresáři **dir**
    - **user** – jméno přihlášeného uživatele
    - **host** – jméno počítače na kterém se nacházíte, je možno se přihlásit vzdáleně
    - **dir** – aktuální adresář (~ je alias pro domovský adresář přihlášeného uživatele)
    - \$ - zakončení promptu u běžného uživatele, u uživatele root se používá #

# Spouštění příkazů

- **Roura (pipe) |**

Pojmenovaná roura (mkfifo)

- připojuje výstup jednoho procesu na vstup druhého procesu. Např. **příkaz1 | příkaz2**

- **Přesměrování**

- `>` přesměrování standardního výstupu do souboru, jestliže soubor existuje bude přepsán
- `>>` jako předchozí, ale data přidá na konec souboru
- `<` přesměrování obsahu souboru na standardní vstup
- `<<text` jako předchozí, ale při výskytu řetězce text zašle znak konce souboru

- Před operátorem přesměrování jde použít deskriptor.

- `0` standardní vstup
- `1` standardní výstup
- `2` standardní chybový výstup

<code>2&gt;</code>	Přesměrování chybového výstupu
<code>&amp;&gt;</code>	přesměrování obou dvou proudů do stejného souboru.
<code>2&gt;&amp;1</code>	Sloučení chybového výstupu se standardním výstupem <i>příkaz &gt;&gt;výstupní-soubor 2&gt;&amp;1</i>
<code>1&gt;&amp;2</code>	Přesměrování std. výstupu příkazu na std. chybový výstup

Volba *noclobber* zabraňuje přepsání souboru pomocí `>`  
`set {-o|+o} noclobber` # zapnutí/vypnutí volby *noclobber*

# Bash (Bourne Again Shell)

- Rozhraní mezi uživatelem a systémem.
- V interaktivním režimu čeká na zadání příkazu od uživatele.
- Příkazy jsou
  - interní (integrované v shellu): např. cd, pwd, fg, bg, kill, ...
  - Externí: např. cp, rm, shutdown, ...
- Přizpůsobení pracovního prostředí pomocí proměnných.
- Nastavení
  - /etc/bashrc, /etc/profile,
  - ~/.bash\_profile, ~/.bash\_login, ~/.profile, ~/.bashrc
  - ~/.bash\_logout

# Klávesové zkratky pro bash (výběr)

- Ctrl + A Přesun začátek řádku, klávesa HOME
- Ctrl + E Přesun na konec řádku, klávesa END
- Ctrl + L Vymaže obrazovku (příkaz *clear*)
- Ctrl + R Prohledání dříve používaných příkazů
- Ctrl + C přeruší (kill) aktuální proces na popředí, signál SIGINT.
- Ctrl + D Ukončí shell, odesílá EOF (End-of-file), (příkaz *exit*)
- Ctrl + Z Stopne proces na popředí, signál SIGTSTP.
- Tab Automatické doplňování jmen souborů a složek
- Šipka nahoru/dolů Procházení historie předchozích příkazů
  - Ctrl + P, Ctrl + N

# Používání virtuálních konzolí

- Linux poskytuje několik virtuálních terminálů
  - lze se přepínat pomocí kombinace kláves **Alt + Fn** [1-6] (obvykle)
  - V GUI se používá pro přechod do textového terminálu **Ctrl + Alt + Fn**.
    - Návrat do GUI většinou pomocí **Alt + F7**
  - Na každém terminálu může být jiný uživatel přihlášený na jiný počítač.
  - K přepínání lze použít **Alt + šipka vlevo / vpravo**



# Odhlášení ze systému

- *logout* odhlášení ze systému
- *exit* odhlášení ze systému (Ctrl +D)
- *shutdown* vypnutí/restart systému
- *halt* vypnutí systému
- *poweroff* vypnutí systému
  
- *reboot* restart systému
  
- *init, telinit, systemctl*

# Práce s příkazovým řádkem

- *příkaz [volby]... [argumenty]...*
- Část **příkaz** určuje, který příkaz se bude provádět.
- Většina příkazů umožňuje zadávat **volby**, které ovlivňují chování
  - současně lze zadávat více voleb, oddělují se mezerou nebo tabulátorem.
- **Argumenty** (vstupní parametry) určují soubory a ostatní objekty, nad kterými má být příkaz proveden.
  - Seznam argumentů se odděluje mezerou nebo tabulátorem.
- Linux rozlišuje malá a velká písmena
  - narozdíl od Microsoft Windows

# Nápověda

- Většina příkazů nabízí interní nápovědu pomocí **-h** nebo **--help**
  - syntaxe: **příkaz -h** nebo **příkaz --help**
  - Jestliže je volba celé slovo, používají se pomlčky dvě.
  - Jestliže je volba jedno písmeno, používá se jedna pomlčka.
    - Za jednou pomlčkou lze napsat více jednopísmenných voleb
      - Např. **-help** bude interpretováno jako 4 různé volby zapsané jako **-h -e -l -p**
- Manuálové stránky
  - Komplexní nápověda k příkazům, nastavení, ...
  - Syntaxe: **man příkaz**
  - Pohyb v nápovědě pomocí šipek/PgDn/PgUp. Vyhledávání na stránce /
    - Fungují klávesové zkratky jako v editoru Vim
  - Ukončení klávesou **q**

