



Projekt v rámci SIPVZ:

**IMPLEMENTACE OPERAČNÍHO SYSTÉMU LINUX DO
VÝUKY INFORMAČNÍCH TECHNOLOGIÍ**

LINUX

Lekce 2

Struktura LINUXU - základ

Obsah lekce:

Cíle	1
Koncepce systému Linux	1
Modulární struktura Linuxu	
Vynucený multitasking – SW	6
Stručný popis vynuceno multistatingu v Linuxu	
Jedno-uživatelský a více-uživatelský SW	
Struktura souborového systému	8
Objekty souborového systému	
Navigační systém souborů	
Montování souborových systémů a médií	9
mount	
tec/vstav	
Abstrakce souborových systémů	
Otázky k opakování	12
Lab	13

Cíle

Po skončení této lekce studenti budou schopni:

- Charakterizovat modulovou povahu Linuxu
- Charakterizovat kernel – jádro, síť, init program, démony, loginy, shell, utility, systém X Windows
- Definovat aspekty Linuxu jako více-uživatelský systém a popsat multitasking
- Definovat strukturu souborového systému
- Popsat jak jádro pracuje s uživatelskými moduly a ovládači jednotlivých zařízení

Koncepce systému Linux

- Modulární struktura Linuxu
 - Malé zaměnitelné části tvoří celek
- Šest hlavních modulů v Linuxu
 - Kernel – jádro
 - Init program
 - Démoni (systém procesů)
 - Síť
 - Login, shells, utility
 - Systém X – Windows

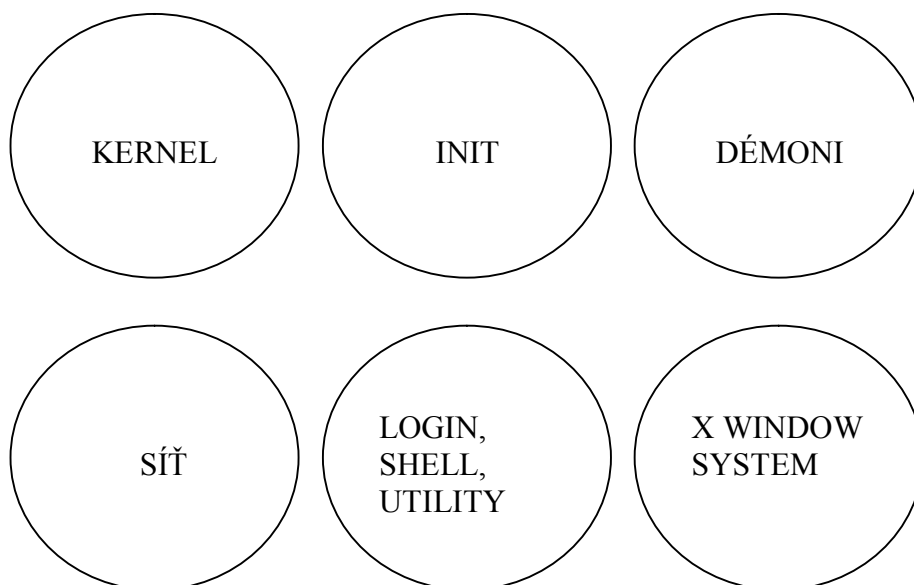
Abychom mohli efektivně implementovat Linux, uživatel musí pochopit koncepci systému Linux. Hlavní koncepcí (vlastností) systému Linux je modularita, multitasking, pro více uživatelů, souborový systém Linuxu, virtuální terminál, a příkazová řádka jako rozhraní.

Modulární koncepce Linuxu

Linux SW je založen na modulárním pojetí, proto se zaměřuje na vytvoření malých zaměnitelných částí, které dohromady tvoří celek. Modulární struktura Linuxu je dynamická ve své povaze (z principu), což znamená, že modul může být přidán nebo odebrán v průběhu bez nutnosti restartovat systém. Nejdůležitějším modulem je jádro (tzv. kernel), jenž slouží jako základna pro další moduly.

