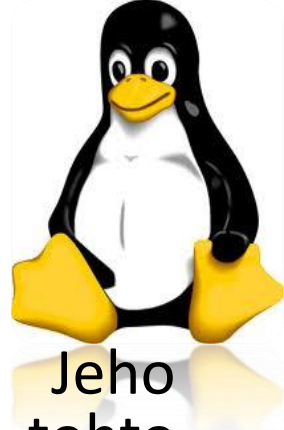




LINUX

1. GNU



Tento projekt bol začatý v roku 1984. Jeho zakladateľom bol Richard Stallman. Komponenty tohto projektu sú šírené pod licenciou GPL a .LGPL. Cieľom tohto projektu bolo vytvoriť voľne šíriteľný OS postavený na UNIX-ovej filozofii.

Kedže pôvodné jadro nebolo doteraz dokončené, používa sa dnes systém GNU spolu s jadrom linuxu, ktoré začal Linus Torvalds. Mimo odbornú verejnosť sa toto spojenie chybne nazýva Linux, ale jeho celý názov je GNU/Linux

Čo znamená GNU?

GNU je rekurzívny akronym pre vetu "GNU is not Unix" (GNU nie je Unix). GNU však tiež znamená *pakôň hrivnatý*. Čo vystihuje aj logo

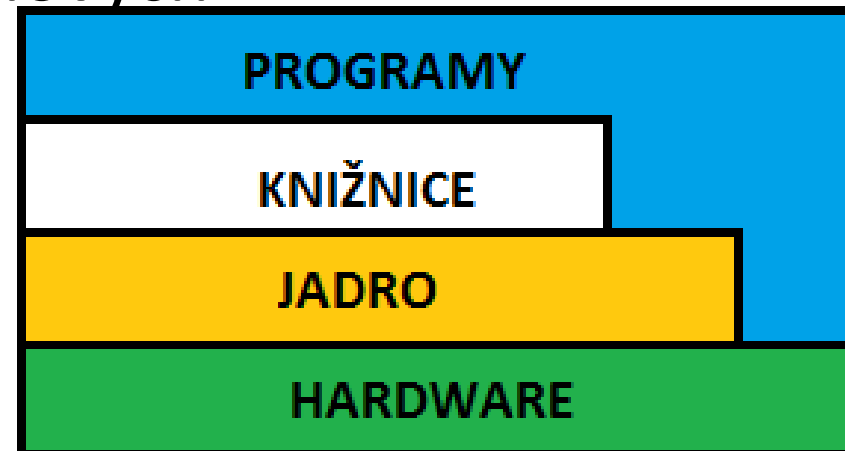


2. Zloženie OS

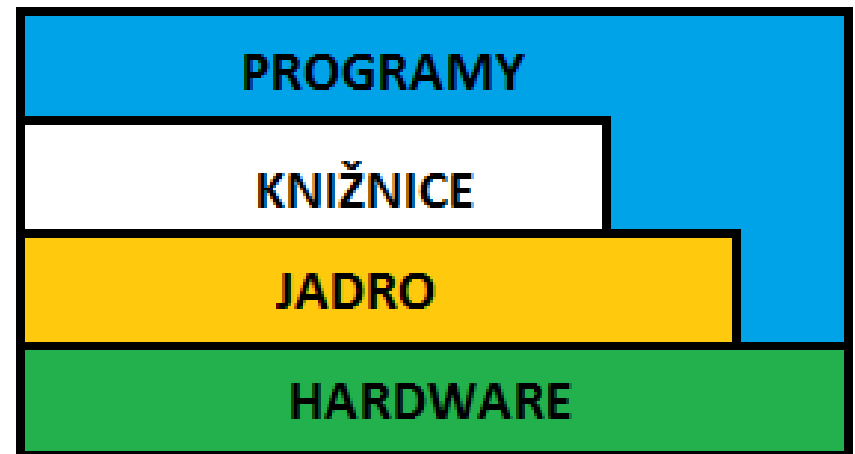
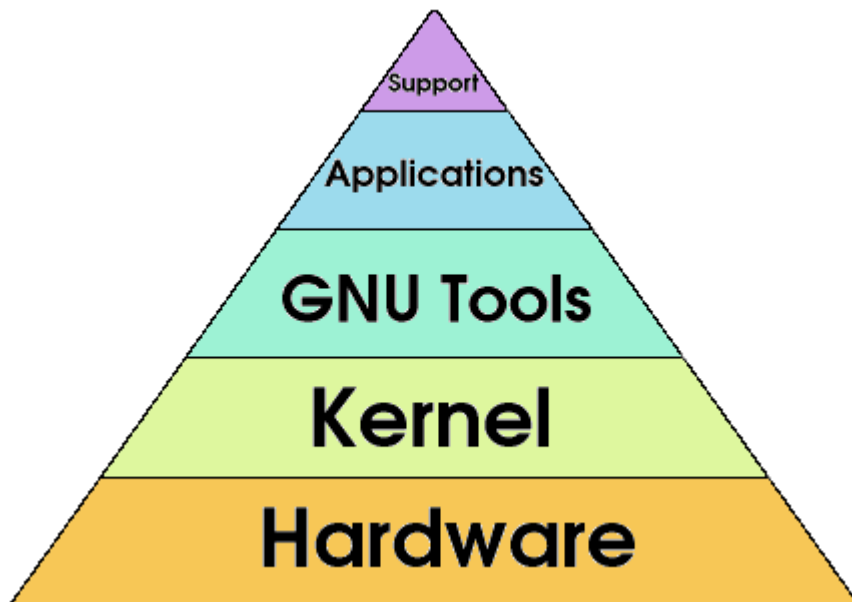
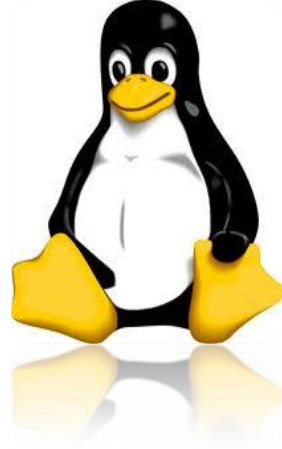