# Nápověda - syntaxe

- Hranaté závorky znamenají, že použití je volitelné.
- Tři tečky udávají, že voleb/parametrů může být více.
- Oddělení znakem | znamená, že se jde použít jedna z možností.
  - Někdy jsou použity čárky “,”
- Uzavření do složených závorek znamená nucený výběr
  - `[ -I { keyword [...] | ALL } ]`
    - Volba `-I` je volitelná, pokud je použita, tak **musí** následovat volba `ALL` **nebo** klíčové slovo (může jich být více oddělených čárkou)

# Manuálové stránky - sekce

- Manuálové stránky se dělí do sekcí
  - rozdělují obsah podle významu.
- Běžně používané sekce jsou 1-8, popisují:

- |   |  |
|---|--|
| 1 | Uživatelské příkazy                          |
| 2 | System calls(funkce poskytované kernelem)    |
| 3 | Library calls (funkce programových knihoven) |
| 4 | Speciální soubory(např. obsah /dev)          |
| 5 | Formát konfiguračních souborů                |
| 6 | Hry  |
| 7 | Práce s textem                               |
| 8 | Příkazy pro správu systému                   |

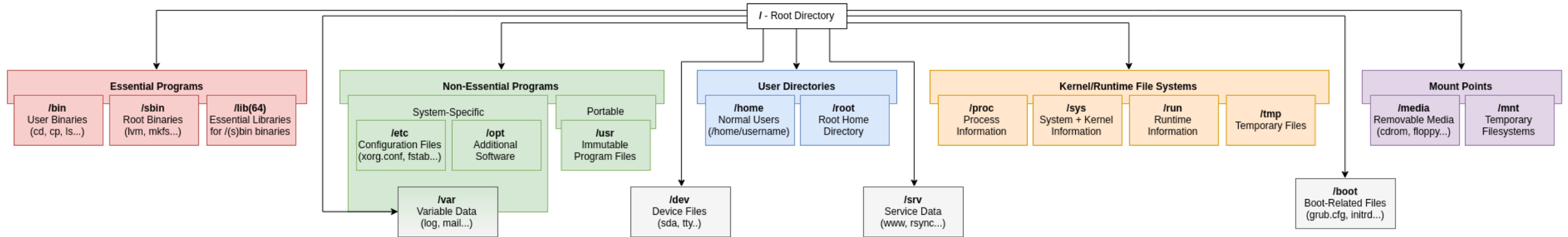
- V závislosti na instalaci systému, mohou existovat i další:

0	Hlavičkové soubory
0p	Hlavičkové soubory(POSIX)
1p	Uživatelské příkazy(POSIX)
3n	Síťové funkce
3p	Perl Moduly
9	Kernel rutiny
l	Lokální dokumentace
n	Nové manuálové stránky

<https://unix.stackexchange.com/questions/3586/what-do-the-numbers-in-a-man-page-mean>

# Adresářový strom

## The Filesystem Hierarchy Standard (FHS)



### Essential Programs:

Directories containing files needed to run essential programs

- **/bin** - Essential binaries such as 'cp' or 'ls' that all users have access to
- **/sbin** - Essential binaries only available to the root user
- **/lib(64)** - Libraries needed for essential binaries in /(s)bin

### Non-Essential Programs (Secondary Hierarchy):

Directories containing files needed to run non-essential programs

- **/etc** - System-specific configuration files for programs in /usr and /opt
- **/opt** - Additional programs not found in distribution repositories
- **/usr** - Portable, read-only, non-essential programs and program files
- **/var** - Used for storing dynamic program data that may change

### Mount Points:

Directories used for mounting devices and file systems

- **/media** - Removable media such as CD-ROMs and floppy drives
- **/mnt** - Temporary file systems such as USB drives

### User Directories:

Directories containing user-specific files

- **/home/(username)** - User files, configuration, and programs
- **/root** - Home directory for the root user

### Kernel/Runtime File Systems:

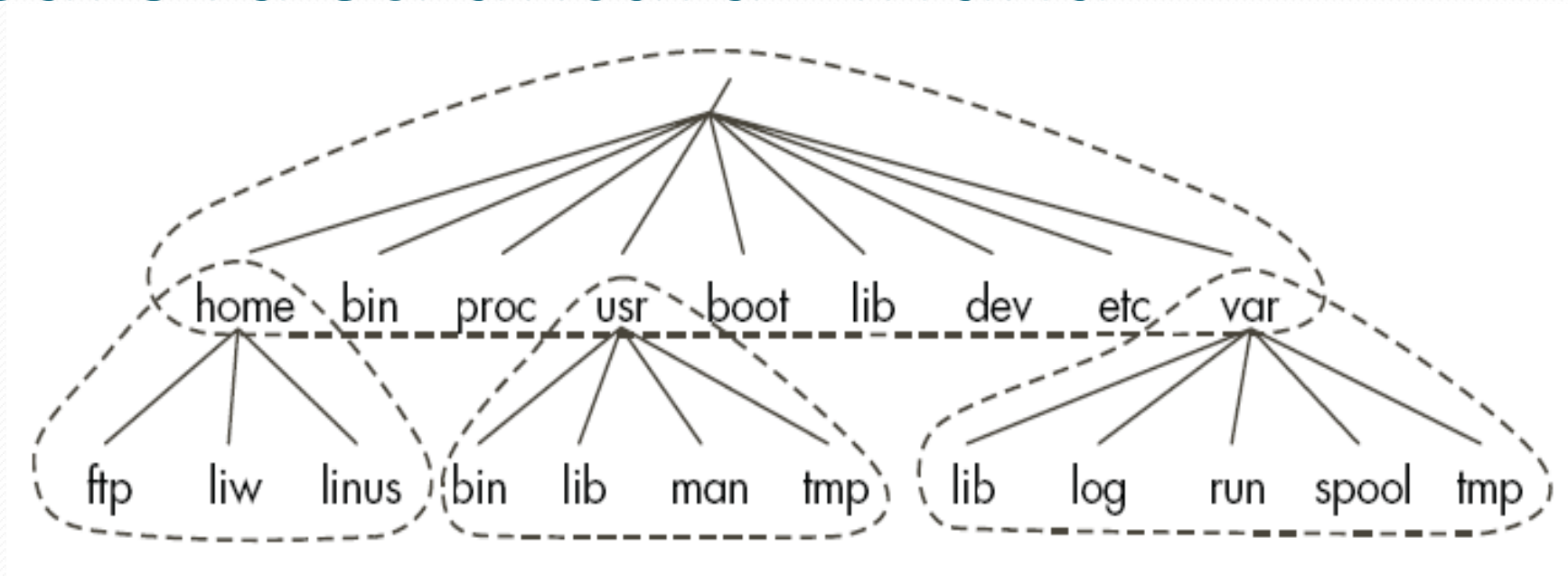
Directories populated by the kernel to provide information to programs and the user  
(Zdroj: <https://i.imgur.com/0vfm4dm.png>)