Dobrým příkladem modulární struktury je Internet. Internet je tvořen z mnoha různých, samostatných počítačů navzájem propojených vodičů nebo kabely k sadě kořenových serverů. Odstranění jednoho z kořenových serverů zamezí přístupu některým počítačům na Internet, ale na funkci Internetu jako takového to nemá vliv. Linux byl navržen podle stejného typu modularity. Jádro se použije jako základ a k němu mohou být přidávány a odebírány moduly. Šest hlavních modulů v Linuxu (Obrázek 2 - 1) je:

- kernel (je nutný)
- init soubor (je nutný)
- démoni (jsou nutní)
- síť (volitelná)
- login, shells, a utility (jsou nutné)
- X windows systém (volitelný)



Obrázek 2 – 1 Šest hlavních modulů Linuxu

1. Kernel

Kernel je jádrem operačního systému, jenž funguje jako přestupný článek (převáděč) mezi fyzickým hardware a SW. Jádro má na starost distribuovat systémové prostředky běžícímu SW. Protože kernel je nezbytný k tomu aby počítač fungoval, je nutné natažení do operační (hlavní) paměti. To je také základem pro všechny další moduly.

2. Init program

Po startu se zavede jádro, Init je modul užívaný jádrem k inicializaci základních programů uživatele. Tudíž, Init je označený jako otec všech programů. Poté co jádro dokončí svou inicializaci, spustí se Init program. Init zeširoka kontroluje systémové soubory, určuje úroveň spuštěné služby, to, které porty se otevrou, a kteří démoni se aktivují. Init pak provede (spustí) potřebné procesy.

3. Démoni (systémový proces)

Démoni jsou aplikace běžící na pozadí. Kontrolující vstupy uživatele v čekací smyčce a čekají na určité příkazy a události, které se provádí v době jejich činnosti. Když démon zjistí, že má úkol, poskytne služby na daný příkaz. Démoni nekomunikují přímo se žádným uživatelem, ale setrvávají na pozadí, dokud nejsou požádáni o poskytnutí nějaké služby. Typická konfigurace Linuxu zaměstnává minimálně 15 samostatných démonů, a může běžet současně až 50 i více démonů.

Dva typické příklady démonů jsou:

- telnetd (telnet server démon) – když je obdržena požadavek na Telnet, démon připojí k odpovídajícímu portu všechny ovládací zařízení a všechny sjednané relace
- httpd (démon hypertextového přenosového protokolu) – když je obdržena žádost o http, httpd inicializuje proces jenž splní tento požadavek.

4. Síť

Síťový modul je užívána jádrem pro předávání data přijatých síťovým hardwarem. Ten ovládá jak příchozí provoz z okolí, tak i odchozí od místního uživatele vyžadujícího službu. Modul manipuluje s oběma datovými toky a vytváří iluzi, že jsme přímo připojeni k ostatním počítačům.

5. Login, shells, utility

Loginy, shells, a utility provádějí operace jenž umožňují následující:

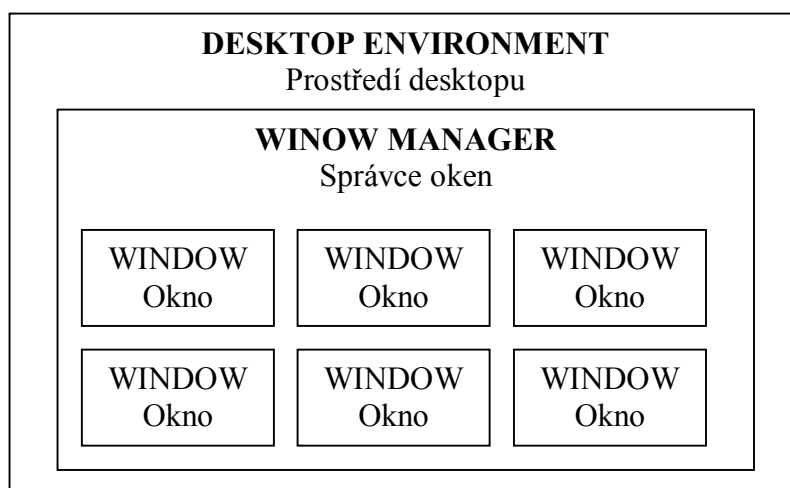
- přístup k uživatelským účtům
- konfigurace prostředí
- provedení příkazu
- konfigurace systému a aplikací

Login má na starost jména uživatelů a jejich hesla. Shelly přijímají (akceptují) a dohlíží na provádění příkazů uživatele. Utility poskytují služby na úrovni uživatele, jako je konfigurace sítě a optimalizace jádra.

