

Operační systémy – test č. 2, Linux

seznam možných otázek

Poslední aktualizace: 11. května 2017

V otázkách se na testu mohou objevit změny, typicky v názvech souborů, služeb nebo použitých přepínačích příkazů. Studenti mají u testu k dispozici počítač, mohou si donést vlastní notebook.

1 Úvod do správy v textovém režimu

1. Vyjmenujte alespoň dva různé textové shelly, které se obvykle používají v UNIXových systémech, každý stručně charakterizujte. Ve kterém souboru je seznam povolených shellů?
2. Zjistěte, jakým typem příkazu je příkaz `echo/passwd/cd/ping/grep/ls/...`
3. Napište alespoň tři různé způsoby, jak lze v textovém režimu vytvořit nový soubor. Co je to *here document* (vložený soubor), jak se používá?
(pozn.: here document je právě to, co jsme zkoušeli při vytváření souboru pomocí cat s určením ukončující sekvence – výstup směřujeme do souboru, na vstup „natvrdo“ – dvojitě – směřujeme to, co chceme mít jako ukončující sekvenci)
4. Jak lze vypsát seznam souborů odpovídajících určité masce bez použití příkazu `ls`? Jak na tom postupu spolupracuje shell? Tímto způsobem vypište seznam všech souborů z pracovního adresáře, které začínají buď písmenem `b` nebo písmenem `M`.
5. Jaký je významový rozdíl mezi příkazy `find`, `locate`, `grep`, `whatis` a `which`, za jakých okolností se který z nich používá?
6. Najděte všechny soubory s příponou `.txt` ve vašem domovském adresáři, vypište jejich plná jména.
7. Najděte všechny soubory s příponou `.txt` ve vašem domovském adresáři, a to rekurzivně, vypište jejich plná jména.
8. Najděte všechny soubory s příponou `.txt` ve vašem domovském adresáři, a to rekurzivně, bez rozlišování malých/velkých písmen, vypište jejich plná jména.
9. Najděte v adresáři `/var` všechny soubory, jejichž vlastníkem je uživatel s UID 1025 (příp. zadán název uživatele), vypište jejich plná jména.
10. Ve svém domovském adresáři vypište všechny soubory, které jsou adresáři.
11. Najděte soubory s příponou `*.sh`, které byly modifikovány právě před 2 dny/před více než 2 dny.
12. Najděte všechny soubory v domovském adresáři, do kterých mají právo zápisu „ostatní“.
13. Vypište seznam všech souborů (rekurzivně z kořenového adresáře), které mají nastaven SUID bit.
14. Zjistěte, kde se nachází spustitelný soubor příkazu `lsmod/ls/cat/passwd/shutdown/man/mount/...` Použijte oba příkazy, které lze k tomuto účelu využít.
Jedná se o příkazy `whereis` a `which`, takže `whereis lsmod`, `which lsmod`
15. Při vyhledávání: co je to *posix sekvence*?