- **/proc** - Information about processes, the kernel and system hardware
- **/sys** - Information about system hardware and the kernel
- **/run** - Information about the system since the last boot
- **/tmp** - Directory for temporary files. Usually a tmpfs that is cleared on boot

### Other directories:

- **/boot** - Files essential for booting the system such as initrd, kernel, and bootloader configuration
- **/dev** - Device files for physical devices such as HDDs as well as data streams (stdin, stdout...)
- **/srv** - Files used for services offered by the system such as www, rsync, and ftp
- **/var** - Variable (changing) files such as lock files, logs, mail, and program files

# Adresářová struktura Linuxu



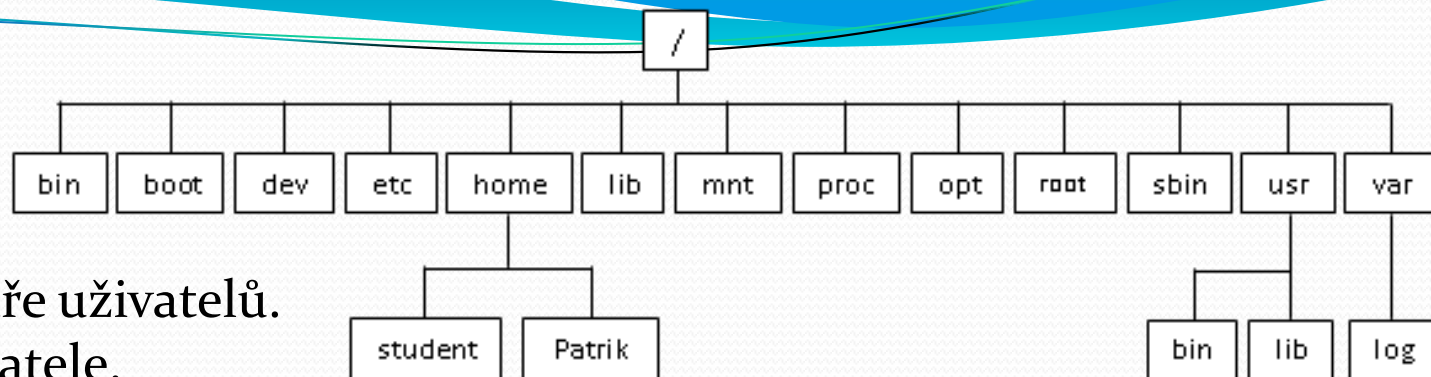
-Svazky

-FHS (Filesystem Hierarchy Standard)



# Adresářový strom

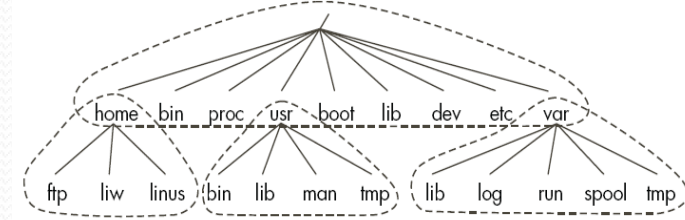
- root – kořenový adresář /
  - /home - Obsahuje domovské adresáře uživatelů.
  - /root - Domovský adresář superuživatele.
  - /lib - Sdílené knihovny pro programy.
  - /dev - Soubory zařízení.
  - /tmp - Dočasné soubory. Programy by měly používat /var/tmp
  - /bin - Běžné uživatelské programy (spustitelný kód).
  - /sbin – Spustitelné programy pro správu systému, většinou přístupné pouze pro roota.
  - /boot – Soubory zavaděče a jádra.
  - /mnt – Dočasné přípojně místo.
  - /media – Přípojně místo pro vyměnitelná média (CD, DVD, USB disk, ...).
  - /etc – Konfigurační soubory.
  - /usr – Neměnné programové soubory.
  - /var – Data proměnná v čase.
  - /opt – Další software, většinou neinstalovaný z repozitáře.
  - /proc – Informace o systému (kernelu, běžících procesech, ...), obraz paměti.