Linux podporuje mnoho typů příkazových interpretů (shell), jenž jsou podobné shellům v Unixech. Nicméně, Bash shell má široký rozsah příkazů, konfiguračních voleb, a největší množství programovacích struktur, a proto je nejpoužívanějším shellem.

6. X Window systém

Systém X Window ovládá grafický HW počítače a funguje jako vrstva mezi HW a aplikacemi. X klient a X server jsou jeho dvě hlavní komponenty. X server zpracovává všechny požadavky vygenerované X klientem (aplikací). Je důležité si nespést X windows systém s window manažerem. Na systému X Windows může běžet mnoho různých typů window manažerů.



Obrázek 2 – 2 X Windows systém

Vynucený multitasking – SW

- Více programů běží na jediném CPU
 - Programy jsou pozastaveny když nastane přerušení
 - Každý program bude zpracován CPU podle priority
 - Příklad: Více potápěčů s jedinou kyslíkovou láhvi.
- Jedno-uživatelský versus více-uživatelský SW
 - Jedno-uživatelský SW nemá mechanismus pro vícenásobný přístup k datům
 - Více-uživatelský SW umožňuje více uživatelům přístup k datům na jediném systému

Původ volně šířícího se SW

Operační systémy s multitaskingem, jakým je i Linux, těží ze skutečnosti, že většina programů stráví většinu svého času čekáním na určité události. Aby nedošlo k přetížení systému multitaskingem - souběžným zpracováním úloh, OS nezatěžuje CPU těmito čekacími intervaly. Tato čekání přerušuje a vybere z fronty další program.

Zde jsou příklady programů, jenž mají čekací smyčky, které lze přerušit, a tím pádem umožnit CPU pokračovat ve vykonávání dalšího programu čekajícího ve frontě:

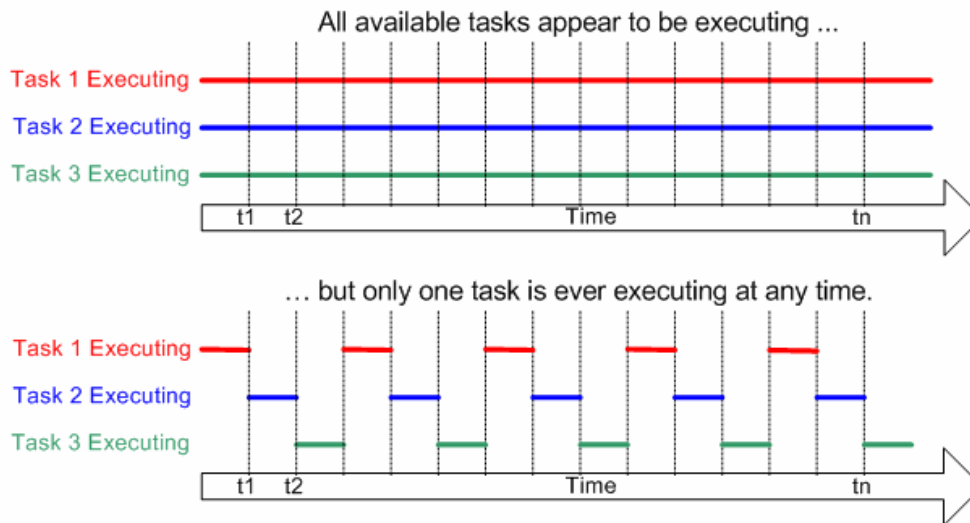
- programy čekající na uživatele aby zadal znak
- programy čekající na harddisk než přečte sektor
- programy čekající na síťový paket než se objeví

Stručný popis vynuceného multitaskingu na Linuxu

Vynucené rozvržení (rozpis, plánování) je schopnost jádra odložit programy a změnit priority tak, aby mohly běžet další programy. Multitasking je schopnost uložit program, vybrat další program, který poběží, a poté znovu obnovit předchozí program ve stavu, ve kterém byl přerušen a uložen.