16. Najděte v souboru `/etc/passwd` všechny záznamy, kde je nastaven shell `/bin/bash`.
(tj. řádky, na kterých je tento řetězec – použijte na tento soubor příslušný vyhledávací program, příp. filtr)
17. Zjistěte, ve kterém souboru vašeho domovského adresáře je řetězec `alias` (vypište pouze názvy souborů, ve kterých je tento řetězec).
18. Z výstupu příkazu `ls -la` zobrazte pouze ty soubory či adresáře, které mají pro svého vlastníka povolena všechna oprávnění `rwX`.
Pozn.: takže první znak na řádce jakýkoliv, další tři mají být `rwX`.
19. Vypište počet registrovaných uživatelů.
(nápoověda: na každém řádku souboru `/etc/passwd` je jeden uživatel, tedy stačí vypsat počet řádků tohoto souboru)
20. Vypište počet podadresářů ve svém domovském adresáři.
21. Jaký je rozdíl ve významu použití apostrofů, obrácených apostrofů a uvozovek? Kdy které používáme? K čemu slouží zpětné lomítko? U každého uveďte příklad použití. Když potřebujeme zkombinovat (vnořit) dva typy uvození, které zvolíme?
22. Když je vytvořena proměnná příkazem `prom=hodnota`, je lokální nebo je v prostředí? Jak to změním?
23. Vytvořte proměnnou `prom`, přiřaďte do ní hodnotu `abc` a exportujte ji do prostředí.
24. Vypište uživateli sdělení, že má zadat číslo (název souboru ke zpracování, atd.), a pak vstup od uživatele načtete do proměnné `prom`.
25. Napište příkaz, kterým se vypočte výraz $250 - 9 * 12$, výsledek se vypíše.
Pozn.: můžete používat kterýkoliv způsob (i příkazy `let` nebo `expr`), doporučuji dvojité závorky, to funguje všude.
26. Napište příkaz, kterým k proměnné `vysledek` přičtete číslo 130.
27. Pokud vytváříme skript, který má být interpretován shellem `bash/perl/tcsh`, jak informujeme o tom, kterým shellem má být interpretován, a jak zajistíme, že bude spustitelný?
(*Pozn.: napište řetězec, který by měl být na začátku skriptu. Pokud nevíte cestu ke spustitelnému souboru zadaného shellu, použijte příkaz `whereis`, který vám tu cestu najde. Spustitelné soubory mají „x“ v přístupových oprávněních, to je taky třeba zajistit.*)
28. Pokud píšeme skript v `bash`, jak se dostaneme k parametrům skriptu, které zadal uživatel při jeho spouštění? A co když je jich víc než 9? Jak může skript vrátit číselnou návratovou hodnotu, například při výskytu chyby? Jak se dostaneme k návratové hodnotě příkazu (programu, skriptu), který jsme ve skriptu spouštěli? (například když chceme zjistit, jestli ten program neskončil s chybou)
29. Jakým způsobem propojujeme příkazy na jednom řádku, pokud chceme, aby se prováděly běžně sekvenčně, a jak je propojíme, když si mají předávat navzájem své vstupy/výstupy?
30. Jak propojíme příkazy na jednom řádku tak, aby spuštění následujícího příkazu v pořadí bylo podmíněno výsledkem (úspěch či konec s chybou) předchozího příkazu? Jak zajistíme, aby se při tomto způsobu propojení provedl určitý příkaz při úspěchu prvního v pořadí, a jiný příkaz při neúspěchu prvního v pořadí?
31. Zajistěte, aby se za 5 minut (tj. 300 sekund) spustil příkaz vypisující řetězec „budíček“ (příkaz pro výpis seznamu procesů, příkaz, kterým se přidá do souboru `casv.log` momentální datum a čas, atd.).
(*Pozn.: jde o příkaz `sleep`, ve skriptech máte jeho použití ukázáno na příkladu u jednoduchého propojení příkazů*)
32. Zajistěte, aby se v případě, že příkaz `ls` skončí neúspěchem, vypisovala hláška „Nelze vypsat obsah adresáře“.

33. Zajistěte, aby se v případě, že příkaz `ls` skončí úspěchem, vypisala hláška „Konec výpisu“.
34. K čemu slouží symbol `&` zapsaný na konci příkazu?
35. Zapište syntaxi příkazu `if`, včetně větve prováděné při nesplnění podmínky.
36. Předpokládejme, že v parametru skriptu `$2` má být název adresáře. Příkazem `if` otestujte, zda se opravdu jedná o existující adresář; pokud ano, přesuňte se do něj, pokud ne, vytvořte ho a přesuňte se do něj.
V podmínce můžete použít buď příkaz `test`, nebo obdobu s hranatými závorkami.
37. Zapište: jestliže je domovským adresářem kořenový adresář, přesuňte se do domovského adresáře.
38. Zapište syntaxi příkazu `for`.
39. Zapište syntaxi příkazů `while` a `until`.
40. Vypište v cyklu všechny parametry skriptu, ve kterém tvoříte příkazy (každý parametr na nový řádek).
41. Jak budete postupovat, když potřebujete napsat zdrojový soubor v jazyce C, přeložit ho a spustit?
(Pozn.: pozor také na to spuštění, výsledný soubor je sice v pracovním adresáři, ale v UNIXu je třeba použít sekvenci s udáním adresáře `./soubor`, protože v proměnné `PATH` nebývá pracovní adresář)
42. Přeložte soubor `kod.c` vhodným překladačem, výsledný spustitelný soubor pojmenujte `vysledek` (bez přípony).