Operačný systém sa skladá z niekoľkých častí. Najhlbšie je jadro, to sa stará o spoluprácu s hardwarom (pamäť, procesor, pevné disky, a pod.) a poskytuje rôzne služby *procesom*. Ako vidíte na obr.1 väčšina programov volá rôzne funkcie knižníc, ktoré sa potom predávajú jadru. Viac menej niektoré programy potrebujú prístupovať k jadru priamo pomocou jeho systémových

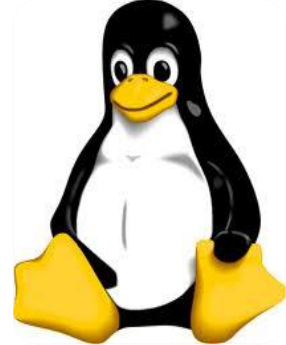
volaní. A existuje aj skupina programov ,ktorá vyžaduje priamy prístup k hardwaru.



2. Zloženie OS

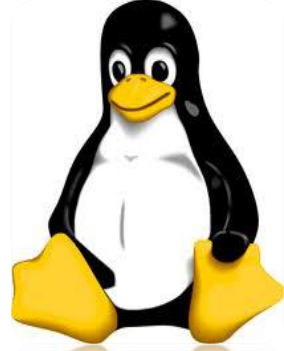


2.1 Jadro



Jadro je srdcom OS. Zaistúje komunikáciu s hardwarom a poskytuje aplikáciám svoje služby ako správu procesov, pamäti, súborových systémov, podporu sietí a pod. Ku komunikácii s hardwarom slúžia ovládače. V pôvodných verziách linuxu bolo jadro monolitické čo znamená, že pre získavanie podpory iného zariadenia sa jadro muselo prekompilovať. Dnešné jadro je modulárne, potrebné ovládače sa vo forme modulov môžu do jadra nahráť, prípadne z neho aj odstrániť.

2.2 Knižnice



Anglicky libraries, sg. library, sú kľúčovou súčasťou OS. Vďaka tomu, že je GNU/Linux *open source*, veľké množstvo kódu je práve v knižniciach. Ide predovšetkým o štandardné veci ako sú napr.: operácie so súbormi .jpeg, alebo matematické funkcie ako sin, ale existujú aj knižnice GTK a Qt, slúžiace pre vykresľovanie tlačítok a ďalšieho rozhrania. Nikdy teda nezabúdajme na knižnice inak sa môžete dostať do problémov.



2.3 Program, proces



Pojmy *program* a *proces* sa často pletú.

