

1. Základní struktura OS

- **jádro (kernel)**
 - obaluje HW
 - existují dva druhy architektur
 - monolitická (linux, BSD,...)
 - mikrokernelová / distribuovaná (mach,...)
 - jádro není tvořeno jen jedním velkým binárem, ale menšími separovanými bináry
 - je to jediné, co je opravdu linux
 - GNU / Linux = unixový OS = linux + nástroje,... [GNU is Not Unix]
 - vše, co je v jádře, běží v módu jádra (ne v uživatelském)
 - lze se do něj dostat pomocí příkazu „syscall“
 - důležité části:
 - **scheduler** [TODO kolem 12:00]
 - **virtual memory**
 - oddělené paměťové prostory, aby si úlohy navzájem neškodily
 - je ale možné přidělit jim i paměť společnou (shared memory)
 - efektivní pro jejich komunikaci
 - systém občas používá overcommit (slíbí více paměti než má)
 - MMU (memory management unit) pak řeší přiřazení apod.
 - brk(x) = funkce, která vrací paměť; x = požadovaná velikost
 - nesmí přistupovat k nule (adresa) a kolem ní
 - jinak trestá segmentation faultem
 - **drivers (ovladače)**
 - bývají modularizovány (nejsou v jádru, stáhnou se dle potřeby)
 - ACPI – řídí spotřebu částí HW
 - má je jen v initial ramdisku
 - načte si potřebné, zbytek zahodí
 - kromě disků i USB, klávesnice, zvukovka,...
 - **file-systém**
 - nadstavba nad ovladač disku
 - přistupuje k nim na úrovni souborů, adresářů atd.
 - časté ext4
 - fsck = kontrola systému, obdoba chkdsk
 - žurnál = pomocná struktura, obsahuje otisk operací
 - **lib (knihovny)**
 - moduly pro enkrypci, počítací, bezpečnostní,...
 - většinou je užívají ostatní moduly jádra
- **userspace (uživatelský prostor)**
 - **lib (knihovny)**
 - např. libc

- **systémové utility**
 - jednoduché programky
 - běží jen v textovém režimu
 - coreutils, binutils,...
- **shell**
 - klasický shell je sh (bourne shell) – problémy s licencí
 - bash (bourne again shell) – náhrada shellu
 - csh (podobné jazyku C)
 - ksh, zsh,...
 - přijímá naše příkazy z klávesnice a vykonává je
 - spustitelné programy hledá v sobě, pak v \$PATH (cat,...)
- **systémové aplikace**
 - tiskový systém,...
- **grafická nadstavba**
 - nejnižší vrstva – xserver

2. Struktura filesystemu, práva

Kořenový adresář

- / = kořenový adresář
- každý unixový systém má jediný kořenový adresář
- má mnoho podadresářů
- nejsou v něm soubory, jen podadresáře
- **bin**
 - (binaries) = binární/spustitelné programy, popř. skripty
 - některé nutné pro start systému
 - soubory se nerozlišují dle přípony
 - ls, shell,...
- **boot**
 - obrazy jádra OS apod.
 - důležité pro start OS
- **dev**
 - (devices) = zařízení
 - odkazy na fyzická zařízení PC
- **etc**
 - konfigurační a další pomocné soubory, startup skripty pro automatické spuštění
 - passwd, shadow,...

- **home**
 - jména a domovské adresáře všech uživatelů systému
- **lib**
 - (libraries) = systémové knihovny (libc,...)
 - kriticky důležité, bez něj nejdou příkazy
- **lost+found**
 - má být prázdný, admin by jej měl po startu zkontrolovat
 - kusy FS po haváriích systému (#čísloinode)
- **mnt**
 - možnost připojit jiný souborový systém
- **opt**
 - (optional) = volitelné doplňky OS
- **proc**
 - (process)
 - většinou pseudoadresář, vytváří se automaticky za běhu systému
 - obsahuje podadresáře procesů (např. init = /proc/1)
- **sys**
 - také pseudoadresář
 - také info o důležitých procesech systému
 - částečná migrace z proc
- **sbin**
 - stejné jako bin, ale pro superuživatele
- **tmp**
 - dočasné soubory
 - každý uživatel do něj má práva zápisu
 - uživatelské soubory může mazat jen daný uživatel
- **usr**
 - obsahuje zrcadlo kořenového adresáře (bin, lib, sbin,...) a další soubory
 - manuály, dokumentace,...
- **var**
 - obsahuje proměnné
 - systémové logy, fronty (např. tisku),...

Další info

- příklad výpisu z ls:
 - o drwxr-xr-x¹ root² root³ 4096⁴ <datum>⁵ bin⁶

¹ = typ souboru a přístupová práva

² = vlastník

³ = skupina vlastníka

⁴ = velikost souboru (v blocích)

⁵ = čas poslední modifikace

⁶ = jméno adresáře

Oprávnění

- rwxr-xr-x ... rozdělení na tři části (každá tři písmenka)
 - o U = ostatní (uživatelé)
 - o G = skupina (souboru)
 - o O = owner
- různé **míry oprávnění**
 - o **r** = read = právo na čtení
 - o **w** = write = právo na zápis
 - o **x** = executable = spustitelný soubor, popř. u adresáře že lze přistupovat k jeho položkám
 - o **s** = nový soubor bude mít stejnou skupinu jako vlastník adresáře
 - o **t** = sticky bit, zůstane v paměti, dokud ho nevyhodíme
- pokud je u adresáře pouze „x“, ale není „r“, tak např. příkaz ls selže
 - o když ale známe spustitelný soubor uvnitř, lze jej spustit
- **typy souboru:**
 - o „-“ = běžný soubor
 - o d = adresář
 - o l = symbolický link
 - obsahuje odkaz na skutečné umístění souboru
 - např. cat --> /home/pets/cat
 - u hardlinku nelze linkovat adresáře, data musí být na stejném disku
 - o b = block device
 - představuje fyzické zařízení, nejčastěji disky
 - má domov v /dev
 - o c = character device
 - klávesnice, myši, konzole,...
 - má domov v /dev
 - o s = socket
 - souborové vyústění síťové operace
 - o p = pipe (potrubí)
 - z jednoho souboru do něj lze zapisovat, z druhého číst