(Zdroj: vlastní)

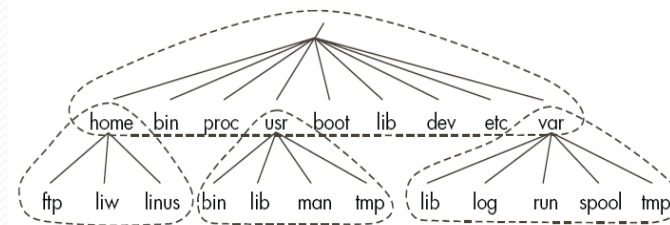


# Výpis adresáře root



```
total 108
drwxr-xr-x  2 root bin   4096 Jul 14 15:53 bin/
drwxr-xr-x  2 root root  4096 Jul 19 14:41 boot/
drwxr-xr-x 15 root root 40960 Oct 12 09:40 dev/
drwxr-xr-x 28 root root  4096 Oct 12 09:40 etc/
drwxr-xr-x  4 root root  4096 Jul 19 14:39 home/
drwxr-xr-x  4 root root  4096 Jul 14 15:50 lib/
drwx----- 2 root root 16384 Jul 14 15:20 lost+found/
drwxr-xr-x  5 root root  4096 Mar 16  2002 mnt/
drwxr-xr-x  3 root root  4096 Jun 10 10:40 opt/
dr-xr-xr-x 36 root root      0 Oct 12  2004 proc/
drwx--x--- 10 root root  4096 Oct  4 16:24 root/
drwxr-xr-x  2 root bin   4096 Jun  7 05:48/sbin/
drwxr-xr-x  2 root root  4096 May 12 06:03 sys/
drwxrwxrwt  8 root root  4096 Oct  4 16:24 tmp/
drwxr-xr-x 17 root root  4096 May 28  2002 usr/
drwxr-xr-x 12 root root  4096 Jun  5 23:46 var/
```

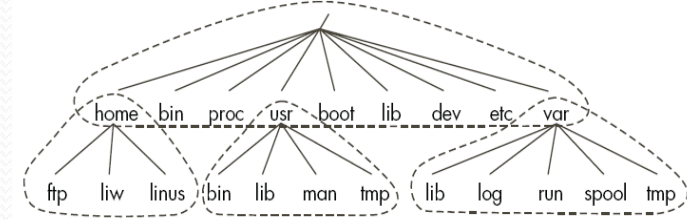
# Kořenový svazek

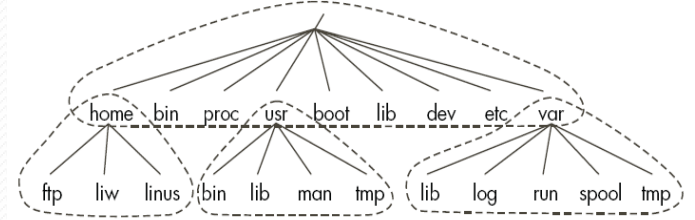


- Kořenový svazek je specifický pro každý počítač.
- Obecně uložen na lokálním disku
  - může být ramdisk, nebo síťová disková jednotka
- Kořenový svazek obsahuje soubory nutné pro zavedení systému a jeho uvedení do stavu, ve kterém mohou být připojené ostatní souborové systémy.
- Obsah postačuje pro práci v jednouživatelském režimu.
- Obecně by kořenový adresář neměl obsahovat žádné soubory
  - Může obsahovat standardní obraz systému (/vmlinuz)

# Kořenový adresář /

- /home
  - Obsahuje domovské adresáře uživatelů
- /root
  - Domovský adresář superuživatele
  - Nepřístupný ostatním uživatelům.
- /lib
  - Sdílené knihovny pro programy v kořenovém souborovém systému.





## • /lib/modules

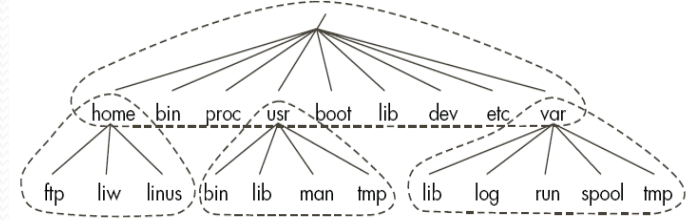
- Zaveditelné moduly jádra systému – zvláště ty, které jsou potřeba pro zavedení systému při zotavení po havárii (například síťové ovladače a ovladače pro souborový systém).

## • /dev

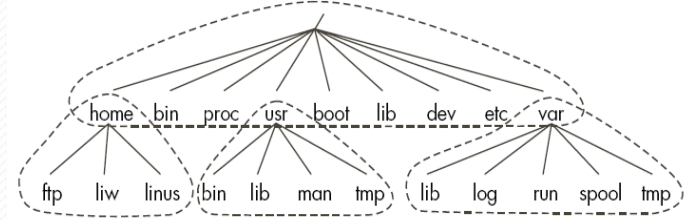
- Soubory zařízení.

## • /tmp

- Dočasné soubory. Programy, které se spouští až po zavedení systému, by měly používat adresář /var/tmp



- /bin
  - Běžné programy
- /sbin
  - Stejně jako /bin, příkazy ale nejsou určeny pro normální uživatele, i když i ti je mohou použít, je-li to nutné a je-li to povoleno.
  - Tento adresář se obvykle nenachází v cestě normálních uživatelů, má jej ale v cestě root.