2 Správa procesů

1. Spusťte program, který vypisuje pravidelně aktualizovaný seznam běžících procesů. Kterou klávesou tento program ukončíte?
2. Vypište seznam *všech* procesů, které v systému běží (jednorázový výpis).
Pozn.: pozor na přepínače – buď `ps aux`, nebo `ps -ef`.
3. Vypište běžící procesy, které spustil uživatel `root/novak/...`
4. Zjistěte údaje o procesu s PID 20 – kdo je vlastníkem procesu, kdo je jeho rodičovským procesem, jak se proces nazývá.
5. Vypište údaje o procesu s názvem `acpid` (zajímá nás především jeho PID, PPID, cesta, se kterou byl spuštěn, vlastník, případně další údaje). Příp.: Vypište údaje pouze o tomto procesu, ne o jiných.
6. Zobrazte stromovou strukturu procesů.
7. Zjistěte PID procesu `init` (nebo jiného), vypíše se požadované číslo.
8. Zjistěte, které soubory má namapovány do svého paměťového prostoru proces s PID 1325 (včetně knihoven, které používá).
9. Zjistěte, kde jsou nainstalovány a kde mají manuálové stránky programy `man`, `mount`, `gcc` a `ls`.
10. Co je to skupina procesů? Co je to relace procesů? Jak jsou označována identifikační čísla skupin a relací? Jak zjistíte tato čísla pro konkrétní proces?
11. Jaký je rozdíl mezi procesem a úlohou? Jak odlišujeme číslo úlohy od čísla procesu v parametrech příkazů? Kterou klávesovou zkratkou úlohu ukončíme, kterou ji pozastavíme? Jak zajistíme, aby úloha hned po svém spuštění běžela/byla pozastavena na pozadí?
(Pozn.: uvědomte si, že úloha na pozadí nemusí být pozastavená, klidně může běžet – například různé demony, používáme přece preemptivní multitasking, záleží na tom, jestli dotyčný proces při přenosu na pozadí „má co dělat“, úloze na pozadí, která je pozastavená, můžeme například poslat signál `SIGCONT`)

12. Spusťte úlohu `cat >/dev/null` na pozadí.
13. Spusťte úlohu `yes >/dev/null` na pozadí.
14. Vypište seznam spuštěných úloh.
15. Úlohu s číslem 1, která běží na pozadí, přeneste na popředí.
16. Úlohu z popředí přeneste na pozadí.
(*pozor, klávesová zkratka, nikoliv příkaz, ten by ani nebylo možné zadat*)
17. Úloha č. 3 je pozastavená na pozadí. Zajistěte, aby pokračovala v práci (na pozadí).
18. Ukončete úlohu s číslem 1.
19. Předpokládejme, že proces s PID 6283 zamrzl. Ukončete ho.
20. Procesu s PID 8251 pošlete signál, který pozastaví jeho činnost.
21. Vypište seznam signálů, které lze poslat procesům.
22. Které příkazy slouží v Linuxu k plánování spouštění úloh? Jaký je mezi nimi rozdíl (ve funkčnosti a způsobu zadávání plánovaných úloh)? Jak se nazývají démony, které zajišťují jejich činnost?
23. Zobrazte seznam naplánovaných úloh (procesů určených ke spuštění).
Pozn.: je jedno, jestli použijete `at` nebo `cron`, u obou k tomu slouží přepínač `-l`.

3 Řízení přístupu a správa uživatelů

1. Vypište seznam právě přihlášených uživatelů.
2. Vypište seznam přihlášených uživatelů včetně informace o tom, kdy se který uživatel přihlašoval.
3. Vypište seznam přihlášených uživatelů včetně informace o tom, kdy se který uživatel přihlašoval a co má právě spuštěno.
4. Zjistěte, který příkaz má právě spuštěn uživatel `novak`.
Pozn.: použijte ten nejkratší příkaz pro dané účely – `w` s uživatelem v parametru.
5. Zjistěte, pod jakým účtem právě pracujete (jako který uživatel jste přihlášení).
6. Zjistěte své UID, GID a všechny skupiny, do kterých patříte.
7. Stručně charakterizujte obsah souborů `/etc/passwd`, `/etc/group`, `/etc/shadow`, `/etc/gshadow`. Vysvětlete význam jednotlivých částí řádku v souborech `/etc/passwd`, `/etc/group`.
Pozn.: pokud si nepamatujete položky na řádcích, můžete si klidně vypsát obsah těchto souborů.
8. Vysvětlete, k čemu slouží adresář `/etc/skel` a soubory `/etc/default/useradd` a `/etc/login.defs`.
9. Ve kterém souboru najdete seznam uživatelů, seznam skupin, hesla uživatelů, hesla skupin? Který adresář slouží jako vzorový při vytváření domovského adresáře nového uživatele?
10. Vysvětlete rozdíl mezi klasickým systémem skupin a systémem skupin UPG.
11. Vytvořte novou skupinu s názvem `studenti`.
12. Odstraňte skupinu s názvem `studenti`.
13. Jednou ze skupin, do kterých patříte, je i `admini`, ale není to právě vaše aktivní skupina. Přepněte se na tuto skupinu.
14. Zatím pracujete jako člen skupiny `users`, ale teď potřebujete pracovat jako člen skupiny `wwwadmins` – přepněte si skupinu.
15. Přejmenujte skupinu `pomocne` na skupinu `udrzba`.