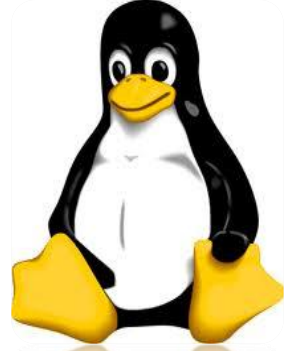
Program je súbor na disku, ktorý si môžete spustiť a on bude vykonávať niaku činnosť. Keď ho spustíte, nahrá sa do operačnej pamäte a tým sa z neho stane **procesom**. Jeden program (napr. textový editor) môže teda byť spustený viacerými užívateľmi. Každý z nich má ale vlastný proces. Skúste si nechať vypísať zoznam procesov:

ps

Toto však vypíše len procesy v aktuálnom shelly. Pre vypísanie všetkých procesov v pamäti napíšete:

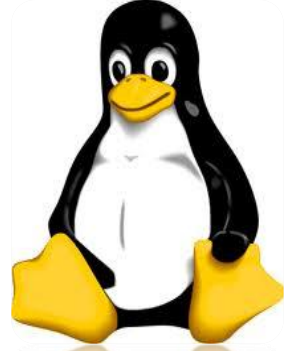
ps - e

3. Súborový systém



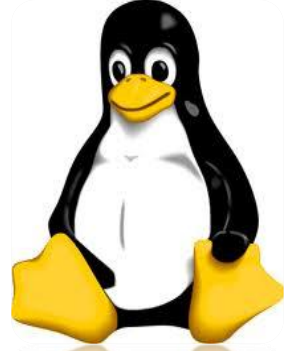
Ako vo väčšine OS vychádzajúcich z UNIXU, je aj v linuxe len jeden koreňový adresár označený '/', . Všetky ďalšie súborové systémy sa pripojujú do jedného stromu, takže ak chceme napr.: pristupovať k súborom na diskete , tak ju pripojíme do adresára /mnt/floppy a pokiaľ máme oddelený diskový oddiel pre adresáre užívateľov, tak ho pripojíme do adresára /home

3. Súborový systém



Súborové systémy sa pripojujú príkazom ***mount*** a odpájajú príkazom ***umount***. Aby sme nemuseli ručne pripájať pri každom spustení systému všetky disky a pri často pripojovaných a odpojovaných súborových systémov, je v ***/etc/fstab*** uvedené čo a ako sa kam pripojuje. Ale väčšina distribúcií nastaví pri inštalácii všetko správne a my sa tak nemusíme o nič starať – jadro je upravené tak, aby automaticky pripojovalo všetky zariadenia.

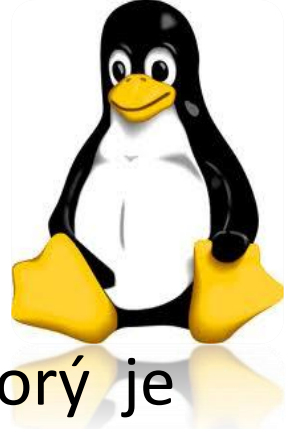
3. Súborový systém



Štruktúra súboru `/etc/fstab` je popísaná v jeho manuálovej stránke (`man fstab`). Príkaz `mount` a parametre špecifické pre jednotlivé súborové systémy sú popísané v manuálovej stránke príkazu `mount`.



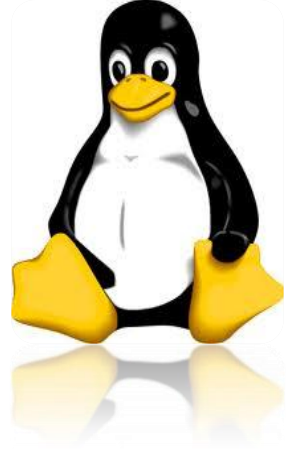
3. Súborový systém



Hierarchia - štruktúra linuxového filesystemu, ktorý je veľmi podobný tradičnému Unixovému, špecifikuje tzv. **FHS** (Filesystem Hierarchy Standard). Tento štandard špecifikuje v akom adresári majú byť aké aplikácie, knižnice či iné súbory.

Kontrola integrity filesystemu – o kontrolu súborového systému sa nemusíme starať, spúšťa sa automaticky pri boote. Programom `tune2fs` môžeme ľahko ovplyvniť v akých periódach bude ku kontrole dochádzať. Rozhodne sa nedoporučuje spúšťať samotný `fsck` za chodu systému, presnejšie povedané robiť s ním kontrolu oddielu.

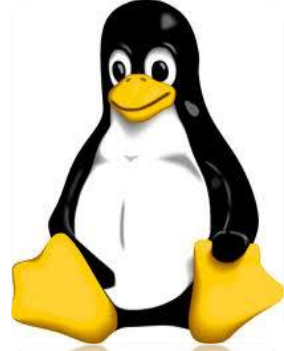
3. Súborový systém



Predchádzanie chybám

Je asi samozrejmé, že PC vypíname pomocou príkazu `halt` alebo `shutdown`, či pomocou tlačítka v grafickom rozhraní. V prípade, že by došlo k silnejšiemu zatuhnutiu, pred stiskom tlačítka reset je vhodné skúsiť `Ctrl + Alt + SysRq + F1`, a potom `Ctrl + Alt + SysRq + S`. Tým dôjde k synchronizácii disku .