- /boot

- Soubory, jež používá zavaděč operačního systému, např. LILO. Často se zde ukládají obrazy jádra (místo v kořenovém adresáři).

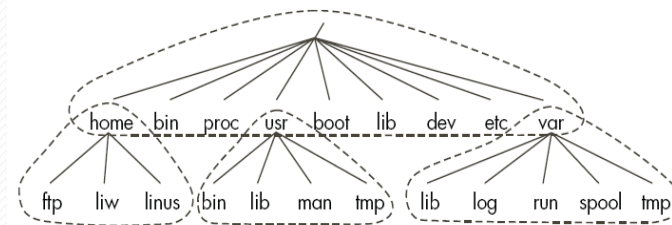
- /mnt

- Přípojně místo pro dočasná připojení dalších systémů souborů správcem systému.

- Přípojná místa pro další souborové systémy

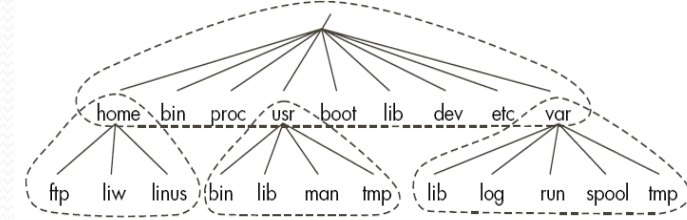
- /proc
- /usr
- /var

# /etc



- uloženy konfigurační soubory
- /etc/rc nebo /etc/rc.d nebo /etc/rc?.d
  - Skripty nebo adresáře skriptů, které se spouští při startu, nebo v případě, že se mění úroveň běhu systému.
- /etc/passwd
  - Databáze uživatelů systému.
- /etc/shadow.
  - Stínová hesla, přístup má pouze superuživatel.
- /etc/group
  - popisuje pracovní skupiny.

# /etc

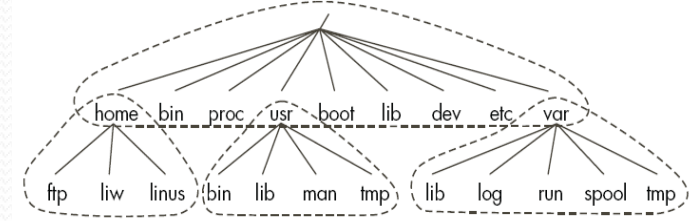


- /etc/fstab
  - Seznamy souborových systémů připojovaných automaticky při startu včetně informací o odkládacích oblastech.
- /etc/inittab
  - Konfigurační soubor **procesu init**.
- /etc/issue
  - Soubor obsahuje výstup programu **getty**.

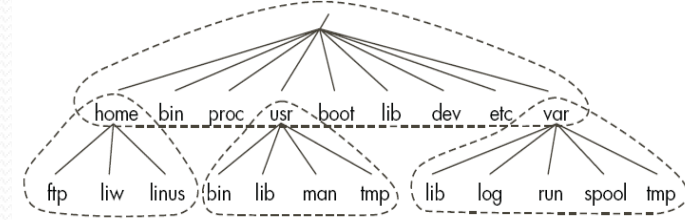


# /etc

- /etc/mtab
  - Seznam připojených souborových systémů.
- /etc/profile
  - Soubor spouštěný při přihlášení uživatele
  - Společné konfigurační nastavení
- /etc/lilo.conf
  - Konfigurační soubor programu lilo



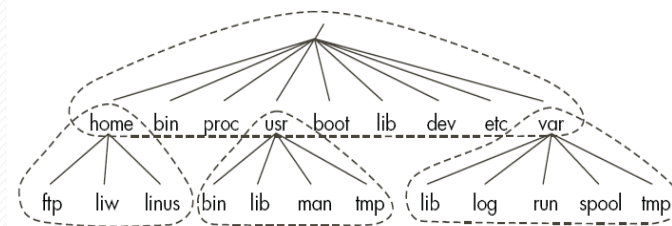
# /usr



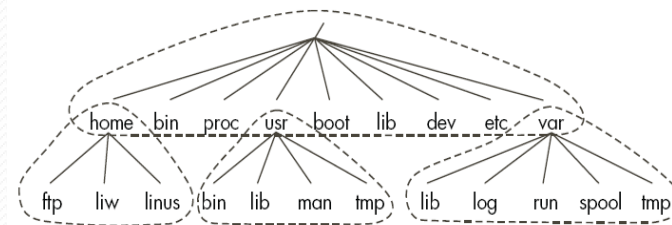
- Souborový systém /usr obsahuje všechny příkazy, knihovny, manuálové stránky a jiné soubory, jejichž obsah se nemění a které uživatel potřebuje při běžném provozu.
- možnost efektivně sdílet v síti. Sdílení tohoto svazku je výhodné jak z hlediska nákladů - šetří se tím místo na disku, tak z hlediska usnadnění správy systému.
  - Žádný ze souborů svazku /usr by neměl být specifický pro některý počítač.
  - Neměl by se při normálním provozu měnit.
- Je-li souborový systém /usr na lokálním disku, může být připojen pouze pro čtení. To snižuje pravděpodobnost poškození systému souborů při havárii systému.

