

Zálohování v operačním systému Linux

Backup in OS Linux

Lukáš Mrtvý

Bakalářská práce
2011



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Lukáš MRTVÝ
Osobní číslo: A08243
Studijní program: B 3902 Inženýrská informatika
Studijní obor: Informační a řídicí technologie

Téma práce: Zálohování v operačním systému Linux

Zásady pro vypracování:

1. Vytvořte literární rešerši na téma zálohování v OS Linux.
2. Popište obecně problematiku zálohování a zpětné obnovy po ztrátě dat.
3. Podrobněji popište alespoň jeden z projektů, který se věnuje zálohování dat v OS Linux.
4. Vytvořte sadu vlastních skriptů pro zálohování.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. VYCHODIL, Vilém. Linux: Příručka českého uživatele. Nám. 28. dubna 48, 635 00 Brno : Computer Press, 2003. 270 s.
2. WELSH , Matt. Používáme Linux . Nám. 28. dubna 48, 635 00 Brno : Computer Press, 2003. 660 s.
3. BOLDRITO, Remo ; ESTEVE, Josep . GNU/Linux Advanced Administration [online]. [s.l.] : Free Technology Academy, 2004 [cit. 2011-02-04]. Dostupné z WWW: [http://ii.iinfo.cz/r/k/gnu-linux-advanced-administration.pdf]
4. Linux - Dokumentační projekt [online]. Nám. 28. dubna 48, 635 00 Brno : Computer Press, 2003 [cit. 2011-02-04]. Dostupné z WWW: [http://ii.iinfo.cz/r/k/Dokumentacni_projekt.pdf]
5. NEWHAM, Cameron . Linux Shell Scripting with Bash. [s.l.] : Sams, 2004. 432 s.

Vedoucí bakalářské práce:

doc. Ing. Martin Syseľ, Ph.D.

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

25. února 2011

Termín odevzdání bakalářské práce:

7. června 2011

Ve Zlíně dne 25. února 2011


prof. Ing. Vladimír Vašek, CSc.
děkan




prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Práce se zabývá metodikou zálohy a zpětné obnovy dat v operačním systému Linux. Dále také obsahuje problematiku zálohování obecně. Je položeno několik otázek, po jejichž odpovědi by měl každý správně určit takový zálohovací plán, který by byl nejvhodnější a měl by vést k realizaci celého zálohovacího systému. Část bakalářské práce se věnuje popisu jedno z mnoha komplexních, již vytvořených zálohovacích systémů. Jedním z posledních bodů je také tvorba vlastního zálohovacího scriptu v operačním systému Linux, konkrétně v distribuci Ubuntu.

Klíčová slova: linux,tar,dump,ubuntu,pracovní stanice,bacula,raid

ABSTRACT

This work deals with the methodology of backing up and following data recovering in operational system Linux. Further it contains the problematic of backing up generally. Several important questions are asked and after their answering anyone should be able to define backing up plan which would be most suitable, and which should lead to the whole back up system implementation. One part of the thesis describes one of the complex, already created backing up systems. Creation of my own backing up script in operational system Linux, in distribution of Ubuntu in particular, is one of the final points of this thesis.

Keywords:linux,tar,dump,ubuntu,workstation,server,bacula,raid

Poděkování

Děkuji vedoucímu své bakalářské práce doc. Ing. Martinu Syslovi, Ph.D. za cenné rady, připomínky a čas, který mi věnoval.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	11
1 PROČ - DŮVOD ZÁLOHOVÁNÍ	12
1.1 MOŽNÉ CHYBY	13
1.1.1 Chyba uživatele	13
1.1.2 Chyba správce	13
1.1.3 Chyba hardwaru	13
1.1.4 Chyba disku.....	13
1.1.5 Systémová chyba.....	14
1.1.6 Softwarová chyba.....	14
1.1.7 Ostatní	14
2 ROZHODNUTÍ O TOM, CO ZÁLOHOVAT	15
2.1 OTÁZKY	15
2.2 PLÁNOVÁNÍ.....	16
2.2.1 Záloha zálohy	16
2.2.2 Jaké periferní zařízení vlastníte.....	16
2.2.3 Způsob rozdělení oddílů.....	16
2.2.4 Způsob konfigurace LVM.....	17
2.2.5 Nastavení databáze.....	17
2.2.6 Dokumentace DHCP,NFS,CIFS	17
2.2.7 Vědět, co opravdu servery obsahují	17
2.3 PLÁN	18
2.4 VĚDĚT, CO OPRAVDU ZÁLOHUJEME	18
2.5 ZÁLOHOVÁNÍ VYBRANÝCH JEDNOTEK ČI FILESYSTEMŮ	18
2.6 ZÁLOHOVÁNÍ CELÉHO SYSTÉMU	20
3 KDY ZÁLOHOVAT	21
3.1 STUPNĚ.....	21
3.2 JAKÝ STUPEŇ ZVOLIT A KDY	22
3.3 DALŠÍ.....	25
3.3.1 Integrita	25
3.3.2 Rychlost.....	25
4 JAK ZÁLOHOVAT	26
4.1 ZÁLOHOVACÍ PROGRAMY	26
4.1.1 Bacula.....	26
4.1.2 Složení.....	26
4.1.3 Popis.....	30
4.2 ZÁLOHOVÁNÍ V LINUXU	31
4.2.1 Jak to funguje?	31
4.2.2 4 možnosti zálohy.....	34

4.2.3	Alt-Boot Full Image Metoda	37
4.2.4	Alt-Boot Partition Image metoda	38
4.2.5	Live metoda.....	40
4.2.6	Alt-Boot Filesystem metoda.....	42
5	ÚLOŽIŠTĚ ZÁLOH	43
5.1	ON-LINE ÚLOŽIŠTĚ.....	43
5.2	OFF-SITE.....	43
5.3	NEAR-LINE	43
5.4	OFF-LINE	43
6	OBECNÉ	44
6.1	TESTOVÁNÍ ZÁLOH	44
6.2	MONITOROVÁNÍ ZÁLOH	45
6.3	MÉDIA	45
6.4	TECHNOLOGIE RAID.....	46
II	PRAKTICKÁ ČÁST	48
7	ZÁLOHOVACÍ SCRIPT.....	49
7.1	CÍL PRÁCE.....	49
7.2	ŘEŠENÍ	49
7.2.1	Struktura aplikace.....	49
7.2.2	Popis programu	50
7.2.3	Ukázky	51
	ZÁVĚR	53
	ZÁVĚR V ANGLIČTINĚ.....	54
	SEZNAM POUŽITÉ LITERATURY.....	55
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	57
	SEZNAM OBRÁZKŮ	58
	SEZNAM TABULEK.....	59
	SEZNAM PŘÍLOH.....	60

ÚVOD

Dobrý zálohovací a obnovovací systém je základem pro každou firmu jakékoli velikosti. Ne vždy je firma schopna investovat do zálohovacího a obnovovacího systému dostatek finančních prostředků. Relativně přes celkem malý rozpočet je možno vytvořit velmi dobrý zálohovací systém, vyhovujícím požadovaným potřebám. V případě, že investice do těchto systémů není vyhovující, hrozí ztráta důležitých dat, času, peněz, ale dokonce i pracovního místa. Ne ve všech případech je potřeba obnovení dat "na minutu" přesně, někdy postačí i záloha ze "včerejšího" večera. V některých případech je nástroj, nebo program, který je potřeba, dodáván spolu s operačním systémem, ale pro vaše požadavky ne vždy stoprocentně využitelným.

Pokud je vaší starostí v podniku jen několik zařízení, v nichž je použit stejný operační systém, je tu řešení. Toto řešení je platné i v případě obrovského podniku se stovkami zařízení různých operačních systémů, například Unix, Linux, Windows, Mac OS. Největším problémem pro mnoho lidí je dezinformace, zkrátka, většina lidí žije v nevědomosti, co je v dnešní době k dispozici, nebo se spokojí s nedokonalým zálohovacím systémem a jejich potřeby není zdaleka schopen naplnit. Volba zálohovacího systému je závislá na šesti nejdůležitějších otázkách, které je potřeba položit si, aby byl váš systém schopen pracovat co nejúčinněji.

Tyto otázky jsou:

- Proč?
- Co?
- Kdy?
- Kde?
- Kdo?
- Jak?

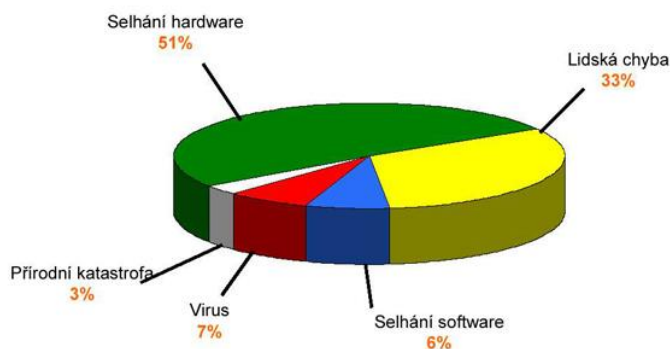
- **Proč?**
 - Důvod ochrany proti selhání
 - Záleží opravdu na tom, jestli ztratíte data
 - Jaká pro vás bude ztráta
 - Jaký typ dat vlastníte a jaká je hodnota každého typu dat
- **Co?**
 - Co se chystáte nechat zálohovat, celý systém, jednotky, nebo pouze složky, popřípadě soubory
 - Na jakém operačním systému se chystáte zálohovat
 - Co je dobré kromě jednotek a systému dále zálohovat
- **Kdy?**
 - Kdy je ten nejlepší čas zálohovat systém
 - Jak často bude probíhat plná záloha
 - Kdy provést inkrementální zálohu
- **Kde?**
 - Kde se bude záloha nacházet
 - Kde je nejlepší místo pro uložení vaší zálohy
- **Kdo?**
 - Kdo je poskytovatelem hardwaru, softwaru
 - Kdo to všechno dá dohromady
- **Jak?**
 - Jak se toho chystáte dosáhnout
 - Existuje několik různých způsobů, jak se chránit před ztrátou
 - Jaké jsou různé metody, např.: skladování, replikace, zrcadlení, RAID

I. TEORETICKÁ ČÁST

1 PROČ - DŮVOD ZÁLOHOVÁNÍ

Pokud si nejste schopni odpovědět na tuto otázku, nemá smysl se dále zabývat problematikou zálohování. Důležitým krokem je zamyšlení nad tím, co všechno se může stát vašim datům a následný pohled na všechny typy dat, které vlastníte.

Člověk by měl být obeznámen s každou obchodní a firemní jednotkou, která nějakým způsobem produkuje a vytváří data, zejména jakým způsobem by byla ovlivněna tato obchodní jednotka v případě, kdyby všechna její data byla ztracena či poškozena. Teprve tehdy je vaše firma schopna posunout se dále. Rizika, která vás v této problematice mohou potkat jsou značná, například porucha hardwaru, působení virů, požár, teroristický útok nebo s lidská chyba. [1]



Obrázek 1 Příčiny ztráty dat podle zdroje [1b]

Na obrázku (Obr. 1) jsou zobrazeny hlavní příčiny ztráty dat dle statistik v procentech. V případě, kdy je postaven systém odolný většině faktorů, zůstává zde existence toho nejdůležitějšího faktoru, a to lidského.

