Rychlokurz správy Debian serveru: Správa serveru

Vložil/a shadow [1], 28 Duben, 2009 - 13:51

- Anonymita [2]
- GNU/Linux a BSD [3]

V tomto seriálu se budeme zabývat správou linuxového serveru postaveného na distribuci Debian GNU/Linux. V prvním díle si probereme GNU/Linux jako takový, adresářovou strukturu, příkazovou řádku, dokumentaci, správu uživatelů a správu softwaru.

Na úvod musím zdůraznit, že správa linuxového serveru je komplexní záležitost, o které bylo napsáno mnoho knih, a ani v nich není obsaženo vše. Berte tento úvod do správy serveru postaveném na GNU/Linuxu pouze jako nouzový rychlokurz, kterému by mělo následovat experimentování a studium dalších materiálů ještě před tím, než se pustíte do provozu ostrého serveru.

Úvod do GNU/Linuxu

GNU/Linux je unixový operační systém. Existují dva základní principy fungování unixových systémů. V těchto systémech je u jednotlivých komponent preferována filozofie "dělej jen jednu věc, ale dělej jí dobře". V unixových systémech je tedy mnoho nástrojů, kde každý dělá jenom jednu věc (jak nejlépe to jde), a tyto nástroje je možné propojit pomocí příkazové řádky, a získat tak tu funkcionalitu, kterou požadujeme.

Druhá charakteristika unixových systémů je fakt, že vše je soubor. Tudíž i kus hardwaru je v systému reprezentován souborem (nebo i více soubory), se kterým je možné provádět běžné operace (samozřejmě dle toho, co hardware umožňuje). Pro představu, první SATA/SCSI disk je reprezentován souborem dev/sda. Pokud zkopírujeme jeho obsah do jiného souboru, získáme soubor se stejným obsahem, jaký má celý tento disk.

Adresářová struktura

Unixové systémy nepoužívají logické jednotky označené písmeny. Všechny souborové systémy se připojují do jednoho velkého adresářového stromu. Jednotlivé souborové systémy je možné připojovat, odpojovat nebo měnit jejich parametry pomocí programů mount a umount.

Adresářová struktura v unixových systémech má svůj význam i systém. Pokusím se ve stručnosti vysvětlit, co najdeme v jednotlivých adresářích. Podotýkám, že adresářová struktura začíná kořenovým adresářem, který se označuje lomítkem (pozor na odlišnost od operačních systémů od Microsoftu, kde se pro oddělení jednotlivých adresářů (resp. složek) používá zpětné lomítko, narozdíl od unixových systémů, kde má zpětné lomítko úplně jiný význam -
neguje význam speciálního znaku):

- /bin - klíčové (uživatelem) spustitelné programy pro spuštění a provoz systému
- /sbin - klíčové (uživatelem root) spustitelné programy pro spuštění a provoz systému
- /lib - klíčové knihovny a moduly jádra
- /boot - obrazy jader a konfigurace zavaděče GRUB (je-li nainstalován)
- /dev - speciální soubory reprezentující jednotlivá zařízení
- /home - domovské adresáře uživatelů (jméno adresáře odpovídá uživatelskému jménu)
- /media, /mnt - adresáře určené k~připojení dalších souborových systémů (CD/DVD, jiné

oddíly, apod.)
- /opt - místo pro instalaci softwaru, který nevyužívá unixovou hierarchii adresářů
- /proc - speciální adresář se speciálními soubory, které umožňují číst informace z~jádra a měnit jeho parametry (za běhu)
- /root - domovský adresář uživatele root
- /sys - speciální adresář podobný /proc zaměřený na hardware
- /tmp - adresář s dočasnými soubory (zde musí být pro správný provoz mnoha programů volné místo)
- /usr - druhotná hierarchie (s adresáři /bin, /lib, sbin, apod.) určená pro software, který není třeba k~základnímu fungovaní operačního systému
- /var - proměnlivá data (nalezneme tu třeba adresář se systémovými logy (/var/log) a různé citlivé adresáře, kde je taktéž třeba zajistit volné místo)

Shell

Základním stavebním kamenem unixových systémů je interpret příkazové řádky (shell). Linuxové distribuce (včetně Debianu) sice obsahují grafické prostředí, a usnadňují tak práci se systémem i běžným uživatelům, avšak na serveru (přesněji na webovém serveru) nemá grafické prostředí opodstatnění. Pro správu systému a realizaci všech potřebných činností budeme používat příkazovou řádku.

Základy práce s příkazovou řádkou (shellem) si nyní probereme. Budu předpokládat práci s výchozím shellem, který se nazývá Bash. Po přihlášení nám shell zobrazí výzvu (prompt), kam budeme zadávat jednotlivé příkazy. Tato výzva zpravidla obsahuje naše uživatelské jméno, jméno serveru a aktuální adresář. Je zakončena buď znakem $, který naznačuje, že nemáme oprávnění uživatele root, nebo znakem #, který značí, že jsme přihlášeni jako uživatel root.