# /usr

- /usr/local
  - Místo pro lokálně instalované soubory.
- /usr/X11R6
  - Všechny soubory systému X Window.
- /usr/bin
  - Zde se nachází uživatelské příkazy. Některé další příkazy jsou uloženy v adresáři /bin nebo /usr/local/bin.
- /usr/sbin
  - Obsahuje příkazy pro správu systému.

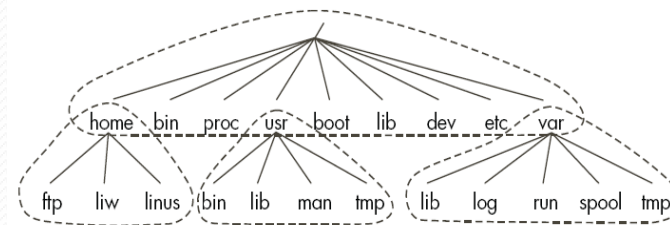


# /usr



- /usr/share/man, /usr/share/info, /usr/share/doc
  - Manuálové stránky, informační dokumenty GNU
- /usr/lib
  - Datové soubory pro programy, které se nemění. Některé globální konfigurační soubory
- /usr/include
  - Hlavičkové soubory pro programovací jazyk C.

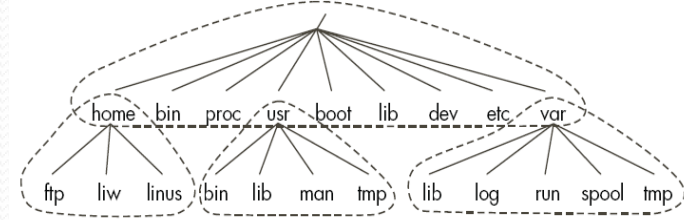
# /var



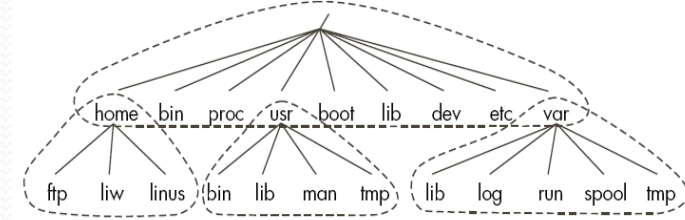
- /var/local
  - Měnící se data pro programy instalované v adresáři /usr/local
- /var/log
  - Adresář obsahuje logovací soubory programů
    - **login** (/var/log/wtmp).
    - **syslog** (/var/log/messages).
- /var/spool (/var/mail, /var/spool/mail)
  - Adresáře pro elektronickou poštu
- /var/tmp
  - dočasné soubory

# /proc

- Ve skutečnosti na disku neexistuje.
  - Vytváří ho v paměti jádro systému.
- Lze získávat různé aktuální informace o systému (procesech)
- Soubory souborového systému /proc nezabírají ve skutečnosti vůbec žádný diskový prostor.

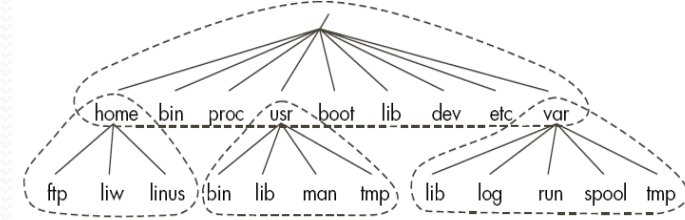


# /proc



- /proc/1
  - Adresář s informacemi o procesu číslo 1.
- /proc/cpuinfo
  - Různé informace o procesoru. Např. typ, výrobce, model, výkon a podobně.
- /proc/devices
  - Seznam ovladačů zařízení pro běžící jádro.
- /proc/kmsg
  - Výstupní hlášení jádra systému.
- /proc/filesystems
  - Souborové systémy konfigurované v jádru systému.

# /proc



- /proc/kcore
  - Obraz fyzické paměti systému. Má velikost odpovídající velikosti fyzické paměti systému.
  - generovaný „na požádání“, když programy přistupují.
- /proc/meminfo
  - Informace o využití paměti, jak fyzické, tak virtuální.
- /proc/modules
  - moduly jádra právě zavedené v paměti.
- /proc/uptime
  - Informuje o tom, jak dlouho systém běží.
- /proc/version
  - Verze jádra systému.



# Zařízení - devices

- Každé zařízení v systému má svoje označení, které koresponduje se souborem v adresáři /dev
- seznam souborů zařízení najdete v souboru /usr/src/linux/Documentation/devices.txt
- Od jádra 2.6 se soubory zařízení tvoří dynamicky
  - Využívá se démon **udev**
    - v uživatelském prostoru

# Seznam zařízení

- /dev/dsp                (/dev/snd)
  - Digital Signal Processor.
- /dev/fd0
  - První disketová mechanika.
- /dev/hda
  - Master disk na primárním IDE řadiči.
- /dev/sda
  - SCSI zařízení

- **/dev/null**
  - Černá díra
- **/dev/psaux**
  - Port myši PS/2.
- **/dev/random a /dev/urandom**
  - Generátory náhodných čísel.
- **/dev/zero**
  - Generátor nul.
- **/dev/lpo**
  - První paralelní tiskárna.
- **/dev/usb/**
  - Zařízení USB