Je třeba stanovit si cenu potenciálně ztracených dat a uvědomit si kolik finančních prostředků by bylo třeba na opětovné pořízení stejných dat. Tato cena pak může být rovna i ceně celé organizace přesto, že tato organizace nebude bez zmíněných dat schopna fungovat.

Mezi další otázky řadíme úvahu finanční náročnosti odstávky potenciálně ztracených dat, v řádu minut, hodin a dnů. Tyto odpovědi pak stanoví přijatelné náklady na požadovaný zálohovací systém v závislosti na čase obnovy. [2]

1.1 Možné chyby

V případě, že se rozhodnete provést zálohu, je také dobré uvědomit si, jakým způsobem provést zálohu. Ujistěte se, že jste si vědomi všech možných příčin ztráty dat, a pak se ujistěte, že máte připraveny záložní plány pro všechny z nich. Zde je několik problému, před kterými byste se měli chránit.

1.1.1 Chyba uživatele

Procentuelně jedna z největších příčin obnovy v každém prostředí. Jedná se o to, že uživatel omylem smaže nějaký soubor, ale vám to přijde říci až za hodinu. Tohle můžete ošetřit nepřetržitou ochranou dat a systémů, ne však v případě zálohy, která že je naplánována jednou za noc.

1.1.2 Chyba správce

Méně častá chyba (ovšem pokud běžní uživatelé nemají root oprávnění). V případě, kdy by uživatel spustil příkaz newfs, smazal by půlku filesystému a to by ovšem byla vaše chyba. Chránění se před touto chybou je zde jednak typická noční záloha, nebo snímkování systému (snapshots).

1.1.3 Chyba hardwaru

Většina knih mluví o chránění se před selháním hardwaru, ale obvykle nezmiňují, že hardwarová chyba může přijít ve dvou formách selhání, jednak chyba disku a systémová porucha. Je důležité zmínit, že na chránění se před těmito chybami platí dvě odlišné metody. V případě chyby disku může pomoci zrcadlení disku, ale v případě systémové chyby to zde neplatí. Chránění se před tímto typem selhání je pak v snímkování systému.

1.1.4 Chyba disku

Chránění se před diskovou chybou je v dnešní době poměrně jednoduché, jediné, v čem se musíte rozhodnout je, jak bezpečně to chcete udělat. Zrcadlení, často označován jako RAID 1 nabízí nejlepší ochranu, ale za cenu znásobení počátečních investic. To je důvod, proč si většina uživatelů vybírá některý z dalších úrovní RAID (Redundantní pole nezávislých disků). Nejpopulárnější je pak RAID 5 či RAID 6. RAID 5 na základě výpočtu a ukládání paritních informací chrání váš disk před ztrátou dat. RAID 6 pak tuto paritu

ukládá dvakrát, což umožní selhání více než jednoho disku. Bližší popis lze naléznout v kapitole č. 8

1.1.5 Systémová chyba

Nejlepší ochranou proti této chybě je jednak dokumentování vašeho systému, používání logování a ostatní monitorovací metody ke sledování vašeho systému. Je třeba dávat pozor na zprávy tykající se špatných disků, řadičů, CPU a pamětí. Díky těmto varováním šanci předejít systémové chybě. Další možností jak se chránit je používat žurnálovací filesystém. Žurnálovací filesystém se chová podobně, jako databáze, sleduje zápisy a částečné zápisy a v případě selhání může takto částečně zapsaná data obnovit do původního stavu.

1.1.6 Softwarová chyba

Chránit se před softwarovou chybou může být těžké, jednak musíte brát v potaz chyby (bugy) operačního systému, databáze, atd. Způsoby jakým se chránit jsou pak časté snapshoty¹(snímkování) či neustálá ochrana systému.

1.1.7 Ostatní

Jako další bych uvedl případy úmyslného vandalismu, elektronické krádeže, vloupání, špatná archivace, přírodní katastrofy.

¹ Snapshot záloha je stav(snímek) systému v určitém časovém okamžiku

2 ROZHODNUTÍ O TOM, CO ZÁLOHOVAT

Je prokázáno, že jednou z nejčastějších příčin ztráty dat je to, že ztracená data nebyla nikdy nakonfigurována tak, aby mohla být po případné ztrátě obnovena do původního stavu.

2.1 Otázky

Vaší snahou by mělo být rozhodnutí jaké soubory zahrnout do zálohy.

Představa nejhoršího možného scénáře, který se může odehrát v případě ztráty dat je hrozivá.

Zde je několik možných variant:

- Celý centrální systém zachvátí požár a následně se zřítí celý komplex k zemi.
- Vzhledem k tomu, že tento stroj byl tak důležitý, museli jste ho replikovat na jiný stroj, který se nachází hned vedle. Samozřejmě, že ze stroje, který začal hořet první, se požár rozšířil i na stroj druhý.
- Představte si centrální server, který řídí veškeré zálohy a uchovává záznamy o tom, kde se nachází jednotlivé zálohy. Druhý server, který náhodou shoří, vydává tak intenzivní teplo, že s sebou "vezme" i tento centrální server, který stojí hned vedle něj.
- Katastrofa pokračuje tím, že shoří všechny *DHCP* servery, *NIS* master servery, *NFS* a *CIFS* servery, databázový server, který je domovem a zároveň inventářem všech telefonních položek na vaše obchodní partnery a klienty, nachází se zde všechny vaše servisní smlouvy, postupy.
- Ještě jste si nezapamatovali čísla na vaše nové off-site obchodníky, tak jste si je vytiskli a připíchnuli na zeď, hned vedle serveru, který zrovna v tuto chvíli zachvátí požár, plameny se jistojistě rozšířily a sežehly celý papír.

Důležité je uvědomit si, co by bylo prvním vašim činem, kdyby se opravdu jeden z těchto scénářů stal skutečností a vzpomenout si, na to, kdy proběhlo poslední zálohování. Zejména kolik ve skutečnosti bylo disků na serveru, který shořel, co to bylo za modely a jakým způsobem byly oddíly formátovány. Zda nebyl některý z nich v zrcadlení s jiným.

Toto je nutno znát dříve, než může vůbec dojít ke katastrofě.

2.2 Plánování

Ujištění, že bude k dispozici scénář, který by bylo potřeba realizovat v případě katastrofy.

2.2.1 Záloha zálohy

Mnoho společností začalo centralizovat kontrolu svých záloh. V případě uchýlení k takovému kroku, je pouze jedna šance v případě selhání.

Obnova zálohovacího serveru není možná v případě, když neexistuje databáze záloh, to platí i naopak. V první řadě je obnovení takto důležitého serveru prvním krokem v případě multisystémového výpadku. Měl by zde být co nejjednodušeji zdokumentován návod pro případnou obnovu tak rozsáhlého serveru.

Dobrou radou je zde ovšem ukládat zálohy záloh i na jiný server, než je server, na něhož ukládáme originální zálohy. Konec konců, i to je úkolem duplicitních záloh, v případě, kdy selže záloha originální. [1]

2.2.2 Jaké periferní zařízení vlastněte

Je předpokladem, že zálohujete na regulérní disky regulérní konfigurace, důležitý je seznam všech použitých disků. Problémem je neznalost typů modelů. Pokud se jedná o modely stejné značky například WD RE4 500 GB, potom není problém, ale mnoho serverů má rozdílné typy disků, ať je to WD AV 40 GB, Seagate 120 GB nebo Hitachi 320 GB. Toto je důležité mít na paměti, Unixové a Mac OS systémy uchovávají takové to záznamy ve speciálním souboru. Windows takovou informaci ukládá například do registrů. Dobré je zahrnutí těchto souborů do zálohy. [1]

2.2.3 Způsob rozdělení oddílů

V případě obnovy operačního nebo databázového systému je potřeba mít na paměti fakt, že oba systémy mají typicky jiné rozdělení oddílů a v případě obnovy, tyto oddíly musí být naprosto totožné, aby obnova proběhla korektně. Tato informace není běžně k dispozici, tudíž je dobré udělat něco navíc a tyto informace si dohledat. V současné době existuje několik scriptů, které jsou zdarma dostupné na internetu, několika příkazy vám tyto informace vypíší do souboru. [1]

2.2.4 Způsob konfigurace LVM

Nejprve, co je to LVM neboli Řízení logických disků. Je to v podstatě systém, který umožňuje přidávat jednotlivá zařízení nebo diskové oddíly do větších logických skupin svazků. Tyto skupiny lze přidáním, nebo odebráním fyzických zařízení podle potřeby zvětšovat či zmenšovat a za chodu přesouvat uložená data mezi jednotlivými fyzickými zařízeními aniž by bylo nutné běžící systém zastavit. [2b]

Mnoho takových "manažerů" je dostupných pro různé operační systémy, například *Veritas Volume Manager*, *Windows Dynamic Drives*, *HP's Logical Volume Manager*.

Informace jako je například vaše konfigurace, není vždy záznamem zálohovacích programů. Existují příkazy, jako například `lvmcfsbackup`, který přehledně zaznamená konfiguraci LVM. V některých případech, bývá tato konfigurace řádně zdokumentovaná, není nutností jí potom přidávat do svého listu záloh.

2.2.5 Nastavení databáze

Další důležitá informace, která by měla být definitivní částí plánu.

2.2.6 Dokumentace DHCP,NFS,CIFS

Téměř vždy je dána koexistence systémů NIS a NFS. Systém NFS umožňuje sdílet data po síti. Počítač poskytující data se nazývá souborový server file server. Počítače využívající tento diskový prostor jsou nazýváni klienty. [3]

Dobrá dokumentace je další definitivní částí plánu. Její aktualizace a dostupnost by měla být průběžně zajištěna. Značným ulehčením je vlastní script, který nastaví většinu za vás. Při instalaci nového serveru pak tento skript můžete vždy spustit. Pro Linux existuje *Linux Kickstart*, který dokáže zautomatizovat tento proces. [1]

2.2.7 Vědět, co opravdu servery obsahují

Nejlepším pojištěním proti téměř všem druhům ztráty dat je, aby byla určena zodpovědná osoba, která by za daná data zodpovídala a byla seznámena se všemi systémy. Pokud by spadl server, tato osoba by měla okamžitě vědět, jestli tento server obsahuje Oracle nebo SQL databázi.

2.3 Plán

Dalším důležitým krokem je plán, který vás povede krok za krokem v případě obnovy z katastrofy. Někdy samozřejmě nebude postup přesně takový, jaký jste naplánovali. Drobné úpravy budou určitě potřeba, ale pro začátek bude tento plán sloužit jako návod jak se zachovat v takovém případě. To bude lepší varianta, než stát na místě a přemýšlet na tím, čím začít nejdříve.

2.4 Vědět, co opravdu zálohujeme

Existují různé způsoby, jak se postavit směrem k výběru toho, co chceme zálohovat. Obecně se dá mluvit o dvou možnostech jakým způsobem zálohovat.

Jednak říci zálohovacímu systému, aby zálohoval všechno, kromě tzv.exclude listu. Lépe řečeno, zálohuje všechno, kromě mnou vybraných složek či adresářů.