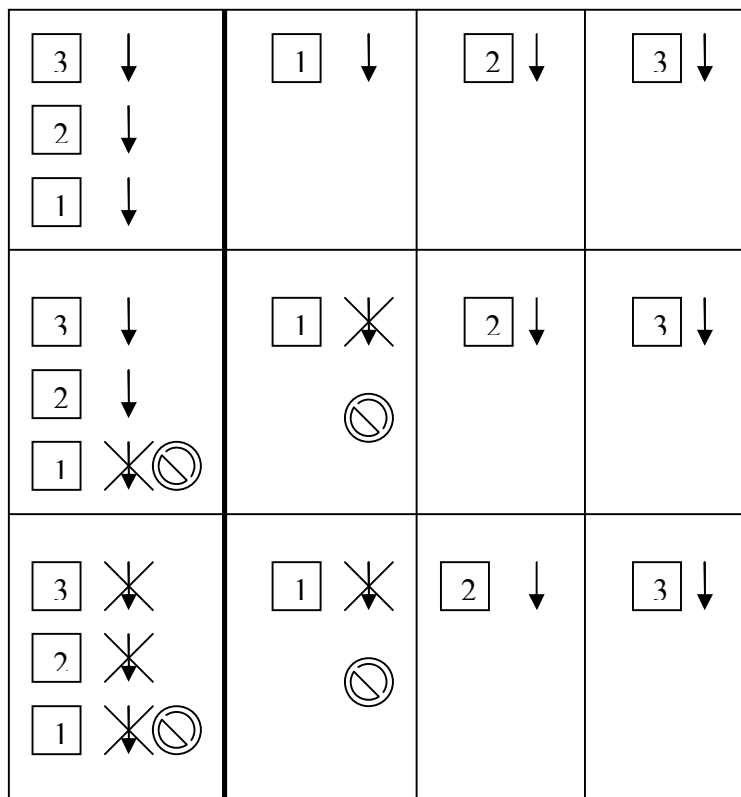
1. V době kdy Linux provádí program, v prostředí může nastat událost jakou je například jeden tik hodinek, nebo přerušení z jiného zařízení.
2. Tato událost způsobí že právě běžící program je pozastaven a dojde ke změně vůči zařízení obsluhým programem přerušení (ISR)
3. Po skončení přesunu dat, ISR může buď:
 - a) vrátit se k programu, který se prováděl před událostí v původním prostředí
 - b) nebo ignorovat tento program ve prospěch programu spojeného s jinou událostí

Je-li vybrána akce (a), pak ISR může změnit stav programu, který čekal na událost zařízení ze stavu pozastavení na stav připravenosti k běhu, zatímco CPU zpracovává původní program. Je-li vybrána akce (b), pak ISR může přepnout program ze stavu připraven do stavu běh, a zvýší prioritu programu.



Obrázek 2-3a.

Na prvním obrázku (Obrázek 2 – 3a) je příklad multitaskingu, kdy běží tři vlákna naráz. V dolním obrázku (Obrázek 2 – 3b) je vynucený multitasking, kdy sice také běží tři vlákna, ale v daný procesorový čas může běžet pouze jeden.



Obrázek 2 – 3b Tři kroky v multitaskingu

Jedno-uživatelský versus více-uživatelský SW

Jedno-uživatelské systémy mohou mít speciální programy které umožňují dvěma konkrétním osobám aby se připojili ke stroji současně. Tím pádem můžeme získat dojem, že systém je více-uživatelský operační systém, ale pro operační systém je to samostatný - jediný uživatel. Systémy tohoto druhu jsou označovány jako sekvenční (jdoucí po sobě) jedno-uživatelské systémy.