Příkazy mohou mít parametry, popřípadě volby. Oddělovacím znakem je mezera. Kurzorové šipky je možné používat k pohybu po zadávaném příkazu a ten libovolně upravovat. Šipkami nahoru a dolů se pohybujeme v historii dříve zadávaných příkazů. Užitečnou funkcí při zadávání názvů souborů nebo cest k souborům je doplňování, ke kterému využíváme tabulátor.

Velmi užitečným programem pro realizaci různých činností s příkazovou řádkou je Midnight Commander, který spustíme příkazem mc.

Dokumentace

Je jasné, že v tak rozsáhlém systému, jakým je GNU/Linux, se nelze orientovat bez podrobné dokumentace. Tu členíme do několika kategorií:

- programová dokumentace, manuálové stránky
- distribuční dokumentace
- obecná dokumentace (knihy, články, weby o GNU/Linuxu)

Jelikož používáme Debian, je vhodné v první řadě doporučit studium oficiální dokumentace této distribuce. Tuto dokumentaci nalezneme zde [4]. Tato dokumentace je velmi podrobná a obsahuje mnoho informací o specifikách Debianu.

Lze také doporučit studium některé obecnější dokumentace, která třeba poskytuje podrobnější pohled na GNU/Linux a některé obecnější záležitosti, třeba práci v příkazové řádce, kterou je třeba ke správě linuxového serveru ovládat alespoň na základní úrovni.

Je však také dobré vědět, že existuje dokumentace přístupná v rámci systému jako takového. Sem řadíme manuálové stránky a doplňující programovou dokumentaci. Manuálové stránky vyvoláme příkazem man. Používání manuálových stránek je popsáno v manuálové stránce programu man. Tu lze vyvolat zapsáním příkazu:

```
man man
```

Manuálové stránky mají jednotlivé příkazy, ale také třeba konfigurační soubory. Pokud bych se rád dozvěděl více o~příkazu mount, zapíši příkaz:

```
man mount
```

Pokud bych se chtěl dozvědět více o~konfiguračním souboru \verb|fstab|, zapíši příkaz:

```
man fstab
```

Jednotlivé konfigurační soubory mnohdy samy o sobě obsahují komentáře (většinou uvozené křížkem), kde jednotlivé volby podrobně popisují.

Debian běžně obsahuje nejenom manuálové stránky, ale i další dokumentaci k jednotlivým balíčkům, a to v adresáři /usr/share/doc. Bývají zde
uloženy vzory konfiguračních souborů, soubory \verb|README| a poznámky pro
balíček od tvůrců Debianu.

Jedno z úskalí dokumentace ke GNU/Linuxu, které je nutné zmínit, je zastarávání dokumentace. Zatímco některé věci téměř nezastarávají (práce s příkazovou řádkou), jiné záležitosti se poměrně rychle mění. Zejména různé návody a postupy jsou specifické třeba výhradně pro konkrétní verzi distribuce, přičemž v následující verzi již bude třeba použít jiného postupu.

Toto se týká jak nastavení systému, tak konfigurace služeb. Příkladem může být v našem postupu konfigurace generování certifikátu pro Apache, které v jedné z předchozích verzí Debianu (Sarge) zajišťoval skript, který byl součástí distribuce Apache, avšak v aktuální verzi tento skript chybí a certifikát je nutné vygenerovat ručně.

Sám tedy spíše doporučuji preferovat dokumentaci, která je čerstvějšího data.

Uživatelé, skupiny a práva

Systém přístupových práv v unixových systémech rozeznává tři subjekty. Vlastníka souboru (příslušného uživatele), skupinu (soubor patří vždy jen do jedné skupiny, uživatel může patřit do více skupin) a ostatní (ti, kdo nejsou ani vlastníkem, ani nepatří do dané skupiny).

Existují tři hlavní přístupová práva pro každý z těchto tří subjektů, právo ke čtení, právo k zápisu a právo ke spuštění (resp. vykonání kódu). V případě adresářů přísluší právo ke čtení možnosti vypsat obsah adresáře, právo k zápisu přísluší možnosti vytvářet, upravovat a mazat soubory v daném adresáři a právo ke spuštění přísluší možnosti adresář otevřít (příkaz cd).

Existují tři speciální přístupová práva, set-UID, set-GID a tzv. sticky bit. Pro adresáře má podstatný význam sticky bit, který znemožní uživatelům mazat soubory, které jim nepatří, i když mají do příslušného adresáře přístup (toto právo je nezbytně nutné pro adresář /tmp).

