



Projekt v rámci SIPVZ:

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

LINUX

Lekce 32

Vybrané příkazy - 5

Obsah lekce:

Cíle	1
Administrace	1
Kill – 9 a ps aux	
Souhrn	
Halt, reboot, shutdown	
Startx – inicializace X Window	
Správa uživatelů	5
Useradd – tvorba nového uživatele	
Userdel – odstranění uživatelského účtu	
Passwd – update/create uživatelské heslo	
Souhrn	
Su – přepnutí uživatele	
Funkce kernelu	8
lsmod, insmod, rmmod – výpis, instalace a odstranění modulů	
depmod – ovládání závislých modulu	
modprobe- nahrání potřebných modul	

Cíle

Po skončení této lekce studenti budou schopni:

- Ovládat vybrané příkazy

Administrace

kill -9, ps aux

Přehled

Příkaz kill odešle jednu variantu signálu k procesu zadaného superuživatелеm. Příkaz kill může být vyvolán obyčejným uživatelem, ale pouze na procesy, které vlastní. Implicitně je vyslán příkaz TERM (terminate – ukončit) a ukončí cílový pid nebo id procesu. Volba -9 při použití s příkazem kill přikáže okamžité ukončení procesu, aniž by mu dala čas pro uložení aktuálně rozdělané práce. Superuživatel může zjistit pid použitím příkazu ps. Ten dovolí uživateli zobrazit informace o aktuálně běžících procesech. Volby a, u, x, dovolují superuživateli zobrazit procesy všech uživatelů, stejně jako jeho vlastních, zobrazí uživatelovo jméno, čas startu a procesy, které nejsou ovládané z terminálu. Superuživatel může použít ps aux a kill -9 společně pro identifikaci a ukončení jakéhokoliv procesu.

Scénář

Některé situace vyvolají proces, více specifický od ostatních aplikací, kdy některý program již delší dobu neodpovídá na vstupní příkazy a nedovoluje uživateli provádět jakoukoliv práci. Uživatel vadné aplikace musí kontaktovat správce systému pro pomoc. Administrátor, který má práva super uživatele, se musí dostat do systému a provést nápravu problému. Jak může administrátor uvolnit uživatelův počítač bez vypnutí či restartu systému?

Příklad

Administrátor musí nejdříve identifikovat proces, který potřebuje být opraven. To může být provedeno několika rozdílnými příkazy, ale jedním z nejvíce užitečných je příkaz ps aux. Jakmile je proces znám, jednoduše může být ukončen použitím příkazové sekvence kill -p .

Krok 1: Příkaz ps aux zobrazí všechny běžící procesy. Problémový proces v tomto příkladu je aplikace Netscape Communicator.

```
$ ps aux
```

```
USER                PID%CPU%MEMVSZRSSTTYSTATSTARTTIMECOMMAND
root                10.00.0112068?SJan140:05init [5]
root                20.00.000?SWJan140:00 [kflushd]
root                30.00.000?SWJan140:00 [update]
root                40.00.000?SWJan140:00 [kpidod]
root                50.00.000?SWJan140:11 [kswapd]
user                1825024.70.02893272?RJan22656:26 [netscape-commun]
user                209420.30.817401048pts/0S13:340:00bash
user                209420.00.6252887pts/0R13:340:00ps aux
```

Předchozí výpis není limitován tím, co bylo vypsáno. Ve skutečnosti má systém dvě až tři stránky procesů. Tento výpis může být rozvětven pro lepší identifikaci procesů. Proces, který způsobil problémy uživatele je aplikace [netscape-commun]. Všimněte si čísla id procesu (PID) 18250.

Krok 2: Příkaz kill -9 dovoluje administrátorovi systému ukončit daný proces. Proces je určen jeho id číslem PID, který je v tomto případě 18250. Pro zabití této aplikace, stačí jednoduše zadat kill -9 18250. Signál -9 okamžitě ukončí aplikaci a dovolí uživateli pokračovat v práci.