Více-uživatelský SW umožňuje libovolnému počtu uživatelům zpustit tu samou aplikaci a přistupovat ke stejným souborům současně. První kopie souboru lze otevřít i editovat, a každou další kopii lze otevřít pouze pro čtení. Jedno-uživatelský SW nedovoluje více uživatelům použít ty samé aplikace nebo přistoupit k těm samým souborům ve stejnou dobu.

Linux byl navržen jako více-uživatelský operační systém. Přístup do systému lze získat přihlášením se do systému v podobě uživatelského účtu, který byl nejdříve zřízen. Stroj může být současně zpřístupněný se stejným účtem skrz více vstupních portů (tj. jediný uživatel může mít libovolně dlouho jeden účet či více různých účtů). Nicméně, když uživatelé přepnou mezi přístupovými porty, nebo se přepne mezi uživatelskými účty, pak přistupovat k vytvořenému souboru jiným účtem.

Každý uživatel přihlášen do Linuxu je připojen ke vstupnímu portu. Vlastnost paralelního přihlášení do systému je často využívaná přepínáním mezi vstupními porty. Dlouhá doba práce může být inicializována (zahájena) na jednom portu. Pak uživatel může přepnout na další port, kde může být zahájena další práce nebo lze navázat na předchozí práci, či si ji prohlédnout.

Struktura souborového systému

- Objekty souborového systému
 - Stromová struktura souborů jako hlavní pramen pro informace
- Standardní hierarchie souborového systému
 - Pravidla pro soubory a adresářová struktura
 - Jak najít konkrétní soubory
- Navigace v souborovém systému
 - Absolutní cesta - /etc/X11/jméno_souboru
 - Relativní cesta - /data/bin/jméno_souboru

Souborový systém, například `ext3`, je detailní sada konvencí uspořádání dat na disku. Konvence definují jak jsou data uložena a čtena z harddisku. Harddisk může mít jeden nebo více oddílů – partišen (particí). Každá partišna na Linuxu (volume-svazek) musí obsahovat souborový systém jenž se musí namontovat. „Souborový systém“ se zaměřuje s „typem souborového systému.“ V hierarchickém souborovém systému jsou adresářů uspořádané do stromové struktury.

Objekty souborového systému

Hlavní úložiště pro informaci na všech Unixu-like strojích je souborový strom, všechno se odvíjí od kořenového adresáře. Tento souborový strom není nic více než hierarchická sada adresářů, která postupně může obsahovat objekty souborového systému (FSO).

FSO	Příklad
Blok souborů	/dev/hda
Adresáře	/mnt
Symbolické linky	/dev/mouse --> /dev/psaux
Znakové soubory	/dev/psaux
Běžné soubory	/home/luke/file.txt

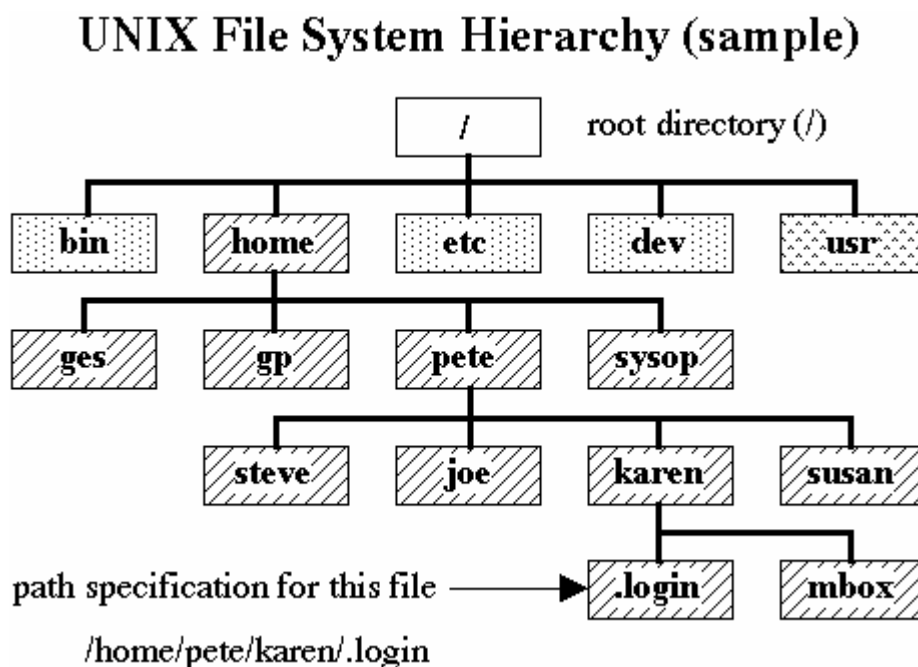
Tabulka 2 - 1 Objekty souborového systému

