

## 34 FS pro OS Linux

### Obsah hodiny



Obsahem této hodiny je popis Linuxových souborových systémů.

### Cíl hodiny



Po této hodině budete schopni:

- orientovat se v Linuxových FS
- popsat souborový systém ext2
- charakterizovat virtuální FS

### Klíčová slova



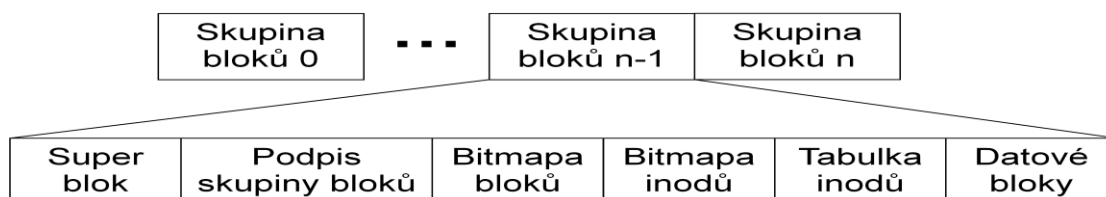
FS ext, RaiserFS, ReiserFS, JFS, XFS, ZFS, Inode, Skupina bloků, Superblok, Virtuální FS.

### 34.1 Ext2

FS ext2, Second Extended Filesystem. Mezi jeho přednosti patří spolehlivost a slušná rychlost všech běžných operací.

Data jsou uložena ve stejně dlouhých blocích. Při vytváření FS pomocí příkazu *mkfs*, nebo pomocí instalátoru distribuce, si lze zvolit velikost bloku. Použitelné velikosti se pohybují od 1024 do 4096 bajtů, menší bloky trochu šetří místo na disku, větší umožňují rychlejší práci se systémem

Disk je z pohledu systému sekvenční řada bloků. Chce-li systém načíst data, požádá ovladač o určený počet bloků jdoucích po sobě.



Obrázek 34-1: Skupina bloků.

Bloky v Linuxu jsou rozděleny na skupiny. V každé skupině jsou bloky, které popisují FS a skupinu bloků.

Skupina bloků obsahuje následující bloky

- superblok (základní popis FS: např. počet volných bloků, velikost bloků, počet bloků ve skupině, číslo prvního inode, tj.inode kořenového adresáře...),
- deskriptor skupiny bloků (popisuje skupinu bloků: např. počet volných bloků, inodů ve skupině, čísla bloků, ve kterých jsou uloženy bitmapy, tabulky inodů atd.),
- bitmapu bloků,
- bitmapu inodů,
- tabulku inodů,
- datové bloky

První a nejdůležitější blok v každé skupině se nazývá superblok. Superblok funguje podobně jako cedule u vchodu do knihovny, která nám sdělí, kde jsme a kde máme hledat katalogy a poskytne nám nejrůznější důležité informace o celém souborovém systému.

Základním prvkem FS ext2 je inode. I Inode je datové struktura, ve které jsou uloženy veškeré informace o souborech<sup>1</sup>. Inode si můžete představit jako kartičku z lístkového katalogu obsahující údaje umožňující vyhledat knihu v knihovně. Systém identifikuje soubory právě podle čísla inode.