16. Vytvořte nového uživatele (účet) s názvem `novak`.
17. Odstraňte uživatelský účet `novak`, jeho domovský adresář nechte.
18. Změňte název (login) účtu `novak` na `novak1`.
19. Uživatele `novak` přidejte do skupiny `ucetni`, jeho původní skupiny nechte.
20. Pokud změníte název skupiny nebo její GID, nebo skupinu odstraníte, mění se něco pro soubory, u nichž je tato skupina nastavena (přidružená)?
Pozn.: vyjádřete se pro všechny tři případy – změna názvu, změna GID, odstranění skupiny.
21. Pokud změníte jméno uživatele nebo jeho GID, nebo uživatele odstraníte, mění se něco pro soubory, u nichž je tento uživatel vlastníkem?
22. Popište postup hromadného přidávání uživatelů (v jakém formátu musí být soubor s uživateli, který příkaz používáme pro načtení uživatelů).
23. Pro účet `novak` nastavte maximální platnost hesla na 90 dnů.
24. Změňte heslo uživatele `novak`.
25. Uzamkněte/odemkněte heslo k účtu uživatele `novak`.
Pozn.: dá se provést více způsoby, je jedno, který použijete – např. s využitím `usermod` nebo `passwd`. U obou máme stejný „delší“ parametr `--lock` a `--unlock`, jednopísmenné jsou odlišné.
26. Odemkněte zamčené heslo k účtu uživatele `novak`.
27. Zajistěte, aby u účtu uživatele `novak` okamžitě vypršela platnost hesla (expirovalo), aby muselo být nastaveno při příštím přihlášení.
28. Nastavte vlastníka souboru `abc.txt` na uživatele `novak`.
29. Nastavte skupinu souboru `abc.txt` na skupinu `studenti`.
30. Nastavte přístupová oprávnění k souboru `soubor.pdf` tak, aby vlastník měl přiřazena všechna oprávnění, skupina právo čtení a ostatní žádná oprávnění (tj. absolutní tvar příkazu).
31. Pro soubor `soubor.pdf` změňte přístupová oprávnění – přidejte skupině právo zápisu a ostatním právo čtení (tj. relativní tvar příkazu).
32. Pro adresář `adresar/soubor prog` nastavte SUID bit.
33. Pro adresář `adresar/soubor prog` nastavte SGID bit.
34. Pro adresář `adresar` nastavte bit `sticky`.
35. Kdo může měnit vlastníka a přidruženou skupinu k danému souboru? Kdo může měnit přístupová oprávnění k danému souboru?
36. Vysvětlete, k čemu slouží (kdy se používá) maska přístupových oprávnění a jak se z ní vypočítávají příslušná oprávnění pro adresář a běžný soubor.
37. Zobrazte masku oprávnění v číselném a pak v symbolickém tvaru.
38. Nastavte masku oprávnění tak, aby při vytvoření nového souboru byla oprávnění k němu nastavena takto: vlastník má všechna práva, skupina právo čtení, ostatní žádné.
39. Které příkazy slouží k navýšení přístupových oprávnění, příp. přihlášení pod jiným uživatelským účtem? Jaký je mezi nimi rozdíl?
40. Získejte oprávnění `roota`, včetně jeho domovského adresáře a proměnných.
41. Co je to *sudo mechanismus* a jak funguje? Jak se konfiguruje? (napište především název souboru, ve kterém je konfigurace uložena)