\$ kill -9 18250

Nebude zobrazen žádný výstup, ale proces, který byl vyvolán, okamžitě ukončí danou aplikaci.

Souhrn

Název příkazu:	ps
Osnova:	ps [aux]
Popis	Zobrazí status procesů, které jsou používány uživateli.
Volby:	a Vybere všechny procesy na stanici i ostatních uživatelů. u Zobrazí výstup s lepší orientací pro uživatele. x Vybere procesy mimo aktuálního terminálu.

Tabulka 32-1 – Popis příkazu ps

Název příkazu:	kill -9
Osnova:	kill -9 pid
Popis	Pošle signál zabít proces (SIGKILL) pro proces zadaný pomocí PID. Příkaz nedovolí programu dokončit činnost.
Volby:	-e Přikáže ukončení při nějaké shodě při dlouhém názvu. -i Zeptá se uživatele jestli chce skutečně proces ukončit. -q Provede příkaz, potlačí všechny chybové zprávy. -v Ukecaný mód, zobrazí informace u ukončení procesu. -V Zobrazí informace o verzi.

Tabulka 32-2 – Popis příkazu kill -9

halt, reboot, shutdown

Přehled

Příkaz shutdown je volán do /etc/init pro ukončení všech procesů systému. Schopnost provést příkaz je imitována pouze na uživatele s vyššími přístupovými právy. Volby, které mohou být použity s tímto příkazem, jsou vypsány na konci této sekce. Ovšem dvě možnosti jsou přímo směřovány na jiné příkazy. Příkazy halt a reboot jsou ekvivalentem příkazu shutdown s přepínačem -h a -r. Příkaz halt způsobí zastavení systému a jeho případné vypnutí. Příkaz reboot vyvolá restart systému poté, co byl vyvolán příkaz shutdown.

Scénář

Jádro systému používá dynamické stránkování dat do mezipaměti pro minimalizaci možných ztrát při výpadku napájení nebo jiných příhodách. Přesně díky této stavbě paměti musí být vyčištěna, když je nějaký program zastaven nebo je připraveno vypnutí systému. Perfektní situace pro tento příkaz je případ, kdy uživatel chce vypnout či restartovat systém.

Příklady

1. Uživatel může vypnout systém Linux odesláním signálu ukončení do souboru /etc/init. Příkaz shutdown dovolí uživateli ukončit všechny procesy a umožní úspěšné vypnutí. Volby, které mohou být přidány do tohoto příkazu, přinášejí mnoho rozličných úloh, jako třeba nové nastartování systému po jeho vypnutí. Příkaz shutdown může být zadán pouze uživatelem s příslušnými systémovými právy.

Krok 1: Uživatel má schopnost říci systému aby se vypnul v zadaný čas nebo po vypršení zadaného času v minutách. Uživatel chce nastavit vypnutí systému v následujících 10 minutách. Zpráva, která může být přidána do tohoto příkazu, se odešle všem přihlášeným uživatelům a informuje je o nadcházejícím vypnutí systému.

Zadáme shutdown -r +10 "This is a test." do příkazové řádky pro instruování systému pro odeslání zprávy, vyčkat deset minut a potom vyvolat vypnutí systému a poté znovu zapnout.

```
$ shutdown -r +10 "This is a test."
Broadcast message from root (tty1) Fri Jan 13 12:34:56 1943

This is a test.
The system is gong DOWN for reboot in 10 minutes !!
```

Krok 2: Sekvence vypnutí systému může být zrušena kdykoliv mezi zadáním příkazu a inicializací procesu vypnutí zmáčknutím kláves <CTRL> a <C>. Pamatujte si, že to musí být zadáno v příkazové řádce, kde byl příkaz vypnutí zadán.

```
[CTRL + C]
Shutdown cancelled.
```