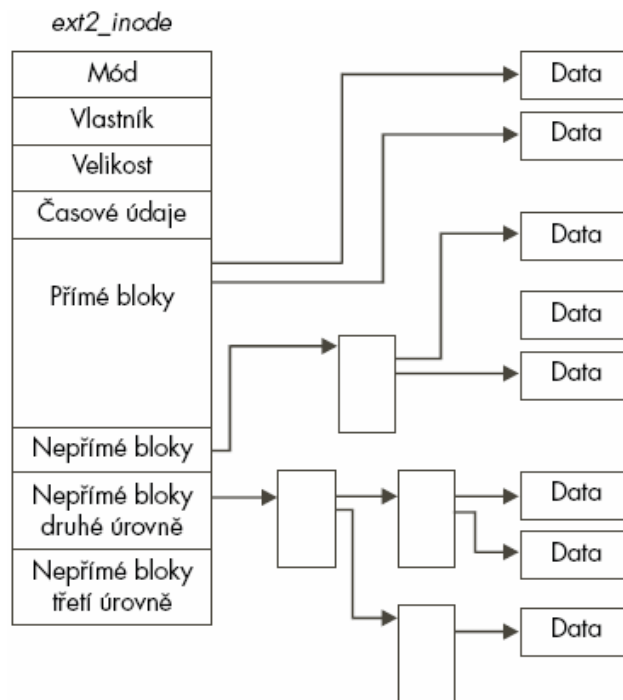
Každý inode uchovává odkaz na obsah souboru a informace o souboru (metadata):

- mód,
- uživatelský a skupinový identifikátor vlastníka souboru,
- velikost,
- časové značky
- ukazatele na datové bloky – přímé a nepřímé bloky

Přímé bloky obsahují ukazatele na fyzické bloky, ve kterých jsou uložena data. Je-li soubor delší než 12 bloků, zavádí se další tři ukazatele na nepřímé bloky. Nepřímé bloky obsahují ukazatele na bloky, ve kterých jsou ukazatele na data. Jedné se o nepřímé adresování dat. To znamená, že k velkým souborům se přistupuje pomaleji.

---

<sup>1</sup> Počet inodů na filesystému si mkfs zvolí podle kapacity oddílu, ve výchozím nastavení je to jeden inode na každých 8 kB kapacity, u malých oddílů na 4 kB.



Obrázek 34-2: Inode

Adresáře jsou z pohledu ext2 speciální soubory, které slouží k vytváření a ukládání přístupových cest k souborům. Kromě toho pouze zde jsou uložena jména souborů, které jsou jinak reprezentována jako čísla inodů.

Adresář	
i1	.
i2	..
i3	jméno souboru
i4	jméno souboru
i4	jméno souboru

Obrázek 34-3: Adresářový soubor

První dvě položky v adresáři jsou vždy „.“ (aktuální adresář) a „..“ (rodičovský adresář). Jména adresářů se oddělují lomítky /. Při vyhledávání určeného souboru systém postupně odděluje jména adresářů, nachází je v seznamu a překládá na číslo inodu. To dělá rekurzivně tak dlouho, dokud nedojde k položce, která není adresářem – a to je hledaný soubor. Hledání souboru začíná vždy od prvního inodu.

## 34.2 FS ext3, ext4

FS ext3, ext4 jsou založeny na FS ext2, ke kterému je připojen žurnál, velikost žurnálu je volitelná.

Přechod z ext2 a na ext3 je jednoduchý, pomocí příkazu *tune2fs*. Lze nastavit i úroveň žurnálování.

```
tune2fs -j /dev/hda1
```

```
tune2fs ^has_journal /dev/hda1
```

Nastavení úrovně žurnálování

```
mount /dev/hda1/mnt/space -t ext3 -o data=journal
```

- *data=journal* → žurnálu se zapisují operace s vlastními daty i metadaty, obnovu lze provést v plném rozsahu
- *data=ordered* → zapisují se pouze transakce s metadaty (po zápise dat – implicitní nastavení)
- *data=writeback* → žurnálují se pouze metada – (odpovídá nejvíce ext2)

## 34.3 ReiserFS a Reiser4

Je to jeden z nejstarších Linuxových žurnálovacích FS. Umí efektivně pracovat s velkým množstvím malých souborů a to proto, že do jednoho bloku umí uložit více souborů nebo konce souborů (tails). Tím se docílí menší vnitřní fragmentace, což znamená lepší využití místa na disku.

Nevýhodou je naopak vyšší externí fragmentace a snížení výkonu u větších souborů. Problematické je i přidávání dat na konec souboru - data se musí přesouvat do jiných bloků. Proto lze uvedenou funkci v případě potřeby vypnout.