Například:

```
1 Include: *
2 Exclude: /tmp, /junk
```

V druhém případě systém zálohuje pouze to, co je uvedeno v tzv.include listu.

Například:

```
3 Include: /data /data2 /data3
```

Zálohovat všechno, mimo mnou vybraných souborů je mnohem snadnější a bezpečnější, než zálohovat pouze výběr souborů. Je zde hrozbou snadný únik důležitých věcí a dat. Mnoho lidí tvrdí, že preferuje zálohování všeho, ale jen několik procent z nich skutečně tuto variantu zvolí.

2.5 Zálohování vybraných jednotek či filesystémů

Zde je několik argumentů pro a proti, v případě, pokud jde o zálohování pouze vybraných dat.

Ušetření místa a síťového provozu

První argument, který se hned nabízí jako plus k této metodě je, že ušetříte spoustu volného místa. Doporučením je mít dvě skupiny, jednak skupinu, ve které bude operační systém a skupinu pro ostatní data. Myšlenka spočívá v tom, že zálohování operačního systému nebude tak časté, jako zálohování pro ostatní data. Někdy se doporučuje provádět zálohu operačního systému jen v případě významných změn, jako například aktualizace operačního systému, rebuildování kernelu, atd. Druhá skupina by pak představovala každodenní zálohování dat, která se častěji modifikují.

Prvním problémem, který se v případě této naskytne je, že když se podíváme na dnešní moderní systémy z hlediska datového poměru, datový tok je jednoznačně větší. V případě, kdy byste do své zálohy nezahrnuli OS, místa ani času o mnoho neušetříte. Když uvážíte inkrementální zálohu, poměr je ještě daleko menší. Velikost operačního systému v poměru s tím, co by mělo být zahrnuto v inkrementální záloze, je téměř zanedbatelná. Ale jen pokud zde není nic důležitého, co by mělo být zálohováno. Týká se to souborů */etc/passwd*, */etc/hosts*, *syslog /var/adm/messages* a další, které by byly důležité při ztrátě operačního systému.

Složitost

Námítka zastánců částečné zálohy je zde na místě, protože existuje speciální záloha, která slouží k zálohování důležitých souborů. Ale problém spočívá v tom, že je to o mnoho pracnější, než zálohování všeho. Představte si, že nejsou zahrnuty konfigurační soubory z většiny záloh, a je nutností zálohovat data pokaždé, kdy změníte tyto konfigurační soubory. To znamená, že vždy musíte udělat něco navíc, v případě změny konfigurace, či modifikaci těchto souborů. V případě zálohování celého operačního systému jako celku, není potřeba se dále zabývat touto starostí.

Rozdělení mezi svazky

Jednou z mála věcí, které by mohly být považovány za plus je rozdělení disků nebo souborových systémů do více záloh. Je snadnější je pak rozdělit mezi více svazků. Pokud se záloha systému nevejde na jeden svazek, je jednodušší automatizovat tento proces rozdělením na dva rozdílné tzv.include listy. Ale aby to byla opravdu výhoda, je třeba použít include listy, ne exclude listy.

Shrnutí

Největším pozitivem je jednak ušetřený čas scriptováním, ale i ušetření pár bytů tekoucích po síti. Nejhorším možným vedlejším efektem je přehlédnutí některých souborů či oddílů.

2.6 Zálohování celého systému

Daleko stručnějšími jsou klady pro zálohování celého systému, než v případě předcházejícího postupu.

Kompletní automatizace

Tímto případem je možno zvýšit zátěž sítě o několik procent, ale jakmile si uvědomíte problémy s vytvořením scriptu či programu, který bude fungovat a následného vytvoření monitorovacího systému, který podává zprávy o operacích, můžete s klidným svědomím říct, že jste udělali maximum.

3 KDY ZÁLOHOVAT

3.1 Stupně

Nejprve je třeba definování několika termínů

Full/Level 0

Plná záloha.

Level 1

Inkrementální zálohou je zálohováno vše, co se změnilo od zálohy Level 0. Opakovanou zálohou je stále zálohováno vše, co se změnilo od zálohy Level 0.

Level 2-9

Každá záloha tohoto stupně zálohuje vše, co se změnilo od zálohy stupně předcházejícího. Záloha stupně 2, uloží vše, co se změnilo v záloze stupně 1 nebo stupně 0, pokud chybí stupeň 1.

Inkrementální

Obvykle záloha, která zálohuje vše, co se změnilo od poslední zálohy jakéhokoliv typu.

Diferenciální

Většina lidí si myslí, že diferenciální záloha zálohuje všechno to, co se změnilo od poslední plné zálohy, ale není to vždy pravda. U rozdílové zálohy jsou zálohovány pouze soubory, u nichž je nastaven atribut *archive*, zálohují se tak soubory, u kterých byl od poslední plné zálohy ručně nastaven atribut *archive*.

Kumulativní inkrementální

Záloha všech souborů, které se změnilo od poslední plné zálohy

Týdenní rozvrh: Jednou za týden stupeň level 0, denní level 1 - 6

Tento plán popsany v tabulce (Tab. 3), počítá s nedělní plnou zálohou a šesti úrovní záloh po zbytek týdne. Výhodou toho plánu je, že zabere méně času a využívá méně medií, než plán předchozí. Mezi dvě nevýhody, které bych zmínil, patří zejména to, že každý změněný soubor dostane zálohu pouze jednou, což přináší riziko v případě, kdy by jedno z médií vykazovalo následně chybu. Za druhé, budete potřebovat šest svazků , abyste obnovili páteční zálohu. .[1]

Tabulka 3 Jednou za týden stupeň level 0, denní level 1 - 6

Neděle	Pondělí	Úterý	Středa	Čtvrtek	Pátek	Sobota
Full/0	1	2	3	4	5	6

Týdenní rozvrh: : Jednou za týden stupeň level 0, denní TOH inkrementální

Jedním z nejzajímavějších nápadů je zálohování, které se jmenuje Tower of Hanoi (TOH). Je založen na pradávne matematické hře puzzle. Hra je složena ze tří kolíků a množství různých velikostí kroužků vložených na kolíky. Kroužek nesmí být umístěn na horní části kroužku s menším poloměrem. Cílem hry je přesunout všechny kroužky z prvního kolíku na třetí kolík pomocí druhého kolíku, pro dočasné úložiště v případě potřeby

Cílem většiny zálohovacích plánů je, aby změněné soubory byly umístěny na více než jeden svazek a zároveň na co nejmenším počtu svazků. TOH plán toho dosahuje lépe, než kterýkoliv jiný zálohovací plán. Pokud používáte TOH plán pro zálohování, většina souborů jsou zálohovány dvakrát - ale maximálně dvakrát. Jsou dvě možnosti TOH plánu.

0 3 2 5 4 7 6 9 8 9

0 3 2 4 3 5 4 6 5 7 6 8 7 9 8

Tyto matematické postupy jsou v podstatě jednoduché. Každý z nich je složen ze dvou proložených sérií čísel (2 3 4 5 6 7 8 9 v prolínání spolu s 3 4 5 6 7 8 9). Demonstrace v tabulce (Tab. 4). [1]

Tabulka 4 Jednou za týden stupeň level 0, denní TOH inkrement.

Neděle	Pondělí	Úterý	Středa	Čtvrtek	Pátek	Sobota
0	3	2	5	4	7	6

Startuje se stupněm 0 (plná) v neděli. Předpokládejme, že se soubor změní v pondělí. Stupeň 3 v pondělí zálohuje vše, co se změnilo od zálohy stupeň 0, takže změněný soubor bude zařazen v pondělní záloze. Když v úterý změním další soubor, potom v úterý večer, musí hledat stupeň zálohy menší než level 2, to je nedělní záloha. Ve středu level 5 zálohuje jen to, co se změnilo též den, protože se odvolává na zálohu stupně 2 v úterý. Ale ve čtvrtek stupeň 4 se neodkazuje na zálohu ze středy, ale odkazuje na zálohu stupně 2 v úterý, atd. [1]

Všimněte si, že soubor, který se změnil v úterý, je zálohován pouze jednou. Aby se předešlo tomuto nedostatku, použijeme modifikovaný TOH plán znázorněný v tabulce (Tab. 5).

Tabulka 5 Měsíční plán systémem TOH

Den	1.týden	2.týden	3.týden	4.týden
Neděle	0	1	1	1
Pondělí	3	3	3	3
Úterý	2	2	2	2
Středa	5	5	5	5
Čtvrtek	4	4	4	4
Pátek	7	7	7	7
Sobota	6	6	6	6

Pokud nejste zmateni a pokud tuto metodiku podporuje váš zálohovací program/script, je doporučeno rozvrhnout plán podle tabulky (Tab. 5). Každou neděli dostanete kompletní inkrementální zálohu všeho, co se změnilo od měsíční plné zálohy. Během zbývajících týdnů, je každá změna zálohována dvakrát, kromě středečních souborů. Tento plán chrání vaše data před chybou média lépe, než všechny předchozí zmiňované plány. Bude potřeba více než jeden svazek v případě obnovy, ale to by neměl být problém, pokud vlastníte sofistikovaný program/script se svazkovým manažerem.

3.3 Další

Zálohování by mělo být naplánováno takovým způsobem, aby nenarušovalo normální běh společnosti. Někdy se tomu ovšem nelze vyhnout, ale nemělo by se to stávat často. Zde jsou dva důvody.

3.3.1 Integrita

I v případě, že jste v práci 24/7, noční doba je doba, kdy jsou soubory většinou stabilní. Samozřejmě jsou zde *batch* programy, které mohou běžet na pozadí a manipulovat s daty. Zákazníci mohou přistupovat na vaše stránky, takže ne všechny soubory mohou být stabilní. Pokud zálohujete ve dne, soubory se neustále mění a pravděpodobně jsou také dost možná otevřené. Otevřené soubory je potom těžké zálohovat. Některé programy zvládají zálohování otevřených souborů lépe, než ostatní, a některé je zálohovat prostě nedokážou. Ale v případě, že se se souborem manipuluje během dne, nikdy nebudete mít jistotu, jakou verzi právě zálohujete. [1]

3.3.2 Rychlost

Dalším důvodem, proč neprovádět zálohu během dne je, že zkrátka rychlost sítě není uprostřed dne plnohodnotná. Rychlost vaší zálohy pak bude značně zpomalená i v případě normálního provozu sítě. Může se stát, že tento problém nastane i v noci, potom je dobré zvážit variantu, kdy pro vaše zálohy zvolíte použití speciálně vytvořené sítě. Jednak vašimi zálohami značně zpomalíte ostatní aplikace a není zrovna moudré zpomalovat síť, pokud jí zrovna potřebují uživatelé. [1]

4 JAK ZÁLOHOVAT

4.1 Zálohovací programy

Existuje mnoho již vytvořených komplexních zálohovacích systémů, např. Duplicity, Clonezilla, Back in Time, Time Vault, nebo Bacula, na kterou se zaměřím.