2. Jak bude vypadat formát příkazu shutdown pokud potřebuje vypnout systém v 12:34 a.m.?

Krok 1: Příkaz ve formě odpovídající předchozímu zadání postrádá správné udání času vypnutí, který je ve formátu HH:MM. Čas je systémem interpretován ve dvacetičtyř hodinovém formátu, kde 01:23 je 1:23 a.m. a 4:50 p.m. je 16:50. S tímto na paměti by čas měl být zadán jako 00:34. Volba -h způsobí, že se systém zastaví po provedení vypnutí. Zadáme shutdown -h 00:34 "This is a test." do příkazové řádky ro začátek sekvence. Nic nebude na obrazovce zobrazeno, než nastane čas vypnutí.

```
$ shutdown -h 00:34 "This is a test."
Broadcast message from root (tty1) Fri Jan 13 12:34:56 1943

This is a test.
The system is gong down for systém halt NOW !!
```

Souhrn

Název příkazu:	shutdown
Osnova:	<code>shutdown [-t sec] [-krhnfFc] time[message]</code>
Popis	Inicializuje a provádí proces ukončení běžících programů, přístupu k souborovému systému a vypnutí systému.
Volby:	<ul style="list-style-type: none">-t sec Nastaví zpoždění mezi zobrazením zprávy a vypnutím systému.-k Pošle varování bez vypnutí systému.-r Systém znovu nashartuje po vypnutí.-h Zastaví systém po vypnutí.-n Vyvolí zastavení systému.-f Rychlí restart, neprovádí fsck po startu.-F Provede fsck po startu.-c Zastaví proces vypínání.time Specifikuje čas vypnutí systému (+minuty, HH:MM, nebo slovo NOW)

Tabulka 32-3 – Popis příkazu **shutdown**

startx – Inicializace X Window

Přehled

Příkaz `startx` inicializuje skript pro vyvolání prostředí X Window pro Linux. Je to konfigurovatelný skript, který je normálně spouštěn bez nějakých přidavných parametrů, protože mnoho proměnných může být konfigurováno přímo ve skriptu. Skript `startx` není samostatně stojící věc, je součástí koncového skriptu `xinit`.

Scénář

Uživatel se rozhodl, že textové rozhraní systému Linux je nudné a těžkopádné. Alternativou k tomuto monotónnímu rozhraní je grafické prostředí X Window. Jak může uživatel rozběhnout toto prostředí v souběžnosti s příkazovou řádkou?

Příklad

Krok 1: Příkaz k vyvolání prostředí X Window je jednoduše `startx`. Všechny nastavení, které budou potřeba, nejsou zadány pomocí parametrů příkazu, ale jsou nakonfigurovány samotným skriptem. Pro normální použití, nemusíme ve skriptu měnit žádné proměnné. Pro načtení GUI (graphical user interface- grafické uživatelské prostředí) zadejte `startx` do příkazové řádky.

\$ `startx`

Systém se nyní pokusí nastavit prostředí X. Výstup bude obsahovat několik odlišných formulářů obsahujících informace. Některé z nich jednoduše zobrazují status načítání ovladačů, jiné zobrazují informace o systému, další o prostředí X. Informace zde vypsané jsou docela užitečné, pokud uživatel rozhraní jen zkouší. Avšak jsou velmi podstatné, pokud uživatel bude muset vyřešit

nějaký problém s načítáním prostředí. Několik stránek výstupu by zde nemělo smysl vypisovat. Jakmile se provede startovací sekvence, X Window systém bude zapnut a na obrazovce bude zobrazeno nové uživatelské prostředí.

Krok 2: Používejte prostředí až do doby než vás omrzí. Nebo dokud vám nezačne vadit blikání obrazovky či její větší vyzařování na těchto rozlišeních či dokud se z Vás nestanou opravdový guru.

Souhrn

Název příkazu:	startx
Osnova:	startx [klient[options]][--server]options]
Popis	Přední část příkazu xinit, je více uživatelsky přívětivá než samotná X.
Volby:	S tímto příkazem se nepoužívají žádné volby ani přepínače.

Tabulka 32-4 – Popis příkazu **startx**

Správa uživatelů

useradd – Vytvoření nového uživatele

Přehled

Příkaz `useradd` (`adduser`) přidá nový uživatelský účet do systému nebo aktualizuje informace o existujícím účtu.