Lze měnit velikost FS (bez odpojení fs) – příkazem *resize-reiserfs*

## 34.4 Další FS

### JFS: IBM Journaling Filesystem

Vznikl začátkem 90. let v laboratořích firmy IBM. Původně byl navržen pro operační systém AIX a časem přenesen i pod OS/2 a následně pod Linux. Byl to pravděpodobně první filesystém, ve kterém se objevilo žurnálování

JFS velice efektivně zvládne všechny druhy zátěže, ale nepodporuje média menší než 16 MB.

## **Systém XFS**

Byl vyvinut firmou Silicon Graphics pro operační systém IRIX a posléze zveřejněn a přenesen i pod Linux, je podobný JFS.

Hlavním cílem autorů tohoto filesystému byl velký výkon při práci s objemnými daty: při jednom z testů se povedlo na 32procesorovém serveru dosáhnout přenosové rychlosti 7 GB/s.

Zdrojový kód je značně složitý. Zdrojové texty XFS v jádru Linuxu čítají více než sto tisíc řádků řádků (JFS má okolo 30000, ext3 cca 15000).

## **ZFS (Zettabyte File System)**

FS vyvinutý společností Sun Microsystems pro operační systém Solaris. Díky jeho uvolnění pod open-source licencí CDDL může být jeho kód portován i do dalších operačních systémů.

Jedná o 128 bitový souborový systém, což v praxi znamená, že dokáže teoreticky spravovat až 16 miliard miliard ( $16 \cdot 10^{18}$ ) kapacit dnešních běžných 64bitových souborových systémů.

ZFS používá k ukládání dat bloky s proměnlivou velikostí. Díky svému zcela odlišnému návrhu zachází ZFS s disky jinak než běžný souborový systém. Využívá společnou kapacitu všech dostupných jednotek, nad kterými pak vytváří vlastní strom souborových systémů. Celková kapacita se nazývá pool (jako bazén). Už tedy nejsou potřeba žádné diskové oddíly (partitions).

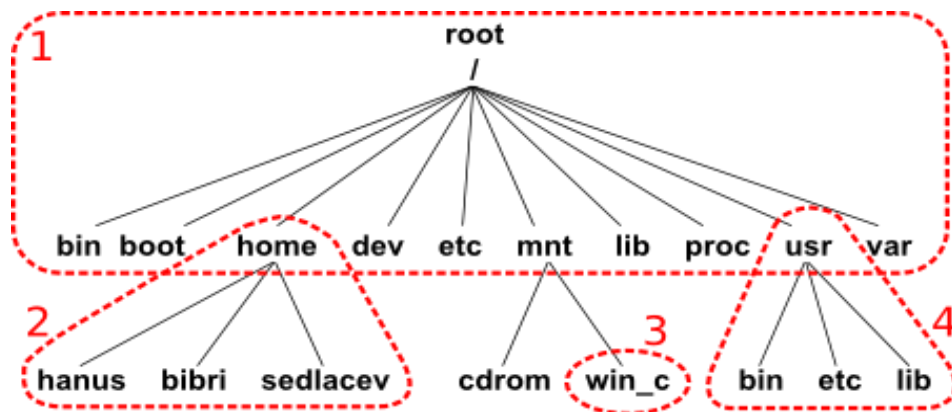
Aby ZFS optimalizoval svou práci, používá dynamické rozdělování zátěže na všechny jednotky. Jakmile je přidán další disk, dojde automaticky k přerozdělení dat mezi disky, takže je pak ihned používán naplno i nový disk.

## **34.5 Virtuální FS**

V OS Linux existuje pouze jeden kořenový adresář, označený /. K němu, tj. do /, se prostřednictvím přípojných bodů (adresářů) připojují všechny další souborové systémy a vytvářejí tak jeden (virtuální) strom. Jedná se o souborové systémy na připojitelných zařízeních a také o souborové systémy umístěné na různých oblastech disku.