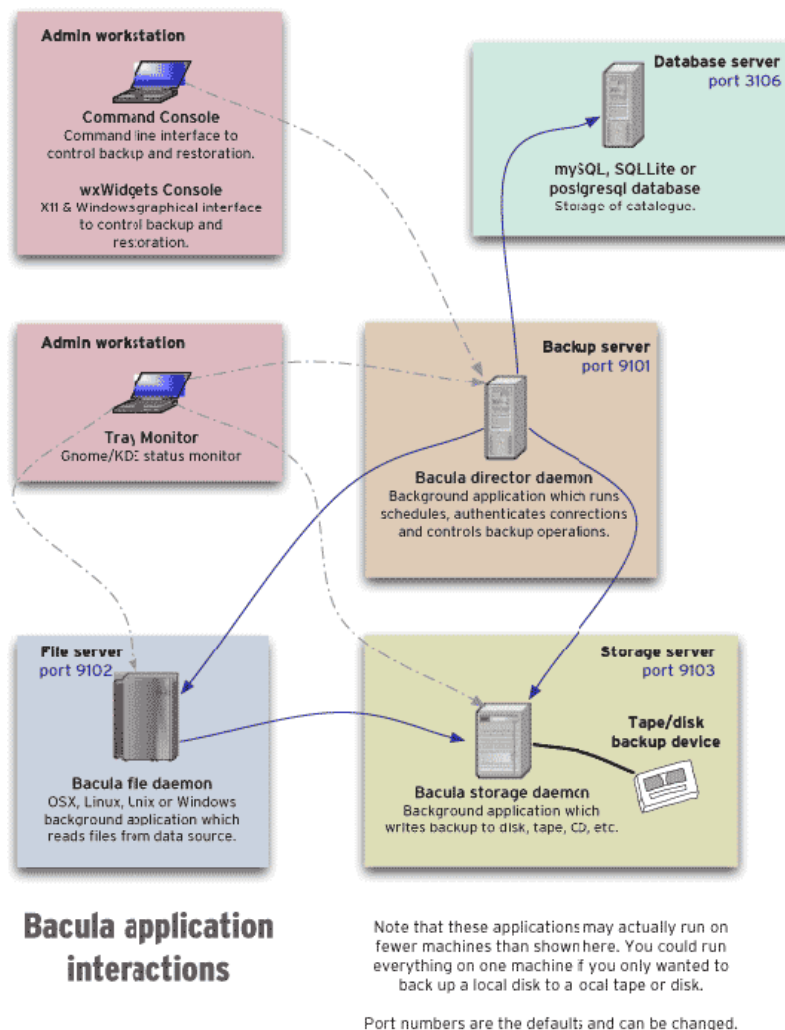
4.1.1 Bacula

Bacula je open-source zálohovací program navržený tak, aby vyhověl uživatelům jednak jednoho počítače až po podnikové síť. Je schopený zálohování jakékoliv kombinace disků, kazet, či optických médií. Server běží na Linuxu nebo Unixu, který hostí klienty různých systémů, jako například Unix, Linux, post-Win95 SE Windows, Mac OS X.

Stránka projektu se nachází na <http://www.bacula.org>

4.1.2 Složení

Bacula je tvořen z těchto pěti hlavní komponentů: Director, Console, File, Storage a Monitor. Každý vykonává jinou část práce. Některé z těchto komponentů jsou instalovány pouze na straně klientů, některé jen na pracovních stanicích. Na obrázku (Obr. 1) je znázorněno detailní schéma.



Obrázek 2 Interakce mezi aplikacemi podle zdroje [5b]

Director

Je to služba dohlížející na všechno zálohování, obnovování, ověření a operaci s archivy. Správce systému používá Director pro plánování záloh a obnovu souborů. Director běží jako daemon (neboli služba) v pozadí. Director potřebuje přístup k DBMS engine², ve kterém uchovává katalog informací o zálohách, má přístup k běžícím Storage službám, aby uložil aktuální svazek, který ukládá. Je běžné, že tyto tři položky běží na stejném serveru (zálohovací server) v malých či středních instalacích. [5b]

² DBMS (A Database Management System) softwarové vybavení, zajišťující práci s databází. [6b]

Console

Program, který umožňuje správci systému nebo uživateli komunikovat s Director. V současnosti je Console dostupná ve třech verzích. Textové rozhraní, rozhraní založené na GNOME a wxWidgets grafické rozhraní. Prvním a nejjednodušším krokem je spustit Console v shellu. Pro většinu správců systémů je toto rozhraní naprosto dostačující. Druhé rozhraní, založené na GNOME, které ještě není zdaleka dokončené, ale již je plně funkční, má naprostou většinu schopností textového rozhraní spouštěného v shellu. [5b]

File

Služba file (nebo také klientský program) je softwarový program. Je nainstalován na zařízení, na kterém budeme provádět zálohu. Je to specifický operační systém, který běží a je zodpovědný za poskytnutí atributů souboru a dat v případě, když si o ně zažádá Director. Tento program běží jako daemon na stroji, na kterém zálohujeme. [5b]

Storage

Storage služba se skládá ze softwarových programů, které provádějí uchovávání a obnovu atributů souboru a dat na fyzická záložní média nebo svazky. Jinými slovy, Storage daemon je zodpovědný za čtení a zápis na pásky (nebo jiná média). Daemon této služby pak běží na stroji, která má zálohovací zařízení. (pásková jednotka). [5b]

Catalog

Tato služba je složena ze softwarových programů, odpovědných za udržování indexů souborů a databází pro všechny zálohované soubory. Služba Catalog povoluje systémovému správci či uživateli rychle najít a obnovit jakýkoliv požadovaný soubor. Tímto se liší od ostatních jednoduchých programů, jako například tar a bru. Udržuje záznamy o všech zálohách, právě prováděných operacích atd.

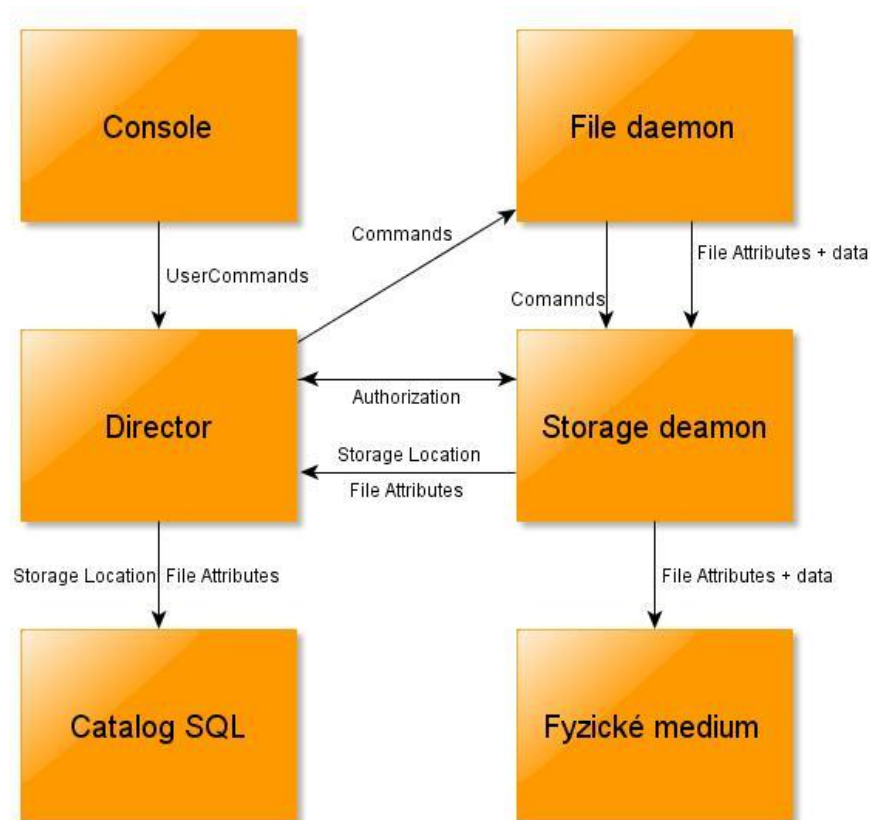
Bacula v současné době podporuje tři různé databázové služby, PostgreSQL, MySQL a SQLite, jedna z nich musí být zvolena pro instalaci. [5b]

Monitor

Poslední služba, ze které se skládá program bacula, která umožňuje systémovému správci či uživateli sledovat aktuální stav všech daemonů, jednak Director, File, nebo Storage daemon. V současné době je k dispozici pouze GTK+ verze, která pracuje s prostředím GNOME, KDE, nebo jakýmkoliv manažerem, který podporuje FreeDesktop.org.

Chcete-li provést úspěšné uložení či obnovení, všechny 4 daemony musí být řádně nakonfigurovány a musí běžet (Director, File, Storage, Catalog - MySQL, PostgreSQL nebo SQLite). [5b]

Interakce mezi službami Bacula znázorněná obrázkem (Obr. 2).



Obrázek 3 Informační tok v programu Bacula podle zdroje [5b]

4.1.3 Popis

SQL katalog

Zde je obsažen seznam souborů, které byly zálohovány, základní atributy každého souboru, včetně kontrolního součtu, ze kterého klienta soubor pochází a svazky, na kterých se nacházejí zálohované soubory. [5b]

Multisvazková záloha

Jedna z rozlišujících se vlastností od ostatních programů je vynikající nativní podpora pro multisvazkové zálohy. V případě, kdy je stroj vybaven měničem, bacula dokáže spolupracovat s tímto měničem tak, že při páskovém zálohování dokáže měnit pásky bez jakéhokoliv zásahu člověka. [5b]

Flexibilní podpora medií

Bacula může zálohovat na různé typy medií, používá disky, stejně tak snadno jako pásky. SCSI měniče fungují systémem plug-and-play. Bacula také obsahuje nástroje pro snadné přemístění na CD (bimagmgr) nebo DVD (dvd-handler). [5b]

Úrovně záloh

Stejně jako u většiny ostatních zálohovacích systémů, bacula rozlišuje mezi plnou zálohou, diferenciální zálohou a inkrementální zálohou. Úroveň zálohy je potom specifikována v naplánovaném kalendáři. Pokud nenalezne plnou zálohu, je možné, že "povýší" inkrementální, či diferenciální zálohu na zálohu plnou. [5b]

Formát uložení

Všechny data, které Bacula zálohuje, jsou uloženy na svazcích. Tyto svazky, tzv. repositáře slouží pro zálohovaná data, i v případě, že je to páska, optické médium, nebo jenom soubor. Bacula používá svůj vlastní unikátní formát, na rozdíl od nástrojů jako tar a dump. Tento formát tak zajišťuje konzistenci dat napříč různými platformami. I když tato vlastnost má nesporné výhody, znamená to také, že je potřeba speciálních nástrojů k extrakci zálohy v případě, kdy nemáte řádnou a fungující instalaci Baculy. Nicméně, Bacula obsahuje tzv. stand-alone programy, kterými můžete obnovit zálohu v případě, kdy by nefungovala databáze. Další klíčový prvek spočívá v tom, že blokový design Baculy umožňuje více File daemonům spolupracovat s jedním Storage deamonem, ačkoliv proces obnovy je zpomalen z důvodů více současných přístupů k Storage deamonu. [5b]