FSO jsou umístěny v souborovém systému, který může být namontován jako soubor do souborového stromu adresářů. Pět příkladů běžných FSOs v Linuxu jsou zobrazeny v Tabulce 2-1. Blok souborů odkazuje na řadiče disků nebo diskovou partíci. Pouze platný blok zařízení může být namontován do souborového systému. Příklad: `/dev/hda` je cesta pro primární IDE harddisk. Adresáře jsou další jednoduché kolekce FSOs. Adresář `/mnt` je běžný adresář jenž obvykle obsahuje další adresáře, podadresáře a soubory. Cesta `/dev/mouse` je určena pro myš. Znakový soubor (znakové zařízení) `/dev/psaux` odkazuje na zařízení mezi jádrem a konkrétním portem PS2 pro myš. Běžný soubor je jenom sbírka uložených informací upravených do souboru se jménem, například `/luke/file.txt`. Typy obyčejných souborů jsou tyto:

- datové soubory
- adresářové soubory
- spouštěcí soubory

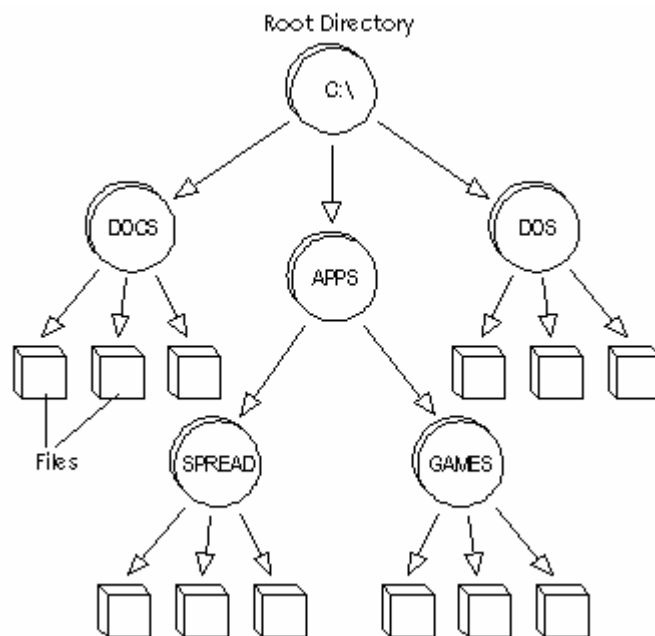
Standardní hierarchie souborového systému

Při pohledu na Obrázek 2-4a je vidět typická kořenová struktura Linuxu, a na Obrázku 2-4b Windowsu. Kořenový adresář je rozpoznán v průběhu zavádění systému. Ostatní svazky musí být namontovány do adresářů. Jména jsou montovací body, jejichž počátkem je kořenový adresář. Jsou tam i základní sady utilit které by měly být vždy přístupné, například `/bin`.



Obrázek 2-4a

WINDOWS FILE SYSTEM STRUCTURE



Obrázek 2 – 4b Kořenová souborová struktura

Standardní hierarchický souborový systém lze najít na www.pathname.com/fhs a je zobrazen na Obrázku 2-5. Linux Standard Base (LSB), jež lze najít na www.linuxbase.org, určený ke zvýšení kompatibility mezi distribucemi Linux. Kořenový adresář je prvotním (základním) adresářem souborového stromu, standardizovaných podadresářů jež jsou zobrazeny na souhrnném Obrázku 2-5.