Při instalaci se totiž běžně disk v Linuxu rozděluje na několik oblastí. Z kořenového FS jsou pak vyjmuty jeho části a umístěny na tyto samostatné diskové oblasti. Při startu systému se pak automaticky připojují do příslušného adresáře ve stromu kořenového adresáře. Uživatelé se pak navenek jeví jako jediný strom.

Připojením všech oddílů a zařízení do jedné stromové struktury se vytváří virtuální souborový systém, funguje na stejném principu jako ext2.



Obrázek 34-4: Virtuální FS v Linuxu

Na Obrázek 34-4: oblasti 1-4 představují diskové oddíly (partitions).

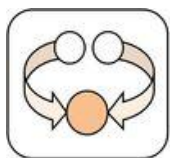
- Oblast 1: kořen se svými podadresáři.
- Oblast 2: do adresáře `/home` byla připojena disková oblast s adresářovým podstromem `home` (do něj se ukládají data uživatelů).
- Oblast 3: do adresáře `mnt/win_c` je připojena disková oblast, na které je FS Windows.
- Oblast 4 do adresáře `/usr` byla připojena disková oblast s adresářovým podstromem `usr`.

Dalšími podstromy, které se obvykle vyčleňují a umísťují na samostatnou oblast disku, bývají `/var`, `/boot`.

Důvody pro umísťování adresářových podstromů na samostatné diskové oblasti jsou různé, např.:

- omezení velikosti adresářů (adresářových stromů), které mají tendenci se zvětšovat,
- jednodušší zálohování: některé části FS se musí zálohovat častěji, některé pouze „jednorázově“ – např. při změně konfigurace, zálohuje se jednoduše celá oblast ...
- při upgradování či přeinstalování systému, zůstanou některé oblasti nedotčeny (např. data uživatelů).

## Shrnutí kapitoly



OS Linux používá řadu FS: FS ext4, RaiserFS, ReiserFS, JFS, XFS, ZFS a další. Mimo to podporuje Linux i FS dalších OS včetně Windows. Nejčastěji se používá FS ext3, ext4, který vychází z ext2, ke kterému byl připojen žurnálovací soubor. Základem jeho struktury je inode. Data se zapisují do bloků, bloky jsou rozděleny na skupiny bloků.

Skupinu bloků tvoří:

- Superblok,
- deskriptor skupiny bloků,
- bitmapu bloků,
- bitmapu inodů,
- tabulku inodů,
- datové bloky.

V OS Linux existuje pouze jeden kořenový adresář, označený /. K němu, se připojují všechny další souborové systémy a vytvářejí tak virtuální FS:

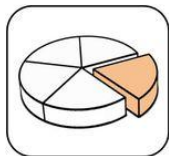
- FS na připojitelných zařízeních,
- FS umístěné na různých oblastech disku

## Kontrolní otázky a úkoly



- 1) Jaké souborové FS Linux používá a podporuje?
- 2) Co je to inode, co je jeho obsahem?
- 3) Co tvoří skupinu bloků?
- 4) Co obsahuje superblok?
- 5) Co obsahuje deskriptor skupiny bloků?
- 6) Co je to virtuální FS?
- 7) Co je důvodem pro rozdělení kořenového adresáře na podstromy a jejich umístění na samostatné diskové oblasti?

## Použitá literatura a jiné zdroje:



- [1] BERNÁTHOVÁ, Anička. Linuxové souborové systémy. LinuxExpres [online]. 15. prosinec 2006 [cit. 2012-03-16]. Dostupné z: <http://www.linuxexpres.cz/praxe/linuxove-souborove-systemy>
- [2] VESELSKÝ, J. LINUX: dokumentační projekt. Vyd. 2. Překlad Marek Kocan, Ludvík Roubíček, Jiří Veselský. Brno: Computer Press, 2003, 1001 s. Operační systémy (Computer press). ISBN 80-722-6761-2.