Scénář

Administrátor systému je často konfrontován s přidáváním, odebíráním nebo aktualizací uživatelských účtů na více uživatelském systému. Při přidávání účtu chce administrátor také zahrnout informace o uživateli, jako přihlašovací jméno, uživatelskou skupinu a shell uživatele.

Příklad

Použitím příkazu `useradd` můžeme přidat uživatele `john` do systému. V tomto případě chceme také zadat, že `john` patří do skupiny `Wolker`, jeho shell je `bash`, jeho domácí adresář je `/home/john` a jeho účet vyprší `June 6 2001`. Také použijeme příkaz `passwd` pro přiřazení hesla účtu, aby k němu nemohl přistupovat někdo jiný.

Krok 1: Následující syntaxe je použita pro přidání uživatele `john`, pokud má uživatel dostatečná práva pro přidání uživatele. Přepínače `-g`, `-s`, `-e`, `-d` jsou použity pro zadání skupiny, shellu, data vypršení a domácího adresáře.

```
$ useradd -g Wolker -s /bin/bash -e 2001-06-06 -d /home/john john
```

Krok 2: Nyní použijeme příkaz `passwd` pro přiřazení hesla.

\$ passwd john

Souhrn

Název příkazu:	useradd
Osnova:	useradd [-dsge] [user_name]
Popis	Vytváří nový uživatelský účet.
Volby:	-d home_directory Specifikuje domácí adresář nového uživatele. -s shell Specifikuje shell nového uživatele. -g initial_group Dovoluje přidat uživatele do dané skupiny. -e tyty-mm-dd Specifikuje datum vypršení uživatelského účtu.

Tabulka 32-5 – Popis příkazu useradd

userdel – Odstraňuje existující uživatele

Přehled

Příkaz userdel jednoduše odstraňuje existující uživatelské účty ze systému. Použit s přepínačem -r příkaz odstraní uživatelský účet stejně jako všechny uživatelské soubory a jeho domácí adresář.

Scénář

Přidávání, odstraňování a aktualizace uživatelských účtů je základní úlohou správce systému. Příkaz userdel je velmi užitečný v případech, kdy uživatelský účet vypršel bez obnovení či pokud byl přístup uživateli ukončen.

Příklad

Pro tento příklad použijeme výsledků z předchozí sekce, kdy jsme přiřadili uživateli john datum June 6 2001 pro vypršení jeho účtu. Informace a záznamy o jeho účtu by měli být odstraněny ze systému. V tomto případě použijeme přepínač -r pro odstranění jeho domácího adresáře i jeho souborů.

Pro odstranění účtu john ze systému stejně jako jeho souborů zadáme následující text:

\$ userdel john

Souhrn

Název příkazu:	userdel
Osnova:	userdel [-r] [user_name]
Popis	Odstraňuje existující uživatelské účty v systému.
Volby:	-r Budou odstraněny všechny uživatelské soubory i jeho domácí adresář.

Tabulka 32-6 – Popis příkazu userdel

passwd – Aktualizace / Vytvoření uživatelského hesla

Přehled

Příkaz `passwd` vytvoří nebo změní heslo pro existující uživatelský účet. Při použití s přepínačem `-u` změní pouze heslo, které vypršelo. Uživatel musí mít práva super uživatele pro spuštění příkazu `passwd` pro jiný uživatelský účet než jeho vlastní.

Scénář

Pokaždé při vytvoření nového uživatelského účtu, nebo pokud se jedná o více uživatelský systém, kde se používá vypršení hesel pro současné účty, je používána utilita `passwd`.

Příklad

Právě jsme přidali nového uživatele `john`. Následujícím krokem mu přiřadíme nové heslo. Jakmile bude zadán příkaz `passwd` bude uživatel dvakrát dotázán na heslo, jednou pro zadání a podruhé pro kontrolu.