42. Spustíte příkaz `mount /dev/sdb1` s přístupovými oprávněními roota.
43. Jak fungují atributy v linuxových souborových systémech – základní a rozšířené? Napište příklad alespoň jednoho základního atributu (s vysvětlením). Které příkazy slouží k zobrazení a změně základních/rozšířených atributů?
44. Na jakém principu jsou založeny POSIX ACL? Co přidávají ke standardnímu mechanismu přístupových oprávnění? Který příkaz slouží ke zjišťování nastavení ACL, a který příkaz slouží přímo k jejich nastavování?
45. Vypište ACL seznam pro soubor `.bashrc` ve vašem domovském adresáři (`/etc/fstab`, `/dev/sda`, ...).
46. Co je to PAM, jak funguje? Ve kterém adresáři jsou uloženy zásobníky PAM pro různé služby? Napište název alespoň dvou PAM modulů a stručně je charakterizujte.
47. Co je to `chroot`, jak funguje? Popište postup. Jak například může proces toto omezení obejít, jaké jsou jeho slabiny?

4 Správa zařízení

1. Vypište volné místo na diskových oddílech, které jsou k dispozici, jednotky v „lidském“ formátu.
2. Zjistěte, kolik místa na disku zabírá adresář `Documents` (Dokumenty) ve vašem domovském adresáři, jednotky v „lidském“ formátu.
3. Zjistěte, které procesy pracují s vaším domovským adresářem, s adresářem `/var/log`, se souborem `/dev/null`, ...
(*tj. které procesy mají daný adresář/soubor „otevřený“; jako parametr příkazu `lsof` prostě použijeme název souboru*)
4. Který příkaz slouží ke kontrole vadných bloků na diskovém oddílu? Napište alespoň dvě varianty pro různé souborové systémy. Který příkaz slouží k vytváření souborového systému? Opět napište alespoň dvě varianty pro různé souborové systémy.
5. K čemu slouží soubor `/etc/fstab`? Jaký je význam jednotlivých polí na řádce v tomto souboru? K čemu slouží soubor `/etc/mtab`?
6. Ve kterém souboru je seznam připojitelných souborových systémů a ve kterém je seznam připojených souborových systémů? Do kterého z nich lze zasahovat?
7. Připojte oddíl, jehož přípojný bod je `/mnt/winC`, pouze pro čtení (předpokládejme, že má záznam v `/etc/fstab`).
8. Odpojte USB flash disk, jehož přípojný bod je `/mnt/flash1`.
9. Odpojte CD disk, jehož přípojný bod je `/media/cd`.
10. Který program lze použít k vytváření diskových oddílů v Linuxu? Jak vypíšeme informace o dostupných discích?
11. Co je to hlavní a vedlejší číslo zařízení a kde tato čísla zjistíme? Co je to `udev`, jak funguje, který démon zajišťuje jeho běh?
12. Vypište podrobné informace o operační paměti – pomocí příkazu `i` výpisem obsahu jednoho dynamického souboru.
13. Zobrazte informaci o operační paměti – kolik je volné, kolik obsazené, jak velký je swap.
14. Vypište informace o využívání paměti procesem s PID 3242, včetně otevřených souborů.
(*protože `i` otevřené soubory jsou součástí paměťového prostoru procesu*)
15. Vypište seznam souborů, které má proces s PID 3726 namapovány do své paměti.

16. Vypište informace o procesoru (kromě architektury i frekvenci a další údaje).
Pozn.: jsou dvě možnosti, je jedno, kterou zvolíte.
17. Vypište podrobné informace o základní desce a komponentách k ní připojených.
18. Vypište přehled USB hubů a k nim připojených zařízení.
19. Vypište přehled USB hubů a k nim připojených zařízení, ve stromové struktuře.
20. K čemu slouží moduly jádra? Napište alespoň tři různé *druhy* modulů jádra. Jaká je přípona souborů, ve kterých jsou uloženy? Který příkaz slouží k zavádění a odstraňování modulů z jádra?
21. Vypište seznam modulů, které jsou právě zavedeny v jádře.
22. Vypište informace o modulu jádra s názvem bluetooth/ahci/video/...
23. Zaveďte do jádra modul fuse/eth0/bluetooth/nfs/...