Standalone obnova

Bacula umí vytvořit bootovací Linux CD. To obsahuje vše nezbytné, pro zálohu a obnovu systému z tohoto CD. Existuje také podpora pro Solaris, FreeBSD a Windows. [5b]

Dokumentace

Dokumentace je Achillova pata mnohých open-source projektů. Díky bohu, toto není případ Bacula, ačkoliv manuál není úplně stoprocentní. Přesto je skvěle a přesně zdokumentován. [5b]

Škálovatelnost

Bacula je navržena zvláště pro podnikové organizace. Je velmi dobře navržená, především její logicky uspořádané komponenty. V případě většího podniku pak není problém instalace rozšířit. [5b]

Testování páskovacího zařízení

Jedna z nejfrekventovanějších úvah, co se týče programu Bacula je, zda bude pracovat s daným páskovacím zařízením. Ačkoliv online manuál nabízí seznam zařízení, která spolehlivě spolupracují, mnohem sofistikovanější odpověď nabízí program *btape*. Tento program umí zjistit konfiguraci páskového zařízení a spustí několik testů, včetně zápisu, čtení a několik dalších operací. Pokud tento program dokončí testy s pozitivním výsledkem, je jistotou, že Bacula bude spolehlivě pracovat s daným páskovacím zařízením. [5b]

4.2 Zálohování v Linuxu

4.2.1 Jak to funguje?

Cílem je provést bare-metal (do původního stavu), dále jen obnovu, Intel systému bez použitých komerčních nástrojů. Většina systémových administrátorů pro typickou obnovu upřednostňuje instalovat minimální operační systém, přes který se provede obnova. Tato procedura představuje hned několik rizik a problémů. Může nastat konflikt s otevřenými soubory. Procedura trvá poměrně dlouho a je těžké zdokumentovat a následně obnovit modifikace operačního systému.

Zde se podíváme na proceduru, která nepotřebuje reinstalaci systému za účelem obnovy zálohy. Můžeme jí popsat šesti kroky

- Záloha důležitých metadat
- Záloha operačního systému nativní utilitou
- Nabootování z alternativního média
- Příprava nové root jednotky
- Obnova bootovacího bloku na novou root jednotku
- Obnova informací operačního systému

Live nebo alternativní boot?

Prvním rozhodnutím, které je třeba učinit je, zda zálohu chceme provést v případě již běžícího systému (live) nebo v případě, kdy budeme chtít nabootovat z alternativního media. V případě Live volby, je pak uživatel odsouzen na platformní filesystemové příkazy. Největší výhodou zálohování systému Live je to, že nevyžaduje žádné prostoje při zálohování, i když se můžou vyskytnout konflikty s již otevřenými soubory nebo databázemi.

Pokud si ovšem může uživatel dovolit vypnout operační systém pro příležitostnou zálohu, je tu alternativní možnost, kdy je operační systém nabootován z disku, jako například linuxové LiveCD distribuce.

(LiveCD je linuxová distribuce která je plně funkční kopií klasického linuxového operačního systému, nabootovaná jednoduše z CD). Alternativní boot, dále alt-bot metoda, nabízí několik nesporných výhod. Není potřeba starost o formát souborového systému. Je to tím, že se zálohuje blok (neboli image), což není možné v případě metody LIVE. Jedinou nevýhodou je to, že bude operační systém po celou dobu zálohy nedostupný. To by neměl být problém pro domácí uživatele a některé malé podniky, ovšem jiní mohou toto omezení shledat jako nepřijatelné.

Obraz celého systému nebo filestystémová úroveň?

Pokud je použita alt-boot metoda, je možný přístup k datům jako filesystem nebo tzv.raw formát diskové jednotky. Pokud se data zálohují použitím raw diskové jednotky (například /dev/hda nebo /dev/hda1), lze říci, že záloha probíhá na úrovni obrazové (image).

Druhá možnost je zálohovat data jako filesystem, připojit je (mount) a použít nástroje jako jsou tar nebo cpio.

Zálohování na obrazové úrovni dává výhodu nestarat se o operační systém. V případě, zálohy všech bytů a provedení následné obnovy na stejný diskový oddíl, všechno bude fungovat. Takto se třeba zálohuje operační systém Windows, pomocí Linuxu. Největší nevýhodou této metody je případ zálohy každého bytu. Nezáleží, zda je využíván či ne. Pokud je 10 GB oddíl, záloha bude mít přesně 10 GB, i v případě, že zde bude jen 1 GB skutečných dat. Zde může pomoci komprese, ale doba, po kterou se systém bude zálohovat se nijak výrazně nesníží. Naopak, při větší kompresi musíte počítat s časovou prodlevou. Nevýhoda se skrývá v případě obnovy, kdy buď můžete obnovit všechno, nebo nic. Jinými slovy, nemůžete vypreparovat soubor ze zálohy a ten po té obnovit.

Celý disk nebo oddělené oddíly?

Pokud používáte alt-boot metodu a rozhodli jste se pro zálohu celého systému (image), je potřeba udělat další rozhodnutí. Zda se bude záloha tvářit jako jeden velký obraz (/dev/hda) nebo několik oddílů (/dev/hda1, /dev/hda2, /dev/hda3).

Zálohovat celý OS jako jeden velký oddíl má jednu velkou výhodu. Následná obnova je neuvěřitelně lehká. Není třeba dělat si starosti o přerozdělování oddílu ani o bootovací blok (Master Boot Record, MBR).

Nevýhodou je velikost dnešních disků. Pokud je zvolen obraz celého OS, bude pravděpodobně potřeba stovek gigabitů. Mimo jiné, takový proces zabere spoustu času, po který váš operační systém bude nedostupný.

Druhá metoda spočívá ve vytvoření několika oddílů a následné zálohy každého oddílu zvlášť. To umožní použít rychlou obnovu jen pro oddíl, na kterém se nachází operační systém. A na zbytek systému použít jeden z nástrojů, jako například Bacula. Který například umožňuje obnovit všechny oddíly najednou. To podstatně zrychlí celý proces.

Mezi hlavní nevýhody patří v porovnání s metodou zálohy celého disku daleko větší složitost. Je potřeba porozumět více věcem. A to jakým způsobem je disk rozdělen na oddíly, ale také zpětné obnově tabulky oddílů a MBR. V případě obnovy, kdy se rozděluje disk na oddíly, je třeba dbát na správnou velikost oddílů. Při vytvoření malého oddílu, nebude následná obnova fungovat. Pokud se vytvoří oddíl příliš velký, znamená to mrhání místem na disku

4.2.2 4 možnosti zálohy

Tři předchozí metody popsané čtyřmi zálohovacími způsoby.

Alt-boot full image

Tento způsob je složen z nabotování LiveCD a vytvoření obrazu celého systému. Je to nejjednodušší úloha a bude fungovat pro většinu lidí. Bohužel potřebuje hodně času, po který bude operační systém vypnutý a také spotřebuje spoustu volného místa.

Alt-boot partition image

Způsob podobný předchozímu, s tím rozdílem, že obraz bude vytvořen jen z oddílu, který bude potřeba zálohovat. Tento způsob ušetří jednak čas, tak i místo, ale vyžaduje větší znalosti souvislostí okolo zálohování.

Alt-boot filesystem

Je složen z nabotování LiveCD, např. starší distribuce Knoppix, nebo jakékoliv jiné distribuce a následného používání integrovaných programů jako jsou například dd, tar, bzip. Výhodou je ušetřené místo za předpokladu, že váš systém není zaplněný daty. Jedná se o nejkomplicovanější způsob a stále se musí počítat s vypnutým operačním systémem po dobu zálohy.

Live

Pokud uživatel používá Linux a zároveň nepoužívá LVM, RAID, či IA64, lze provést zálohu v případě, kdy bude operační systém běžet. Tento způsob je komplikovanější za použití dual-bootu, protože takovéto systémy nemají běžný filesystem. Uplatnění se najde v případě, pokud váš operační systém nemůže být vyřazen z provozu ani na minutu

Kroky v teorii

Je třeba rozebrat si v teorii kroky, které je potřeba udělat.

Krok 1: Zálohovat důležité metadata

Nejdůležitějším krokem je nejdříve uložit metadata. Jsou to data obsahující informace o tom, jak je hardware fyzicky nakonfigurován. V tomto případě, jakým způsobem je rozdělený disk na oddíly. Tato informace se obvykle nenachází v normální záloze. Proto je dobré udělat kopii MBR a tabulky oddílů. MBR a tabulka oddílů se nachází v prvních

512 bajtech vašeho harddisku. MBR se skládá ze tří částí: zaváděcí blok je uložen v prvních 446 bajtech, tabulka oddílů v dalších 64 bajtech a pro spouštěcí kód zůstává 2 bajty.

Pokud chceme uložit tuto informaci na platformě Linux, spustíme tyto příkazy, které nám uloží informace o rozdělení disku:

```
4 # fdisk -l > /backup/fdisk.txt
5 # cp /etc/fstab /backups/fstab.txt
```

Pro zálohu MBR a tabulky oddílů spustíme tento příkaz, který vytvoří soubor /backups/mbr

```
6 # dd if=/dev/hda of=/backups/mbr bs=512 count=1
```

Toto poslouží v případě obnovy.

Krok 2: Záloha OS za pomoci nativního programu

V případě použití Live metody, se budou hodit programy jako tar a cpio. Tar je jeden z nejpopulárnějších programů určených nejen k záloze. Navíc je více přenositelnější, než cpio. Pokud uživatel tedy zvolí alt-boot filesystem metodu, bude muset porozumět jednak slovům jako filesystem, mount, tar, cpio. Pokud zvolí metodu alt-boot partition image, jedině, co bude potřebovat je pouze program dd a jeden příkaz.

Krok 3: Bootování systému z alternativního media

Aby bylo možné provést obnovu, je potřeba zajistit minimálně funkční shell, který uživatel dostane v případě, že nabojuje LiveCD. Zbytek procedury se pak provede s již plně funkčním shellem.

Krok 4: Obnovení bootovacího bloku

Zaváděcí blok je zvláštní část disku, která umožní načtení operačního systému. Obsahuje jen pouze nejnutnější množství kódu k vyhledání a nahrání kernelu. Na platformě Intel se tento blok nazývá také jako MBR, lze jej obnovit pomocí programu dd ze zálohy, která byla vytvořena v kroku 1.

Obnova se provede tímto příkazem:

```
7 # dd if=/backups/mbr of=/dev/hda bs=512 count=1
```

V případě, kdy chceme, aby operační systém z LiveCD poznal, že jsme nainstalovali nový MBR, je nutné spustit *fdisk /dev/hda* a zvolit *w* pro zápis na oddíl.

Krok 5: Oddíly a formátování disku

Kroky uvedené výše obnoví MBR a tabulku oddílů, takže není potřeba přerozdělovat oddíly znovu. Pokud je použita Live metoda nebo alt-boot filesystem metodu, je nutno vědět, jakým způsobem vytvořit filesystem za pomoci *mkfs* příkazu. Dále je potřeba zálohovat *fstab*, kvůli informacím, jaké oddíly mají mít jaký filesystem. V případě metod alt-boot full image nebo alt-boot partition image, není potřeba dělat si starosti s formátováním. Tento proces se udělá automaticky při obnově z obrazu (image).