- /** - začátek stromové struktury
- /bin** - základní spustitelné soubory - dalo by se přirovnat k adresáři DOS v MS-DOSu nebo adresáři Command ve Windows
- /boot** - zde je umístěno jádro (kernel) systému
- /dev** - soubory v tomto adresáři reprezentují jednotlivá zařízení - např. /dev/hda je primární master IDE disk
- /etc** - konfigurační soubory systému
- /home** - domovské adresáře uživatelů
- /lib** - základní knihovny systému a moduly pro kernel
- /lost+found** - ztracené a opravené soubory po chybách ext2
- /mnt** - sem se většinou připojují další zařízení, třeba /mnt/floppy reprezentuje většinou disketovou mechaniku
- /opt** - zde bývají aplikace, které nejsou standardní součástí distribuce
- /proc** - soubory jednotlivých procesů - dalo by se říci, že je to mapa paměti RAM
- /root** - domovský adresář superuživatele (root)
- /sbin** - další spustitelné soubory, které jsou součástí systému
- /tmp** - odkládací adresář
- /usr** - další stromová struktura, obsahuje velké množství informací, jako knihovny, zdrojové kódy, spustitelné soubory, konfigurační soubory a další.
- /usr/X11R6** - kompletní soubory X Windows
- /var** - soubory, jejichž obsah se během chodu systému většinou mění - třeba mail fronta

Obrázek 2 – 5 Hierarchický standard souborového systému

Navigace v souborovém systému

Linux pokračuje v tradicích Unixu, který používá hierarchická jména souborů a konkrétní soubor je specifikovaný cestou. Cestu tvoří začátek, doplněn adresáři, a jménem souboru. Existují dvě cesty a to absolutní a relativní.

Jméno s absolutní cestou

Jméno s absolutní cestou udává úplnou cestu:

```
/mnt/home/luke
```

První zpětné lomítko (/) znamená začátek nebo kořenový adresář. Následující zpětná lomítka (/) oddělují jména adresářů a jméno souboru.

Jméno s relativní cestou

Jméno s relativní cestou je relativní k aktuálnímu adresáři

```
mail/fired/letters
```

Například, po napsání příkazu `pwd`, je vrácena absolutní cesta:

```
/mnt/home/luke
```

Napsání `cd home` způsobí chybu protože musí být zadána absolutní, nebo úplná cesta. Napsáním `cd /home` se vrátíme do adresáře `home`. Ale, napsáním `cd mail` z adresáře `/mnt/home/luke` je uživatel přesměrován přímo do adresáře `mail` protože `/mnt/home/luke` je rodičovským (nadřazeným) adresářem adresáře `mail`. Zahrnout plné jméno cesty `/mail` není nezbytné, protože příkaz `cd mail` udává název cesty relativně k adresáři ve kterém se právě nacházíme.

Souborový systém a media jenž lze namontovat

- `mount`
 - Montování souborových systémů obsažených na zařízeních
 - `umount` – odmontování souborového systému
- `/etc/fstab`
 - Umožňuje uživatelům namontovat souborové systémy
 - Zjednodušené namontování souborových systémů
- Abstrakce souborového systému
 - `/proc` a `/dev/xxx`

Souborový systém a media jenž lze namontovat

Uživatelé často potřebují zpřístupnit data uložena v souborových systémech na vyměnitelných mediích (disketa, zip disk, CD-ROM, atd.). Abychom zpřístupnili tyto souborové systémy současně s dalšími systémovými soubory, zařízení jenž obsahuje souborový systém, musí být namontováno příkazem `mount`.

mount příkaz

Příkaz `mount` říká Linuxu kam umístit soubory v adresářovém stromu, a zobrazuje všechny namontované souborové systémy. Někdo by si mohl myslet, že příkaz `mount` namontuje zařízení: nicméně on namontuje souborové systémy nacházející se na zařízeních. Níže zobrazený příklad ukazuje jak namontujeme CD – ROM do adresáře `/mnt/cdrom/`:

```
$ mount -t iso9600 /dev/hdb /mnt/cdrom
```