V případě souborů je nejpodstatnější set UID právo, které zajistí, že program spuštěný uživatelem bude spuštěn s právy svého vlastníka. Tak je zajištěno, aby uživatel mohl realizovat činnosti, ke kterými by jinak potřeboval oprávnění uživatele root.

Přístupová práva a vlastnictví souborů a adresářů se upravují pomocí příkazů chmod a chown.

Mezi uživateli má speciální postavení uživatel root (superuživatel), pro kterého žádná omezení neplatí. Z tohoto důvodu je třeba být velice opatrný při provádění činností s právy uživatele root. Kupříkladu, snaha vymazat veškerý obsah adresáře /tmp/neco pomocí příkazu rm -rf může při

obyčejném překlepu způsobit katastrofu. Pokud bychom oddělili úvodní lomítko a zapsali:

```
rm -rf / tmp/neco
```

pak by se začal mazat rekurzivně celý adresářový strom a všechno, co je na něj připojené. Takový příkaz by spolehlivě zničil celý systém.

Správa uživatelů

K běžné správě uživatelů slouží příkazy adduser a deluser. Ty umožňují vytváření uživatelských účtů a jejich mazání. K úpravě
uživatelských účtů slouží příkaz usermod a ke změně hesla příkaz passwd. Pro vytváření uživatelů za pomoci skriptů se hodí
neinteraktivní verze příkazů pro zakládání a rušení uživatelských účtů, a sice příkazy useradd a userdel.

Poštovní schránku získá uživatel automaticky, avšak u typu schránky Maildir je třeba nejprve založit příslušnou strukturu adresářů. Toho
lze snadno docílit třeba tak, že uživateli zašleme úvodní e-mail:

```
echo "Vitejte na serveru domena.cz" | mail -s "Uvitani" uzivatel@localhost
```

Program mail je součástí balíčku mailutils, který můžeme nainstalovat příkazem:

```
aptitude install mailutils
```

Pokud chceme uživateli založit databázi a přístup k ní, musíme se obrátit na možnosti příslušné databáze (v našem případě MySQL). Databázi můžeme založit takto:

```
mysqladmin create databaze
```

Uživatele s přístupem k této databázi založíme takto:

```
echo "GRANT ALL ON uzivatel.databaze to uzivatel.localhost \
IDENTIFIED BY 'heslo';" | mysql --user=root mysql
```

Správa softwaru

Správu softwaru v GNU/Linuxu obvykle zajišťuje centrální nástroj, správce balíčků. V případě distribuce Debian je správcem balíčků Apt.

Na balíček lze nahlížet obecně jako na komponentu. Může se jednat o konkrétní program, ale také knihovnu, dokumentaci k programu, datové soubory k programu, hlavičkové soubory (ty bývají nutné pro kompilaci softwaru, v Debianu mají přídomek -dev), apod.

Balíčky se nacházejí v tzv. repositářích. To jsou umístění (ať už fyzická - CD/DVD média, nebo síťová, s~konkrétním URL), kde se kromě vlastních balíčků nachází i jejich index, který umožňuje správci balíčků udržovat přehled o obsahu jednotlivých repositářů.

Mezi balíčky existují tzv. závislostní vazby. Typicky je to závislost programu na nějaké knihovně, kterou pro svou funkci využívá. Může se ale jednat i o jinou formu závislosti, třeba balíček se spustitelnými soubory nějakého programu bude závislý na balíčku, který obsahuje jeho datové

CC-BY-SA Security-Portal.cz | secured by paranoid sense | we hack to learn

soubory.

Správce balíčků si udržuje informace o obsahu repositářů, ale také o obsahu systému (resp. o nainstalovaných balíčcích). Sám se skládá z více částí, instalaci balíčků nezajišťuje sám, nýbrž pomocí nízkoúrovňového nástroje (v Debianu je to dpkg).

Jenom správce balíčků umí řešit závislostní vazby. Příslušné nízkoúrovňové nástroje toto neumí. Je tedy vhodné pro instalaci softwaru upřednostňovat primárně správce balíčků.

Stejně tak software, který je v oficiálních repositářích Debianu, je podporován, tzn. pokud se v některém programu třeba objeví bezpečnostní chyba, bude brzy k dispozici aktualizovaný balíček.

Repositáře Debianu

Repositáře Debianu jsou tři. Hlavní repositář je main a obsahuje pouze software, který vyhovuje požadavkům Debianu (tj. svobodný software). Repositář contrib obsahuje svobodný software, který však závisí na nějakém nesvobodném, který je zpravidla v repositáři non-free.

Jednotlivé repositáře lze přidávat, ubírat a upravovat v souboru /etc/apt/sources.list.