Krok 6: Obnova OS na nový disk

Nejprve je potřeba zajistit přístup k datům, která jsou již zálohovaná. Například, pokud je použit NFS, lze to udělat následujícím způsobem:

- Nainstalovat druhý harddisk ve stejné nebo větší velikosti
- Nainstalovat síťové a NFS ovladače, nakonfigurovat síť a připojit (mount) NFS filesystem.
- Nainstalovat síťové a SMB/CIFS ovladače, nakonfigurovat síť a připojit SMB/CIFS filesystem.

Předpoklady

Následující části obsahují postupy pro každou ze čtyř metod, které jsou popsány výše. Alt-boot full image, alt-boot partition image, alt-boot filesystem a live.

Tyto procedury jsou založeny na několika předpokladech, které zde budou uvedeny.

Předpoklad pro daný příklad je, že DHCP je k dispozici. Je schopen přiřadit IP adresu vždy, kdy se nabootujeme z LiveCD. V případě, že by DHCP nebylo dostupné, je potřeba manuálně nastavit IP adresu. Lze to provést příkazem

```
8 $ su -
9 # ifconfig eth0 ipadresa up
10 # route add default gw ipadresa-brany
```

Je třeba zvolit funkční IP adresu, která komunikuje s ostatními počítači v síti, nebo IP adresu DNS serveru do souboru */etc/resolv.conf* (ve formě: *jmenoserveru ipadresa-dns-serveru*).

Další předpoklad spočívá v tom, že je použit nakonfigurovaný NFS server, s dostatkem volného místa pro uložení zálohy. V případě, kdy by chtěl uživatel uložit data ve Windowsovském tvaru, k tomu slouží příkaz:

```
11 # mount -t smbfs -o username=administrator, password=heslo
    //adresaserveru/share /backups
```

4.2.3 Alt-Boot Full Image Metoda

Je nejjednodušší metodou, zejména pro uživatele, kteří nemají velké zkušenosti s Linuxem. Operační systém musí být vypnutý, funguje s jakýmkoliv operačním systémem a musí zde být dostupné úložiště s dostatečně velkým prostorem pro zálohu celého disku.

Postup:

Vytvoření zálohy

Záloha důležitých metadat

Tento krok lze přeskočit, protože metadata jsou zálohována a obnovena v případě, kdy je celý disk zálohován jako jeden obraz (image).

Bootování systému z alternativního media.

Nejprve je potřeba vložit LiveCD do mechaniky, restartovat počítač a zvolit bootování systému z CD nebo Flash disku. Také přepnout se na uživatele root, vytvořit složku a do ní připojit NFS server.

```
12 $ su -
13 # mkdir /backups
14 # mount nfserver:/data08/upload /backups
```

Zálohování OS nativní utilitou

Celý disk lze zálohovat disk příkazem *dd*, přímo do NFS složky. Tento příkaz specifikuje, že root disk (*/dev/hda*) se přenesse do složky */backups* pod názvem */hda.dd*.

```
15 # dd if=/dev/hda of=/backups/hda.dd
```

V případě úspory místa, lze spustit následující příkaz, ale za cenu celkově pomalejší zálohy.

```
16 # dd if=/dev/hda |gzip -c > /backups/hda.dd.gz
```

Obnova:**Bootování systému z alternativního media**

V první řadě zopakovat postup s vložením LiveCD a nabootování systému, spuštění terminálu a zadání následujících příkazů.

```
17 $ su -
18 # mkdir /backups
19 # mount nfserver:/data08/upload /backups
```

Obnovení bootovacího bloku:

Tento krok lze vynechat v případě, kdy se provádí obraz celého disku.

Oddíly a formátování disku:

Tento krok lze vynechat v případě, kdy se provádí obraz celého disku.

Obnova OS na nový disk:

Pro obnovu analogicky spustit tento příkaz.

```
20 # dd if=/backups/hda.dd of=/dev/hda
```

V případě použití komprese za pomoci gzipu, je potřeba soubor nejprve soubor rozbalit a poté jej obnovit. To se provede následujícím příkazem.

```
21 # gzip -dc /backups/hda.dd.gz | dd of=/dev/hda
```

Poslední krok je vyjmout CD a restartovat systém, nyní by mělo být vše obnoveno.

4.2.4 Alt-Boot Partition Image metoda

Tento příklad také používá příkazu *dd* a je více komplexnější. Systém musí být stále vypnutý, ale v případě, kdy se bude zálohovat pouze oddíl s operačním systémem, značně se urychlí práce.

Postup:**Vytvoření zálohy****Záloha důležitých metadat**

Pro zálohu MBR a tabulky oddílů je potřeba spustit tento příkaz.

```
22 # dd if=/dev/hda of=/backups/mbr bs=512 count=1
```

Bootování systému z alternativního media.

Nejprve se vloží LiveCD do mechaniky a restartuje počítač. Je potřeba zvolit bootování systému z CD nebo Flash disku, zadat tyto příkazy.

```
23 $ su -
24 # mkdir /backups
25 # mount nfserver:/data08/upload /backups
```

Zálohování OS nativní utilitou

Zálohování se provede příkazem *dd*. Zde se ovšem liší tento postup od předchozího. Nejdříve se musí vyhledat oddíly, které budou zálohovány.

```
26 # fdisk -l
```

Výstup bude v podobném tvaru

```
27 Device Boot Start End Blocks Id System
28 /dev/hda1 * 1 13 104391 83 Linux
29 /dev/hda2 14 4874 39045982+ 83 Linux
30 /dev/hda3 4875 5005 1052257+ 82 Linux swap / Solaris
```

Následnými příkazy se zálohují oddíly, které jsou potřeba zálohovat.

```
31 # dd if=/dev/hda1 of=/backups/hda1.dd
32 # dd if=/dev/hda2 of=/backups/hda2.dd
```

Pokud chce uživatel použít kompresi, použij příkazy analogicky.

```
33 # dd if=/dev/hda1 | gzip -c > of=/backups/hda1.dd.gz
34 # dd if=/dev/hda2 | gzip -c > of=/backups/hda2.dd.gz
```

Je zde možnost použít znaku & (ampersand) na konci, to umožní, aby se tyto příkazy přenesly do pozadí, ale je potřeba dávat pozor a neustále monitorovat tyto procesy.

Obnova:**Bootování systému z alternativního media.**

V první řadě je potřeba zopakovat postup s vložením LiveCD a naboťování systému, spuštění terminálu a zadání následujících příkazů. Nejprve zkontrolovat tabulku oddílů.

```
35 # fdisk -l
```

Výstup zobrazí upozornění, že disk /dev/hda nemá validní tabulku oddílů.

```
36 Disk /dev/hda doesn't contain a valid partition table
```

Tímto se nyní uživatel nemusí zabývat a vytvoří složku, do které připojí NFS server.

```
37 $ su -
38 # mkdir /backups
39 # mount nfserver:/data08/upload /backups
```

Obnovení bootovacího bloku, příprava disku:

Nejprve je třeba obnovit MBR a tabulku oddílů

Obnova se provede příkazem:

```
40 # dd if=/backups/mbr of=/dev/hda bs=512 count=1
```

V případě, aby Knoppix nebo kterákoliv jiná distribuce obsažená na LiveCD poznal, že jsme nainstalovali nový MBR, je nutné spustit *fdisk /dev/hda* a zvolit *w* pro zápis na oddíl. Samotný restart by trval poněkud delší dobu.

Obnova OS na nový disk:

Nyní je ve skutečnosti dokončena příprava obnovení operačního systému. Použije se příkaz *dd* v obráceném pořadí pro každý svazek zvlášť.

```
41 # dd if=/backups/hda1.dd of=/dev/hda1
42 # dd if=/backups/hda2.dd of=/dev/hda2
```

V případě komprese pak:

```
43 # gzip -dc /backups/hda1.dd.gz |dd of=/dev/hda1
44 # gzip -dc /backups/hda1.dd.gz |dd of=/dev/hda1
```

4.2.5 Live metoda

V tomto postupu je hojně využit program *tar*, zároveň umožňuje tato metoda, aby byl OS online.

Postup

Vytvoření zálohy:

Záloha důležitých metadat

Pro zálohu MBR a tabulky oddílů spustit tento příkaz

```
45 # dd if=/dev/hda of=/backups/mbr bs=512 count=1
```

Bootování systému z alternativního media.

Nejprve je třeba vložit LiveCD do mechaniky, restartovat počítač a zvolit bootování systému z CD nebo Flash disku. Po restartování se nahraje prostředí obsažené v distribuci.

Nyní je potřeba vytvořit složku a připojit NFS server.

```
46 # mkdir /backups
47 # mount nfserver:/data08/upload /backups
```

Nyní lze přejít k samotné záloze, pro tu lze zvolit nástroj tar. Za předpokladu, že jeden z oddílů je připojený jako složka /boot, zbytek OS jako /.

```
48 # cd /boot
49 # tar cf /backups/boot.tar .
50 # cd /
51 # tar cf /backups/system.tar --excluce /mnt --exclude /proc --
    exclude /boot --exclude /backups .
```

Existuje zde samozřejmě verze s kompresí, opět zde platí, že celkový proces zálohy tato komprese může značně zpomalit.

```
52 # cd /boot
53 # tar cfz /backups/boot.tar.gz .
54 # cd /
55 # tar cfz /backups/system.tar.gz --excluce /mnt --exclude /proc --
    exclude /boot --exclude /backups .
```

Obnova:**Bootování systému z alternativního media.**

V první řadě zopakovat postup s vložením LiveCD a nabootování systému, spuštění terminálu a zadání následujících příkazů. Nejprve zkontrolovat tabulku oddílů

```
56 # fdisk -l
```

Výstup zobrazí upozornění, že disk /dev/hda nemá validní tabulku oddílů

```
57 Disk /dev/hda doesn't contain a valid partition table
```

Tímto se nyní uživatel nemusí zabývat a vytvoří složku, do které připojí NFS server.

```
58 $ su -
59 # mkdir /backups
60 # mount nfserver:/data08/upload /backups
```

Obnovení bootovacího bloku, příprava disku:

Nejprve musíte obnovit MBR a tabulku oddílů.

Obnovu lze provést příkazem:

```
61 # dd if=/backups/mbr of=/dev/hda bs=512 count=1
```

V případě, aby distribuce obsažená na LiveCD poznala, že je nainstalován nový MBR, je nutné spustit *fdisk /dev/hda* a zvolit *w* pro zápis na oddíl. Samotný restart by trval poněkud delší dobu.