Příkaz `passwd` musí být zadán společně s uživatelským jménem.

```
$ passwd john
Enter new password:
Re-enter password:
Password has been changed successfully!
$
```

Souhrn

Název příkazu:	passwd
Osnova:	<code>passwd [-u] [username]</code>
Popis	Vytváří nebo aktualizuje heslo pro uživatelské účty.
Volby:	<code>-u</code> Obnoví vypršené heslo.

Tabulka 32-7 – Popis příkazu `passwd`

su - - Přepnutí uživatele

Přehled

Příkaz `su` přepne uživatelské id (já) na jiného zadaného uživatele nebo implicitně na uživatele `root`. V tabulce A-57 jsou popsány další dostupné možnosti.

Scénář

Příkaz `su` přepíná z aktuálního uživatelského účtu do účtu super uživatele aniž by se musel uživatel odhlásit. Příkaz `su` může být zadán s přepínačem `-c` a názvem příkazu a ten bude spuštěn s právy superuživatele. Spuštění ovšem vyžaduje heslo super uživatele.

Jestliže si uživatel přeje přihlásit se jako root během sezení, může využít příkazu su.

do příkazové řádky zadáme su - pro přihlášení na účet super uživatele.

\$ su -

Password:

Po správném zadání hesla uživatele root, příkazová řádka bude začínat symbolem křížku (#) jestliže je uživatel v shellu.

#

Souhrn

Název příkazu:	su
Osnova:	su [-c -m -s] [-] user
Popis	Přepne uživatele a skupinu.
Volby:	- Odstraní všechny tiskové úlohy uživatele nebo všechny pokud zadá root. -c Spustí samostatný příkaz s právy uživatele root. -m Nebude resetovat proměnné prostředí. -s Spustí shell. user Specifikuje jméno uživatele.

Tabulka 32-8 – Popis příkazu su

Funkce kernelu

lsmod, insmod, rmmod – Výpis, instalace a odstranění modulů

Přehled

Příkaz lsmod vypíše nahrané moduly zatímco insmod a rmmod instalují a odebírají nahrávané moduly. Tyto příkazy mohou být provedeny pouze super uživatelem. Některé z dostupných možností těchto příkazů je detailněji popsáno v tabulkách A-58 a A-59 na konci této sekce.

Scénář

Rozdílné ovladače pro aplikace Linuxu a programů jsou sestaveny pomocí modulů. Souborový systém ufs (Unix File System) je obvykle používán a jeho modul může být nainstalován pomocí insmod.

Příklady

1. Mnoho systémů uchovává moduly v adresáři /lib/modules/ následovaným adresářem s aktuální verzí systému Linux.

Krok 1: Po vyhledání správného adresáře, zadejte lsmod do příkazové řádky pro shlédnutí modulů, které jsou aktuálně nahrané.

```
$ lsmod
Module      Size  Used by
rt18139    121881(autocefan)
```

Jediný aktuálně nahráný modul je rt18139 Jak můžeme načíst modul ufs?

Krok 2: Pro instalaci modulu ufs, zadáme do příkazové řádky insmod ufs.

```
$ insmod ufs
```

Jestliže je modul úspěšně nainstalován, vrátí se příkazová řádka. Jestliže nebyl modul nainstalován, zobrazí se chybová zpráva obsahující nevyřešené úlohy instalace. To se může stát, pokud ještě nebyl modul správně zkompileován nebo aktivován.

Krok 3: Pro výpis načtených modulů včetně nově načteného ufs, zadáme lsmod do příkazové řádky.

```
$ lsmod
Module      Size  Used by
ufs         561600 (unused)
rt18139    121881 (autoclean)
```

Moduly ufs a rt18139 jsou nyní jediné, které jsou nahrány. Jak může být ufs odstraněno?