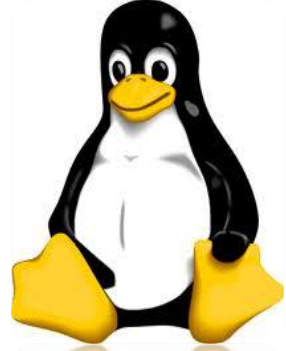
4. Procesy



Na používateľskej úrovni sú súbory identifikované svojím menom (ktoré zodpovedá menu spustiteľného súboru), ale systém ich jednoznačne identifikuje číslom. Ako je vidieť na časti príkazu *top*.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
13743	root	15	0	98.4m	21m	1660	S	12.3	8.5	1:52.07	X
14057	misak	15	0	61884	45m	19m	S	4.3	9.0	0:59.27	konqueror
14047	misak	15	0	80420	36m	22m	S	2.3	7.5	1:14.01	amarokapp

4. Procesy



Prvým procesom, ktorý je spustený a má poradové číslo 1 je **init**. Všetky ďalšie procesy sú jeho „potomkovia“ a jeho vypnutie vedie k ukončeniu celého systému.

Jeho hodnota býva najčastejšie 32768. Skutočná hodnota na vašom systéme je uložená v súbore `/proc/sys/kernel/pid_max`.

Pokiaľ systém dosiahne maxima, začne prideľovať voľné čísla zasa od začiatku.

5. Informácie o bežiacich procesoch



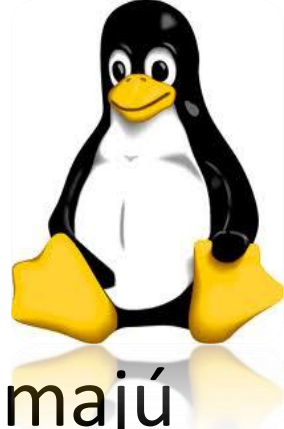
Príkaz ***ps*** bez parametrov vypíše len procesy spojené s jedným konkrétnym terminálom.

```
$ ps
  PID TTY          TIME CMD
 30482 pts/6    00:00:00 bash
   583 pts/6    00:00:00 ps
```

Voľba ***x*** zaistí vypísanie všetkých vašich procesov

```
$ ps x
  PID TTY          STAT TIME  COMMAND
 13955 ?        Ss   0:00  /bin/sh /usr/kde/3.5/bin/startkde
 13982 ?        Ss   0:00  gpg-agent --daemon
...
 32128 ?        Ss   0:01  gvim semestralka.tex
   553 ?        S    0:00  ispell -a -S -m -C -d czech
   590 pts/2    R+   0:00  ps x
```


5. Informácie o bežiacich procesoch



Pokiaľ vás zaujímajú procesy, ktoré majú spustené aj ostatné, použite voľbu **a**, ale takto by sme získali len procesy pripojené k danému terminálu. Preto musíme pridať aj voľbu **x**, ktorá zabezpečí výpis procesov všetkých užívateľov na všetkých termináloch.

```
$ ps ax
  PID TTY          STAT TIME COMMAND
    1 ?           S      0:00 init [3]
    2 ?           SN     0:00 [ksoftirqd/0]
  ...
32128 ?           Ss     0:01 gvim semestralka.tex
32525 ?           S      0:03 konsole
32526 pts/7       Ss     0:00 /bin/bash
   623 ?           S      0:00 ispell -a -S -m -C -d czech
   649 pts/6       R+     0:00 ps ax
```

5. Informácie o bežiacich procesoch



Parameter ***u*** zaistí podrobnejší výpis.

```
$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.1  1476    396 ?        S      Jan05    0:00 init [3]
root         2  0.0  0.0      0      0 ?        SN     Jan05    0:00 [ksoftirqd/0]
...
misak    32128  0.0  3.3  19084   8520 ?        Ss     18:34    0:01 gvim
semestralka.tex
misak    32525  0.1  6.5  30460  16796 ?        S      18:57    0:03 konsole
misak    32526  0.0  0.6   3544   1776 pts/7    Ss+    18:57    0:00 /bin/bash
misak      623  0.1  4.1  12088  10736 ?        S      19:16    0:01 ispell -a -S -m
-C -d czech
misak      726  0.0  0.3   2896    964 pts/6    R+     19:28    0:00 ps aux
```

Koniec