Nyní je potřeba připravit oddíly pro obnovu filesystému. Pokud *fstab* při záloze ukazoval, že zde byly ext2 filesystémy, je třeba spuštění těchto příkazů. Samozřejmě je tato část individuální v závislosti na tom, jaký souborový systém je použit.

```
62 # mkfs -t ext2 /dev/hda1 -L /boot
63 # mkfs -t ext2 /dev/hda2 -L /
```

Obnova OS na nový disk:

Zde je potřeba připojit dva oddíly.

```
64 # mount /dev/hda1
65 # mount /dev/hda2
```

Nyní se přepnout do složek */hda1 /hda2* a provést obnovu.

```
66 # cd /mnt/hda1
67 # tar xpkf /backups/boot.tar
68 # cd /mnt/hda2
69 # tar xpkf /backups/system.tar
```

V případě komprese:

```
70 # cd /mnt/hda1
71 # tar xpkfz /backups/boot.tar.gz
72 # cd /mnt/hda2
73 # tar xpkfz /backups/system.tar.gz
```

4.2.6 Alt-Boot Filesystem metoda

Tento postup je skoro totožný, jako postup předchozí, s tím rozdílem, že operační systém musí být po celou dobu operace vypnutý. Zásadní rozdíl mezi touto a Live metodou je pouze to, že u Live metody se nekládá LiveCD do mechaniky a musí se spoléhat na vestavěné příkazy a programy operačního systému.

5 ÚLOŽIŠTĚ ZÁLOH

Nejlepší možná rada je organizace, je potřeba vaše zálohy dobře popsat, dát jim unikátní jména, čísla a seřadit je logicky tak, abyste je v případě obnovy nemuseli hledat. Je potřeba řídit se tím, co skutečně chce daný uživatel zálohovat. V případě neuvážené zálohy, kdy je zvolena metoda "raději vše, nežli nic", se zaplní repositáře daleko dříve.

5.1 On-line úložiště

On-line úložiště je typicky nejdostupnějším úložištěm pro vaše data, která mohou být obnovena řádově v milisekundách. Typickým příkladem je interní harddisk nebo diskové pole. Tento typ úložiště je velmi komfortní a rychlý, ale relativně drahý. Tyto data jsou pak poměrně citlivá vůči odstranění nebo přepsání a to buď díky náhodě, nebo úmyslným zaviněním v případě viru. [3b]

5.2 Off-site

Tento systém ochrany dat vám pomůže uchránit data jednak před katastrofami, nebo před specifickými problémy sítě. Mnoho lidí uchovává své data v off-site úložištích. Tato úložiště mohou být podobná běžné kanceláři, nebo to můžou být sofistikované, vysoce zabezpečené bunkry, které jsou chráněny před jakoukoliv katastrofou. [3b]

5.3 Near-line

Je úložištěm typicky méně přístupným, ale za to levnějším, než on-line úložiště, ale stále užitečné, co se týče úschovy dat. Příkladem je pásková knihovna, která dokáže provést obnovu od několika sekund po několik minut. Jedná se o zařízení, které obsahuje jednu nebo více pásek, čtečku čárových kódů a automatické zařízení, které umí vybrat a načíst žádanou pásku. [3b]

5.4 Off-line

Lidská akce je vyžadována v případě přístupu k dané záloze. Například musí být připojen kabel k zařízení, které umožní přistoupit k záloze, nebo je potřeba manuálně vložit pásku do zařízení. [3b]

6 OBECNÉ

6.1 Testování záloh

Testování záloh je důležitým krokem navrhování skriptu či programu, kdy je potřeba otestovat nejen funkčnost programu, ale také navrženého systému záloh. Může se stát, že bude záloha otestována až v případě, kdy dojde k nějaké katastrofě či selhání a bude marné snažit se obnovit nefungující zálohu.

Zde je několik tipů, kterých je třeba držet se v případě testování záloh:

- Obnova vícero samostatných souborů (z důvodů, kdy bude potřeba najít jeden specifický soubor).
- Obnova starších verzí souborů.
- Obnova celého disku nebo filesystému, porovnání výsledků s originálem (zejména velikost).
- Předstírání, že celý systém postihne katastrofa, snaha jej znovuobnovit.
- Předstírání, že svazek se zálohou je částečně poškozený, pokus použít jinou zálohu.
- Obnova pár svazků z off-site úložiště.
- Představa, že váš celý zálohovací server se porouchal, provedení všech nutných kroků k obnově do původního stavu.

V případě obnovy databáze je vhodné postupovat podle těchto rad:

- Zkouška obnovení jen části databáze.
- Obnovení celé databáze na jiný server, takto se uživatel dozví, které soubory zapomněl uvést do zálohy.

Důležitým úkonem je časté testování, jelikož obnova, která funguje dnes, nemusí fungovat zítra. Jedinou konstantní věcí zde zůstává změna. Jeden návrh je vytvoření seznamu způsobů využití postupů a náhodné testování jedné podmnožiny z celého seznamu každý měsíc. Změny hardwaru, řízení, sítě, operačního systému, atd.

6.2 Monitorování záloh

V případě, kdy operace týkající se záloh, není monitorována, uživatel neví vůbec o ničem, co se děje. Každá záloha by měla mít svůj log, který by měl být denně aktualizován. Tato aktualizace je jednoduchým krokem. Příkladem je program `dump`. Je zásobárnou stavových zpráv, některé jsou důležité, jiné méně. Co je skutečně potřeba vědět, je populární zpráva `DUMP IS DONE`, kterou uživatel dostane za použití příkazu `grep -v`, který vyřadí fráze, které nejsou potřeba a zůstanou pouze žádoucí zprávy stavu.

6.3 MÉDIA

POPIS MEDIÍ

- **Magnetická media**

- magnetické pásky

- pevné a pružné disky

- **Optická media**

- optický disk

- přepisovatelný optický disk

- CD-ROM DVD-ROM

- **Disková pole RAID**

6.4 Technologie RAID

Tato technologie je způsob práce se dvěma či více pevnými disky, jako s jednou logickou datovou jednotkou (blokovým zařízením). RAID nelze považovat za jistý typ zálohování. Primární úlohou RAIDu je pak zajištění vyšší rychlosti, nebo zajištění proti výpadku pevného disku na straně serveru.

Na základě úrovně RAIDu poskytuje kompromis mezi odolností proti výpadku jednoho či více disků, výkonem a kapacitou. Existuje několik typů RAIDu. Je vhodné se zaměřit na ty nejpoužívanější a nejzajímavější. [4b]

Hlavní cíle RAID

Redundance

Redundance (nadbytečnost) představuje uložení zvláštní informace, která je následně využita pro její obnovení, v případě výpadku jednoho disku. Redundance se využívá například při zrcadlení disku, kde dochází ke zdvojení disku. Logický disk se skládá ze dvou fyzických disků. Zápis informace je proveden na oba disky a při závadě na jednom disku jsou data dostupná na disku druhém. [7]

Paralelizace

U diskových systémů je hlavním cílem paralelizace vyvážení zátěže pro zvýšení propustnosti, snížení času odezvy u velkých objemů dat a zlepšení přenosové rychlosti za pomoci dělení dat na více disků. [7]

RAID 0

Data jsou rozmístěna střídavě na všech discích z pole. V případě ztráty jednoho disku jsou ztracena data z celého pole. Nejlepší výkon v poměru všech ostatních stupňů RAIDU. Kapacita pole je rovna počtu disků [4b]

RAID 1

Data jsou zrcadlena na všechny disky z pole. Kapacita pole je stejně velká jako velikost nejmenšího disku z pole. V případě čtení lze číst z více disků současně, čtení je rychlejší. Zápis je pomalejší, vzhledem k tomu, že data jsou zapisována na všechny disky současně. Data v poli jsou bezpečná v případě selhání $n-1$ disků. Kde n se rovná počtu disků v poli.

[4b]

RAID 5

Základním minimem jsou zde tři disky. Paritní informace a data jsou rozmístěna střídavě na všechny disky. Kapacita je zde $n-1$ disků, kde platí stejné pravidlo, jako u RAID 1. Čtení dat se dá přirovnat k rychlosti RAID 0, zápis je zde pomalejší, kde se musí k datům počítat parita. [4b]

RAID 6

Minimem jsou zde disky čtyři. Rozdíl je zde v tom, že v RAIDu 6, na rozdíl od RAIDu 5 jsou přítomny dva paritní bloky. V případě selhání zde platí pravidlo $n-2$, to znamená, že unese selhání výpadku jakýchkoliv dvou disků. Je zde znatelná náročnost na výpočetní výkon, kde se počítají dva paritní bloky. [4b]

RAID 10

Kombinace RAIDu 1 a RAIDu 0. Je nutné podotknout, že právě v tomto pořadí. Vytvoří se dvě pole RAID 1 a nad nimi se postaví pole RAID 0. Výhodou je pak dobrý poměr mezi odolností a výkonem. [4b]

RAID 15 (RAID 51)

Tak jako v případě RAID 10, RAID 15 je kombinace RAIDu 1 a 5. Poskytuje poměrně velkou robustnost, ovšem za cenu úbytku velkého množství kapacity. [4b]

II. PRAKTICKÁ ČÁST

7 ZÁLOHOVACÍ SCRIPT

7.1 Cíl Práce

Požadavky

Dal jsem si za úkol vytvořit script v Linuxu, který by zvládl každý spustit a pomocí interaktivního menu se pokusit vytvořit zálohovací plán pro své soubory. V menu by se jednak měl pohybovat, měnit různá nastavení a parametry. Po té, co nastaví všechny potřebné údaje, script vygeneruje další soubor, který bude obsahovat kód včetně nastavených parametrů. Tento soubor podle zvolených parametrů pak může být automaticky spouštěn z crontabu, nebo si jej uživatel může spustit podle potřeby sám.

7.2 Řešení

7.2.1 Struktura aplikace

- Menu
- Výběr způsobu zálohy
- Výběr uložení zálohy
- Výběr monitorování
- Nastavení automatického zálohování
- Výběr souborů určených k záloze
- Nápověda
- Generování externího souboru s kódem

7.2.2 Popis programu

Vlastní program se sestává z menu, ve kterém se uživatel pohybuje pomocí zadávání čísel v rozsahu 1 - 9. Aby uživatel mohl vygenerovat zálohovací script, je třeba, aby nastavil většinu těchto údajů. Jako první uživatel volí mezi zálohou inkrementální, nebo zálohou plnou. V případě, že je zvolena inkrementální záloha, ověří se, zda je dostupný zdroj, ze kterého se budou provádět změny. Pokud ne, je vytvořena záloha plná.

Dále uživatel musí nastavit, zda chce provést zálohu souboru do složky na disku, nebo na specifický FTP server. V případě externího disku, tento disk musí být připojen do systému pomocí příkazu mount. Po zvolení volby se kontroluje, jestli zadaná cesta skutečně existuje, v případě FTP serveru se pak provede testovací připojení, zda jsou údaje zadané správně.

V dalším kroku uživatel nastaví, zda chce provádět monitorování zálohy. Jsou zde tři možnosti, jednak monitorování do souboru, monitorování zasílané na zadanou emailovou adresu, nebo je zde nedoporučovaná volba v případě, kdy uživatel nechce provádět monitorování zálohy. Ověření pak probíhá testování dostupnosti zadané složky, v případě emailu se provádí testování za pomoci univerzálního regulérního výrazu. Ukázka Program 2.

Třetí krok se sestává ze zvolení určité doby, ve kterou uživatel nechává svá data zálohovat. Implementace probíhá do crontabu, tudíž je potřeba zajistit potřebnou formu vstupu. A to ve tvaru: minuta hodina den z měsíce, měsíc a den z týdne, zde se opět provádí ošetření za pomoci regulérního výrazu. V této nabídce má pak uživatel možnost zrušit toto plánování v případě, že už uložil jeden ze svých modulů. Zrušení proběhne pouze v případě, kdy má uživatel vybrán a nastaven aktuální modul, který chce zrušit.