2. Nahrané moduly mohou být odstraněny pomocí příkazu rmmmod.

Krok 1: Pro odstranění nahráného modulu ufs, zadejte rmmmod ufs.

```
$ rmmmod ufs
```

Krok 2: Prodíváme se na aktualizovaný seznam nahráných modulů opětovným zadáním lsmod.

```
$ lsmod
Module      Size  Used by
rt18139    121881(autocefan)
```

Souhrn

Název příkazu:	insmod
Osnova:	insmod [-p -s -v] module
Popis	Instaluje moduly.
Volby:	-p Zkontroluje modul, zda může být nahrán. -s Pošle výpis do syslogu, ne do terminálu. -v Ukecaný mód..

Tabulka 32-9 – Popis příkazu insmod

Název příkazu:	rmmod
Osnova:	rmmod [-r -s] module
Popis	Odstraní modul.
Volby:	-r Odstraní balík modulu. -s Pošle výpis do syslogu, ne do terminálu.

Tabulka 32-10 – Popis příkazu rmmod

depmod – Ovládá závislé moduly

Přehled

Příkaz depmod vytváří kartotéku závislostí založenou na symbolech nalezených v souborech modulů a která může být využita s příkazem modprobe pro načtení závislých modulů. depmod také zobrazit, které moduly jsou na daném modulu závislé a naopak.

Scénář

Předpokládejme, že pro hlavní program je závislý na několika dalších podprogramech. Tento příkaz nám ukáže, které moduly jsou zapotřebí a vytvoří soubor pro načtení všech těchto modulů.

Příklad

Příkaz může být použit pro vyhledání modulů, na kterých závisí modul ltmodem.o.

Zadáme depmod -e ltmodem.o.

```
$ depmod -e ltmodem.o
depmod: *** Unresolved symbols in ltmodem.o
depmod: bh_mask
depmod: schedule_timeout
depmod: request_region
depmod: pcibios_read_co
```

Zde je několik modulů potřebných pro ltmodem.o.

Souhrn

Název příkazu:	depmod
Osnova:	depmod [-a -e -v] modules
Popis	Vytvoří soubor závislostí pro moduly.
Volby:	-a Vyhledá a vytvoří seznam závislostí pro všechny moduly v konfiguračním souboru.. -e Zobrazí všechny nevyřešené označení pro každý modul -v Zobrazí každý modul jak byl zpracován.

Tabulka 32-11 – Popis příkazu depmod

Přehled

Příkaz modprobe je používán pro nahrání specifického modulu kernelu požadovaného pro nějaké zařízení, souborů a všech modulů, které jsou pro něj potřeba. Jestliže je zde několik modulů vypsaných příkazem modprobe nahraje se pouze první modul pokud, další pouze pokud první selže.

Scénář

Když instalujeme nové zařízení, souborový systém nebo nový typ softwaru, uživatel potřebuje instalovat také potřebné moduly (příslušné k zařízení nebo softwaru) pro úspěšné spuštění zařízení nebo programu. Použitím modprobe nenainstalujeme pouze vypsané moduly, ale také všechny jenž budou potřeba pro další moduly.

Příklad

Příkaz může být použit pro instalaci nezbytných modulů pro nový souborový systém jako třeba MS-DOS. Použijeme následující kroky pro nainstalování tohoto souborového systému.

Krok 1: Po přihlášení za uživatele root, zadáme modprobe pro instalaci MS-DOS souborového systému a všech potřebných modulů.

\$ modprobe msdos

Krok 2: Nyní použijeme příkaz lsmod pro shlédnutí jestli byly moduly úspěšně přidány. Zadáme příkaz samostatně pro zobrazení všech modulů nainstalovaných v systému. Všimněte si, že byl nainstalován modul fat, protože je potřebný pro modul msdos.

\$ lsmod

```
Module      #pages:Used by:
msdos       20
fat         60
serial      71 (autoclean)
```

Souhrn

Název příkazu:	Modprobe
Osnova:	Modprobe [-ar] [modules]
Popis	Nahraje zadané moduly a všechny pro ně potřebné.
Volby:	-a Když je zadán více než jeden modul, příkaz je nahraje všechny. -r Odstraní všechny zadané moduly a stejně tak závislé moduly.

Tabulka 32-12 – Popis příkazu modprobe