5 Správa sítě

1. K čemu slouží soubory `/etc/hosts`, `/etc/network` a `/etc/resolv.conf`?
2. Který soubor slouží ke zrychlení DNS překladu? Ve kterém souboru najdeme adresu DNS serveru (tj. informaci, kam se má náš resolver obracet, když má provést překlad)?
3. Zjistěte IP adresu serveru `www.google.com`.
4. Zjistěte doménový název adresy `91.213.160.118`.
5. Zjistěte IPv6 adresu serveru `www.root.cz`.
6. Zjistěte adresy doménových serverů v doméně `seznam.cz`.
7. Zjistěte dostupnost serveru `www.google.com`.
8. Vypište cestu přes routery k serveru `www.google.com`.
9. Zjistěte, které porty jsou otevřené a které procesy s nimi pracují (stav spojení).
10. Zjistěte, které TCP/UDP porty jsou otevřené.
11. Zjistěte, které procesy naslouchají na síti.
12. Vypište informace o síťových rozhraních na svém počítači.
13. Vypište informace o síťových rozhraních na svém počítači, včetně statistiky.
14. Zaktivněte/zneaktivněte síťovou kartu `eth0`.
15. Zjistěte svou MAC adresu.
16. Zjistěte svou IPv4 a IPv6 adresu.
17. Vypište hlavní směrovací tabulku.
18. Vypište lokální směrovací tabulku.
19. Nastavte výchozí bránu sítě na `10.2.0.1`.
20. Vypište tabulku sousedních uzlů sítě (pro IPv4 i IPv6).

6 Nasazení systému

1. Jak se jmenuje a kde je soubor, ze kterého startuje jádro Linuxu? Kde jsou soubory zavaděče operačního systému?
2. Jak probíhá zavádění Linuxu? (stačí stručně)

3. Vypište informaci o tom, s jakými parametry nastartoval operační systém.
4. Vypište verzi systému a verzi jádra.
Pozn.: buď vypíšete soubor `/proc/version`, nebo použijete `uname -a`, obojí je možné.
5. Zjistěte, jak dlouho už váš systém běží (bez vypnutí).
Pozn.: buď vypíšete soubor `/proc/uptime`, nebo použijete příkaz `uptime`, obojí je možné, ale příkaz má „čitelnější“ výstup.
6. Co to jsou úrovně běhu a co mají společného s procesem `init` a službami? Napište obvyklý význam úrovní běhu 0–6.
7. Zjistěte momentální úroveň běhu (stavu procesu `init`).
8. Okamžitě vypněte systém.
9. Okamžitě restartujte systém.
10. Zajistěte, že za 10 minut se vypne systém.
11. Co se to služba a co je to démon? Napište názvy alespoň dvou obvyklých démonů.
12. Co je to inicializační systém? Stručně shrňte rozdíly mezi systémy `sysvinit`, `upstart` a `systemd`.
13. K čemu slouží položky v adresářích `/etc/rcx.d` (místo `x` číslo)? K čemu slouží položky v adresáři `/etc/init.d`?
14. Spusťte/zastavte službu s názvem `bluetooth/rsync/cups`.
Ve všech inicializačních systémech funguje příkaz `service`, takže třeba `service rsync start`
15. Zjistěte momentální stav služby `bluetooth/rsync/cups`.
16. Následující příkazy přiřaďte k těm inicializačním systémům `sysvinit`, `upstart` nebo `systemd`, ve kterých jsou nativně používány. `service`, `update.rc`, `chkconfig`, `initctl`, `systemctl`.
Pozn.: pozor, první z příkazů funguje ve všech inicializačních systémech.
17. Jaký je rozdíl mezi binárním a zdrojovým balíčkem, a co naopak mají společného? Co je obvykle součástí názvu souboru s balíčkem?

Jaký je rozdíl mezi zdrojovým balíčkem a „surovým“ zdrojovým kódem? Napište alespoň jeden příkaz fungující v distribucích založených na RedHat a alespoň jeden příkaz z distribucí odvozených z Debianu (stačí název příkazů) sloužící k instalaci a aktualizaci balíčků.
18. Máte k dispozici balíček `wireshark`. Nainstalujte tento balíček v textovém režimu (příkazem podle své distribuce).
19. Zobrazte informace o balíčku `wireshark` (použijte příkaz podle své distribuce).
20. Aktualizujte balíček `wireshark` (postupem podle své distribuce).
21. Ve kterém log souboru najdeme informace o průběhu startu systému? Ve kterém je průběh bootování? Ve kterém je popis činnosti systému (až po nastartování)?
22. Co je to `syslog`, k čemu slouží? Jak se jmenuje jeho démon?
23. Popište princip činnosti firewallu v Linuxu – jak se jmenuje příslušný modul jádra, jak se jmenuje jeho obslužný program, jak funguje (tabulky, chainy), vysvětlete význam alespoň dvou tabulek a dvou chainů.
24. Co je to SELinux?