# Práce s adresáři

- Všechno je soubor.
  - Adresáře jsou spec. Typem souborů.
  - V Linuxu jsou i zařízení reprezentovány souborem
- **pwd, cd, ls, mkdir, rmdir**
- / . .. ~
- <https://www.kernel.org/doc/man-pages/>

(Linux Community, 2019)

# Práce s adresáři

- **pwd**
  - Zobrazí aktuální adresář (Print Working Directory nebo Present Working Dir)
- **cd**
  - Změna adresáře (Change Directory).
  - `cd ~` Přechod do domovského adresáře. Jde použít pouze *cd*.
  - `cd ..` Přechod o adresář výše (rodičovského(parent)/nadřazeného adresáře).
  - `cd -` Přechod do předchozího adresáře.
  - **Absolutní a relativní cesta.**
    - Absolutní cesta vždy začíná v kořenovém adresáři, to znamená, že začíná /
    - Relativní cesta začíná v aktuální pracovním adresáři.
- **ls**
  - Výpis obsahu adresáře (List the contents of a directory).
  - `ls -l` Výpis obsahu adresáře v dlouhém formátu (nejčastěji používané)

(Linux Community, 2019)

# Práce s adresáři

- **mkdir**
  - Vytvoření adresáře
- **rmdir**
  - Odstranění adresáře. Adresář musí být prázdný.
    - vysvětlení na přednášce.
- **rm**
  - Odstranění souboru nebo adresáře.
  - S volbou *-r* , *-R* nebo *--recursive* odstraní i neprázdné adresáře rekurzivně.

(Linux Community, 2019)

# Práce se soubory

- **cp**
  - Kopíruje soubory a adresáře.
- **mv**
  - Přesun souborů. Lze použít i pro přejmenování.
- **rename**
  - Přejmenování souborů.
- **ln**
  - Vytváří tvrdé odkazy na soubory.
  - Symbolický odkaz s volbou (-s or --symbolic)
- **which**
  - Zobrazí umístění spustitelného souboru.

(Linux Community, 2019)

# Práce se soubory

- **file**
  - Určí typ souboru podle obsahu
- **touch**
  - Mění časovou známku. Když soubor neexistuje, tak vytvoří prázdný.
- **dd**
  - Převod a kopírování souborů
- **df**
  - Zobrazí volné místo na systémech souborů
- **du**
  - Odhad množství zabraného místa

(Linux Community, 2019)



# Práce s textem

- Výpisy
  - echo – jednoduchý výpis na obrazovku
  - printf – formátovaný výpis
  - cat – spojování a výpis souborů
  - tac – jako cat, ale pozpátku
  - more – stránkovaný výpis
  - less – stránkovaný výpis po řádcích
  - head – vypíše pouze začátek souboru
  - tail – vypíše pouze konec souboru

(Bruchanov, 2012 ; Linux Community, 2019)

# Práce s textem

- Porovnání
  - cmp – porovnání souborů po bytech
  - diff – vypíše rozdíl dvou textových souborů
  - sdiff – sloupcové porovnání
- Řazení
  - sort – řazení
  - tsort – topologické řazení
  - uniq – vynechání opakujících se řádků
  - shuf – náhodné rozházení vstupu po řádcích

(Bruchanov, 2012 ; Linux Community, 2019)

# Práce s textem

- Vyhledávání
  - grep – vyhledávání v souboru, základní reg. výrazy
  - egrep – vyhledávání v souboru, rozšířené reg. výrazy
- Editace po řádcích
  - cut – rozdělení řádku podle sloupců
  - paste – spojení souborů po řádcích
  - rev – obrácení znaků na řádcích
  - basename – zjištění jména souboru z kompletní cesty
  - dirname – zjištění adresáře z kompletní cesty

(Bruchanov, 2012 ; Linux Community, 2019)

# Práce s textem


- Rozdělování
  - split – rozdělí soubor na kousky
  - csplit – rozdělí soubor podle obsahu
- Pomocné
  - nl – výpis souboru s očíslovanými řádky
  - wc – počítání znaků a řádků vstupu
  - hexdump – hexadecimální výpis souboru
  - strings – výpis textových řetězců z binárního souboru

(Bruchanov, 2012 ; Linux Community, 2019)


# Přístupová práva

- Přístupová práva se ukládají do 12 bitů.
- Všechny kombinace jsou v přístupových právech možné
  - ne všechny jsou však významné.

- ➡ • Vlastník
- ➡ • Skupina
- ➡ • Ostatní



-r--r--r--	1	root	root
drwxr-xr-x	2	root	root
-rw-r--r--	1	root	root
-rw-r--r--	1	root	root
drwxr-xr-x	2	root	root
drwxr-xr-x	6	root	root
drwxr-xr-x	2	root	root



## Typy souborů

**d** znamená adresář

- soubor

l symbolický link

**b, c, s, p, ...**

# Přístupová práva

# příkaz chmod

- Právo **r**
  - Viditelný obsah adresáře (požadováno právo vstupu **x**)
  - Číst soubory
- Právo **w**
  - Vytvářet a mazat soubory v adresáři
    - Vlastníkem souboru je jeho tvůrce.
  - Editovat soubory
- Právo **x**
  - Vstup do adresáře
  - Označení pro spustitelný soubor
- **SUID** (Set User ID) a **SGID** (Set Group ID)
  - Spouštění programu s efektivním UID (GID) vlastníka.
    - Význam pro binární soubory
  - SGID pro adresář : nově vytvořený soubor má nastaveno vlastnictví skupiny na skupinu vlastníci adresář.
    - Používáno při sdílení souborů
- **Sticky bit**
  - Po ukončení příkazu zůstává v systémové paměti.
  - V adresáři může soubory měnit pouze jejich vlastník.
    - /var/tmp