Práce se správcem balíčků

Pro příkazovou řádku má správce balíčků Debianu nástroj aptitude. Před kterýmkoliv úkonem, který zahrnuje instalaci nebo aktualizaci softwaru je vhodné nejprve aktualizovat jeho databázi:

```
aptitude update
```

Instalaci softwaru zvládneme již na základě předchozí kapitoly:

```
aptitude install jmeno_balicku
```

Jelikož si správce balíčků udržuje podrobné informace o nainstalovaných balíčcích a jejich verzích, stejně jako o balíčcích, které jsou k dispozici v repositářích, je aktualizace systému velmi jednoduchá:

```
aptitude soft-upgrade
```

Výhodou tohoto procesu je, že tímto způsobem dojde k aktualizaci veškerého softwaru, který jsme instalovali (s výjimkou toho, který jsme neinstalovali pomocí správce balíčků).

V případě změny repositářů (např. z repositářů pro stabilní větev na repositáře z testovací větve) je třeba vyřešit závislosti poněkud agresivnějším způsobem (včetně odstranění některých balíčků). To zajišťuje následující příkaz:

```
aptitude full-upgrade
```

Správa služeb

Jak již nyní čtenář bezpochyby tuší, jednotlivé služby se konfigurují prostřednictvím editace příslušných textových souborů v adresáři \verb|/etc|. Tyto soubory bývají okomentované a mívají podrobnou dokumentaci jak v manuálových stránkách, tak na webových stránkách příslušného projektu.

Běh služeb lze ovládat pomocí skriptů v /etc/init.d. Službu lze zastavit:

```
/etc/init.d/apache2 stop
```

A opět spustit:

```
/etc/init.d/apache2 start
```

Obě akce lze provést rychle za sebou pro minimalizaci výpadku:

```
/etc/init.d/apache2 restart
```

To však ne vždy bývá dobré provádět za provozu, zejména, když to není nezbytně nutné. Třeba zrovna Apache umí znovu načíst svou konfiguraci, aniž by musel být ukončen:

```
/etc/init.d/apache2 reload
```

Před touto akcí je velmi vhodné pečlivě zkontrolovat konfigurační soubory, jestli v nich není někde chyba. Pokud ano, služba se odmítne znovu spustit
a nastane delší výpadek. Apache má nástroj, který umožní prověřit konfigurační soubory:

```
apache2ctl configtest
```

Ne všechny služby mají podobné možnosti a nástroje. V takovém případě lze experimenty v produkčním prostředí omezit na dobu, kdy server využívá co nejméně klientů (třeba v noci), nebo změny testovat na nějakém testovacím serveru.

Řešení problémů

K řešení jakéhokoliv problému je třeba nejprve získat potřebné informace. Informace o proběhlých událostech obsahují systémové logy a logy některých služeb v adresáři /var/log.

Pokud o problému informuje nějaká chybová hláška, bývá vhodné ji přímo zadat do vyhledávače. V mnoha případech tak snadno najdeme přímo řešení našeho problému.

Ne vždy bývají směrodatné informace z logů. Pak přichází na řadu diagnostické nástroje. K diagnostice problémů se sítí se hodí nástroje jako netstat, ping, host a ifconfig či nástroj ip.

Informace o procesech nám podá top, popřípadě jeho vylepšená verze htop, kterou doporučuji nainstalovat a používat.

Činnost jednotlivých programů lze velmi podobně sledovat pomocí nástrojů typu strace. Informace o používání systémových prostředků nám podá lsof.

Při "divném" chování některých programů bývá vhodné nejprve zkontrolovat, jestli je na oddílech (zejména na těch obsahujících adresáře /tmp a /var) volné místo. To lze realizovat příkazem:

```
df -h
```

Informace o jádře (zejména o proběhlých událostech) lze získat příkazem dmesg.

Máme-li dostatek informací o problému a zejména jeho příčinách, měli bychom být schopni jej

CC-BY-SA Security-Portal.cz | secured by paranoid sense | we hack to learn

vyřešit. Pokud toho schopni nejsme, ať již pro nedostatek informací či nejasnost v rámci řešení problému, nezbude než se obrátit buď do nějakého diskusního fóra či e-mailové konference, popřípadě na placenou podporu, pokud ji využíváme.

Tím bych první díl našeho rychlokurzu zakončil. V příštím díle se podíváme na bezpečnost.

Zdroj: Michal Dočekal, Webový server na platformě Debian GNU/Linux [5]

URL článku:
https://security-portal.cz/clanky/rychlokurz-spr%C3%A1vy-debian-serveru-spr%C3%A1va-serveru

Odkazy:
[1] https://security-portal.cz/users/shadow
[2] https://security-portal.cz/category/tagy/anonymita
[3] https://security-portal.cz/category/tagy/gnu/linux-bsd
[4] http://www.debian.org/doc/
[5] http://www.shadow.cz/wiki/seminarky/bp

CC-BY-SA Security-Portal.cz | secured by paranoid sense | we hack to learn