Volba `-t` říká `mount`, že specifikujeme typ souborového systému. Typ souborového systému `iso9600` je souborový systém nacházející se na CD – ROMech. `/dev/hdb /mnt/cdrom` znamená, že sekundární blok IDE zařízení bude namontován do adresářového stromu ve virtuálním bodu namontování `/mnt/cdrom`. Pokud byl příkaz umístěn do souboru `/etc/fstab` pro namontování CD-ROM, může být použit následující příkaz:

```
$ mount /mnt/cdrom
```

Poznámka: K zobrazení souborového systému, který je k dispozici pro namontování, lze zjistit v souboru `/proc/filesystems`.

Lze použít následující příkaz:

```
$ cat /proc/filesystems
```

etc/stab

Za normálních okolností může pouze root použít příkaz `mount` k namontování souborového systému. Soubor `/etc/stab/` se využívá k tomu, aby poskytl uživateli možnost namontovat konkrétní systémy souborů, nastaví implicitně montovací body, jedná se o jednoduché použití příkazu `mount`. Zde je příklad obsahu souboru `/etc/stab/`:

pole 1	pole 2	pole 3	pole 4	pole 5	pole 6
/dev/hda3	/	ext2	defaults	1	1
/dev/hda1	/boot	ext2	defaults	1	2
/dev/cdrom	/mnt/cdrom	iso9660	noauto,user,ro	0	0
/dev/hda4	/usr	ext2	defaults	1	1
/dev/fd0	/mnt/floppy	auto	noauto,user	0	0
none	/proc	proc	defaults	0	0

Každé konkrétní pole reprezentuje následující:

- jméno souborového systému konkrétního zařízení
- bod namontování pro souborový systém
- typ souborového systému
- volby pro montování (implicitně je `rw`)
- výpis opakování
- souborový systém kontrola pořadí

Abstrakce souborového systému

Abstrakce souborového systému je souborový systém jako objekt, který je umístěn v abstraktním (vymyšleném) souborovém systému. Souborový systém `proc` je vymyšlený souborový systém který podává zprávu o stavu programů, ovladačích zařízení, a vnitřních

událostech jádra. Přístup k tomuto vymyšlenému systému souborů, nebo abstraktního systému, lze získat skrz adresář `/proc`. Jinými slovy, je zobrazen obsah souborů uvnitř adresáře `/proc`, nezmění svazky harddisku, ale přejde na rutinu jádra, jenž zobrazí stav systému jako sérii adresářů a souborů.

Abstrakce	Popis
<code>/proc</code>	Poskytuje rutiny pro řízení stavu jádra.
<code>/dev/xx0</code>	Poskytuje přímý přístup k zařízení <code>xx0</code> .
<code>/dev/null</code>	Vrací End of File když se čtou, zapisují a ignorují všechny vstupy.
<code>/dev/zeros</code>	Vrací vždy nulový znak.

Tabulka 2- 2 Tradiční abstrakce

Otázky k opakování

1. Jak byste popsali modulární koncepci LINUXU?
2. Vyjmenujte 6 hlavních modulů LINUXU.
3. Stručně popište jednotlivé moduly.
4. Co je to Multitasking?
5. Jaký je rozdíl mezi jedno-uživatelským a více-uživatelským systémem?
6. Co je souborový systém?
7. Pokuste se popsat hierarchii souborového systému Linuxu a Windowsu.
8. Vyjmenujte a popište několik hierarchických standardů souborového systému.
9. Jaký je rozdíl mezi absolutní a relativní cestou?
10. K čemu slouží příkaz `mount`?

Lab

1. Vyhledejte pomocí Internetu nebo v dokumentaci nejčastěji používané souborové systémy pro Linux.
2. Vyzkoušejte si příkazy uvedené v této lekci.
3. Vyzkoušejte si namountovat například svůj flash-disk.