# Implicitní přístupová práva, Příklad

Nastavení masky	Vytvoření objektu	Přidělená práva
-----------------	-------------------	-----------------

umask 022

touch soubor	rw-r--r--
mkdir adresář	rwxr-xr-x

umask 27

touch soubor	rw-r-----
mkdir adresář	rwxr-x---

# Práce s procesy

- top, htop, ps, pstree, kill, pgrep, killall
- nice, renice, nohup
- jobs, fg, bg
- CTRL+Z                      Stop the process
- &      ampersand
  - When a line ends with an ampersand **&**, the command is executed in background
- && double ampersand (logical AND)
  - the 2<sup>nd</sup> command is executed only if the first one succeeds (returns a zero exit status)
- || double vertical bar (logical OR)
  - The 2<sup>nd</sup> command is executed only when the first command fails (returns a non-zero exit status)

*\$ echo first || echo second ; echo third*

*\$ rm file1 && echo It worked! || echo It failed!*

(Cobbaut, 2015; Linux Community, 2019)



# Archivace a komprimace

- gzip (GNU zip), bzip2
- gunzip
- bzip2recover
- tar
  - Append
  - Create
  - Compare
  - Delete
  - List
  - Update
  - Extract

(Cobbaut, 2015; Linux Community, 2019)

# Skriptování

- Textový soubor obsahující příkazy
- Argumenty, proměnné, návratové kódy
- Čtení standardního vstupu
- **Řídící příkazy**
  - test
  - if, case
  - while, until, for
  - break, continue
  - exec, trap

Tab. 13: Proměnné používané ve skriptech [5, 6, 7, 8]

Proměnná	Popis
\$#	Počet argumentů.
\$0	Jméno příkazu.
\$1, \$2, ..., \$9	Jednotlivé argumenty příkazu (poziční parametry). Další jsou přístupné pomocí příkazu <i>shift</i> , případně pomocí uvedení čísla ve složených závorkách <code>\${10}</code> (pouze <u>Bash</u> ).
\$*	Seznam argumentů chápaný jako jedno slovo.
\$@	Seznam všech argumentů chápaný jako posloupnost slov.
\$?	Návratový kód posledního příkazu.
\$\$	Identifikace aktuálního procesu.

# Příkaz test

Forma argumentu	Funkce	Forma argumentu	Funkce
<code>-d <u>adresar</u></code>	Určený <u>adresar</u> existuje a je adresářem	<code><u>c1</u> -eq <u>c2</u></code>	Číslo <i>c1</i> je rovno číslu <i>c2</i>
<code>-e <u>soubor</u></code>	Určený <u>soubor</u> existuje	<code><u>c1</u> -ge <u>c2</u></code>	Číslo <i>c1</i> je větší nebo rovno číslu <i>c2</i>
<code>-r <u>soubor</u></code>	Určený <u>soubor</u> existuje a máte právo čtení	<code><u>c1</u> -gt <u>c2</u></code>	Číslo <i>c1</i> je <u>větší</u> než číslu <i>c2</i>
<code>-s <u>soubor</u></code>	Určený <u>soubor</u> existuje a má nenulovou velikost	<code><u>c1</u> -le <u>c2</u></code>	Číslo <i>c1</i> je menší než číslu <i>c2</i>
<code>-w <u>soubor</u></code>	Určený <u>soubor</u> existuje a máte právo zápisu	<code><u>c1</u> -lt <u>c2</u></code>	Číslo <i>c1</i> je menší nebo rovno číslu <i>c2</i>
<code>-x <u>soubor</u></code>	Určený <u>soubor</u> existuje a máte právo spouštění	<code><u>c1</u> -ne <u>c2</u></code>	Číslo <i>c1</i> se nerovná číslu <i>c2</i>
<code>-L <u>soubor</u></code>	Určený <u>soubor</u> existuje a je symbolickým odkazem	<code>! <u>výraz</u></code>	Logická negace výrazu
<code><u>s1</u> -nt <u>s2</u></code>	Soubor <i>s1</i> je novější než soubor <i>s2</i>	<code><u>výraz</u> -a <u>výraz</u></code>	Logický součin. Oba výrazy musí být pravdivé, aby výsledek byl pravdivý
<code><u>s1</u> -ot <u>s2</u></code>	Soubor <i>s1</i> je starší než soubor <i>s2</i>	<code><u>výraz</u> -o <u>výraz</u></code>	Logický součet. Alespoň jeden z výrazů musí být pravdivý, aby výsledek byl pravdivý.
<code>-n <u>r1</u></code>	Řetězec <i>r1</i> má nenulovou délku		
<code><u>r1</u> = <u>r2</u></code>	Řetězec <i>r1</i> je shodný s řetězcem <i>r2</i>		
<code><u>r1</u> != <u>r2</u></code>	Řetězec <i>r1</i> není shodný s řetězcem <i>r2</i>		

Tab. 14: Vyhodnocovací kritéria příkazu *test*.

(Cobbaut, 2015; Sysel, 2006; Linux Community, 2019)