Poslední krok, který zbývá k dokončení, je výběr souborů určených k záloze. Uživatel nejprve vybere název modulu a poté přidá libovolný počet souborů, které jsou určeny k záloze. Je zde potřeba brát v potaz, že soubory se komprimují za pomoci parametru tar -j. Lze zde taktéž zadané cesty k souborům mazat.

V případě, kdy jsou nastaveny všechny tyto údaje, je možnost provést uložení. Uložení se sestává z exportování všech klíčových proměnných a samotného kódu, který vykoná zálohu do souboru. Taktéž se importuje nastavení do crontabu a celý proces je hotov. V určenou dobu se pak spustí tento soubor neboli modul.

7.2.3 Ukázky

Zde je provedeno několik ukázek programu, jednak ošetření vstupu. Vložení do crontabu, zálohování tar. Čtení ze vstupu adresářové cesty, ošetření, zda je soubor čitelný, zda lze zapisovat, kontrola ftp.

Ošetření vstupu podmínkou za pomoci regulérního výrazu

```
74             v_minuta=
75 until [ "$v_minuta" = "0" ]; do
76     echo "zadej:0-59 nebo *"
77     read -p Minuta: minuta
78     if [[ $minuta =~ ^[0-9]$|^([0-5])[0-9]$|^[*]$ ]]; then
79
80             clear
81             echo "OK"
82             v_minuta=0
83     else
84             echo "ŠPATNÝ ROZSAH"
85     fi
```

Kontrola dostupnosti FTP serveru

```
85 ftp -n << END_SCRIPT
86     open $FTPS
87     user $FTPU $FTPP
88     mkdir backup
89     cd backup
90     put ftptestfile.txt
91     get ftptestfile.txt retrieval.$$
92     bye
93 END_SCRIPT
94
95 if [ -f retrieval.$$ ]
96 then
97     echo "FUNGUJE"
98     rm -f retrieval.$$
99     text2="zaloha na FTP"
100    prom2=ftpbckp_pass
101 else
102    echo "NEFUNGUJE"
103 fi
```

Podprogram pro výběr souboru určeného k záloze

```
104 echo -n "Akce: "; read akce
105
106 case $akce in
107 0)
108   echo -n "Soubor/adresar: "; read -e v_pridat
109       if [[ -f "$v_pridat" || -d "$v_pridat" ]]; then
110           array[${#array[*]}]=$v_pridat
111       else
112           echo "Takova cesta neexistuje"
113       fi
114 ;;
115 1)
116   echo -n "Index: "; read vindex
117       if [ "$vindex" = "" ]; then
118           echo "Spante"
119       else
120           unset array[$vindex]
121           echo
122           echo "Obsah:"
123           for I in ${!array[*]}; do
124               echo $I: ${array[$I]}
125           done
126       fi
127 ;;
128 2)
129   echo
130   echo "Obsah:"
131   for I in ${!array[*]}; do
132       echo $I: ${array[$I]}
133   done
134 ;;
135 3)
136   volbic=
137 ;;
138 *)
139   echo "chyba,taková možnost tu není"
140 ;;
141 esac
```

ZÁVĚR

Tato práce shrnuje obecné, ale i konkrétní otázky, které se vztahují k vytyčení si dobrého zálohovacího systému, s důrazem na co nejmenší investici. Hlavním cílem je navrhnout co nejpřesnější zálohovací systém, který by splňoval určitá kritéria. Jednak otázky, které si musíme klást v případě, kdy požadujeme co nejúspěšněji zálohovat jakákoliv data. V první části práce je zmíněno, kdy je nejlepší provádět zálohu a jakým způsobem a také na jaké zařízení, respektive úložiště zálohu provést.

Avšak nelze přesně stanovit požadovaný optimální zálohovací postup, který by byl univerzální a vyhovoval všem. Důležité je zde pravidlo, že netřeba používat drahý zálohovací systém, ale dbát na včasné a pravidelné zálohy, aby data byla vždy aktuální a plně k dispozici v případě výpadku.

Další kapitola se věnuje již vytvořenému open-source zálohovacímu systému, který je multiplatformní co se týče operačního systému a uživatelsky ověřeného systému. Jeho nasazení je úspěšně implementováno i v komerčním sektoru.

Samotnému zálohování v operačním systému Linux je věnována další kapitola. Obsahuje několik kroků, které uživatel musí splnit, aby provedl úspěšnou zálohu a obnovu. Tyto kroky jsou dále rozděleny podle způsobu, který nejvíce vyhovuje uživateli. Může nastat situace, kdy si uživatel nemůže dovolit vynucené vypnutí operačního systému, nebo kdy potřebuje ušetřit místo, datový tok i čas.

Závěr práce patří praktické části. Obsahem je vytvoření zálohovacího scriptu, který je tvořen v bashi. Skládá se z interaktivního menu, kde si uživatel vybírá typ zálohy, čas zálohy, umístění zálohy atd. Tento script bych zhodnotil jako dostačující volba pro uživatele, kteří nemají velké zkušenosti s Linuxem. Jeho interaktivní prostředí umožní uživateli naplánovat jednoduchou inkrementální či plnou zálohu.

ZÁVĚR V ANGLIČTINĚ

This thesis sums up general but also particular questions relating to identifying a good back up system with emphasis on the least investment. The main objective is to propose the most accurate back up system which would meet certain criteria. Such criteria may be achieved by asking questions, for example how to back up, when to back up and so on, which is necessary when requiring backing up any data most successfully. The first part of this thesis answers the questions when it is most suitable to perform the backing up, how to do it and also on which device, the storage respectively, to perform the backing up.

However, it is not possible to determine required optimal backing up process that would be universal and fitting everyone. The most important is the rule that it is not necessary to use the most expensive backing up system but ensure timely and regular back ups so the data are always up to date and fully available in case of failure.

Following chapter deals with creating the open-source backing up system, which has several platforms in terms of operational system and system proved by the users. Its usage is successfully implemented even in commercial sector.

Backing up in operational system Linux is described in next chapter, containing several steps which must be done by the user, in order to perform successful backing up and its recovery. These steps are further divided according to the way suiting the user best. It is also possible that a user cannot do the forced switch-off of the operational system or he needs to save space, data stream and time.

The practical part is at the end of this thesis. It contains the creation of backing up script performed in bash. It has the interactive menu where the user chooses the type of back up, its time, placement and so on. This script can be said to be sufficient for users who have not much experience with Linux. Its interactive interface enables the user to plan simple incremental or full back up.

SEZNAM POUŽITÉ LITERATURY

- [1] PRESTON, W. Curtis. Backup and Recovery. Sebastopol : O'Reilly Media, 2007. 768 s.
- [2] KRUTÍLKOVÁ, Hana. Programové vybavení pro zálohování a obnovu souborů. Zlín, 2007. 64 s. Bakalářská práce. Univerzita Tomáš Bati ve Zlíně
- [3] VYCHODIL, Vilém. Linux: Příručka českého uživatele. Nám. 28. dubna 48, 635 00 Brno : Computer Press, 2003. 270 s.
- [4] BOLDRITO, Remo ; ESTEVE, Josep . GNU/Linux Advanced Administration [online]. [s.l.] : Free Technology Academy, 2004 [cit. 2011-02-04]. Dostupné z WWW: <<http://ii.iinfo.cz/r/k/gnu-linux-advanced-administration.pdf>>
- [5] Linux – Dokumentační projekt [online]. Nám. 28. dubna 48, 635 00 Brno : Computer Press®, 2003 [cit. 2011-02-04]. Dostupné z WWW: <http://ii.iinfo.cz/r/k/Dokumentacni_projekt.pdf>
- [6] NEWHAM, Cameron . Linux Shell Scripting with Bash. [s.l.] : Sams, 2004. 432 s.
- [7] ŠINCER, Jakub . PAMETOVÁ MÉDIA PRO ARCHIVACI DAT. Brno, 2008. 48 s. Bakalářská práce. VYSOKÉ UCENÍ TECHNICKÉ V BRNE.
- [8] WELSH , Matt. Používáme Linux . Nám. 28. dubna 48, 635 00 Brno : Computer Press®, 2003. 660 s.
- [1b] TDP-Ontrack - Zachraňujeme disky [online]. 2004 [cit. 2011-06-05]. Zachraňujeme disky. Dostupné z WWW: <<http://www.obnova-dat.com/clanky/041118a-zachranujeme-disky.shtml>>.
- [2b] M. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 2009, last modified on 2009 [cit. 2011-06-05]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/LVM>>.
- [3b] Backup. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 2004, last modified on 2011 [cit. 2011-06-05]. Dostupné z WWW: <<http://en.wikipedia.org/wiki/Backup>>.
- [4b] Linux E X P R E S [online]. 2009 [cit. 2011-06-05]. Správa linuxového serveru: RAID teoreticky. Dostupné z WWW: <<http://www.linuxexpres.cz/praxe/sprava-linuxoveho-serveru-raid-teoreticky>>.

[5b] Wiki bacula [online]. 2007 [cit. 2011-06-05]. What is bacula?. Dostupné z WWW:
<http://wiki.bacula.org/doku.php?id=bacula_manual:what_is_bacula>.

[6b] DBMS. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) :
Wikipedia Foundation, 2006, last modified on 2006 [cit. 2011-06-05]. Dostupné z WWW:
<<http://cs.wikipedia.org/wiki/DBMS>>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

NFS	Protokol pro vzdálený přístup k souborům přes počítačovou síť
SMB	Protokol navržený pro sdílení souborů a tiskáren. Je možné data z Unixu zpřístupnit i pro uživatele Windows
CIFS	Původní SMB protokol, převzatý a vylepšený firmou Microsoft. Přidává rozšíření typu (práva, symbolické odkazy apod.)
NIS	Složitý a zároveň universální systém pro zpracování údajů z databází.
DHCP	Aplikační protokol, používá se pro automatické přidělování IP adres jednotlivým PC.
IA64	64bitová architektura počítačových procesorů, implementovaná v procesorech Itanium a Itanium2.

SEZNAM OBRÁZKŮ

<i>Obrázek 1 Příčiny ztráty dat podle zdroje [1b]</i>	<i>12</i>
<i>Obrázek 2 Interakce mezi aplikacemi podle zdroje [5b]</i>	<i>27</i>
<i>Obrázek 3 Informační tok v programu Bacula podle zdroje [5b]</i>	<i>29</i>

SEZNAM TABULEK

Tabulka 1 Každý den stupeň zálohy full/level 0.....	22
Tabulka 2 Jednou za týden stupeň level 0, denní dif./level 1	22
Tabulka 3 Jednou za týden stupeň level 0, denní level 1 - 6.....	23
Tabulka 4 Jednou za týden stupeň level 0, denní TOH inkrement.	24
Tabulka 5 Měsíční plán systémem TOH.....	24

SEZNAM PŘÍLOH

Příloha P1: Zálohovací script v linuxu

PŘÍLOHA P I: ZÁLOHOVACÍ SCRIPT V LINUXU

Zde přikládám CD se zálohovacím scriptem.