

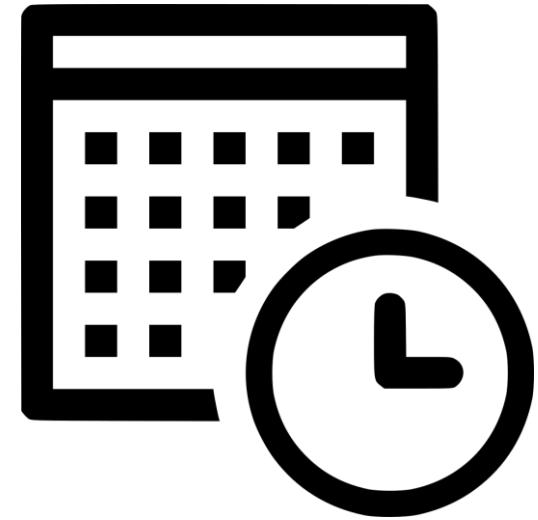
Úvod do operačných systémov

Cvičenie 8



Náplň cvičenia

- Jednoduché príkazy pre sieťové potreby
- Precvičovanie pred priebežným hodnotením



Sieťové príkazy

- Multifunkčný príkaz pre sieťové nastavenia **ip** objekt akcia
 - Nahrádza staršie príkazy ako **ifconfig** alebo **netstat**
 - Objekt **addr** (alt. **address**, **a**) pre prácu s IP adresou a jej vlastnosťami
 - Akcia **show** pre zobrazenie informácie pre všetky adresy (pri špecifikovaní **dev** **nazov** len konkrétne zariadenie)
 - Akcia **add** pre pridanie a **del** pre vymazanie IP adresy danému zariadeniu
 - Prepínače **-4** a **-6** pre špecifikovanie verzie IP
 - Objekt **link** pre správu a zobrazenie sieťových rozhraní
 - Akcia **show** pre zobrazenie informácie pre všetky rozhrania (pri špecifikovaní **dev** **nazov** len konkrétne zariadenie, prepínač **-s** pre štatistiky)
 - Akcia **set** pre nastavenie vlastností pre rozhranie (pri špecifikovaní **nazov** **up** sa zapne, **nazov** **down** sa vypne, **nazov** **mtu 9000** nastaví MTU(Maximum transmission unit) na hodnotu 9000, **nazov** **promisc on** povolí promiskuitný mód)
 - Objekt **route** pre správu a zobrazenie tabuľky smerovania (routing table)
 - Akcia **show** pre zobrazenie tabuľky (pri špecifikovaní **dev** **nazov** len konkrétne zariadenie)
 - Akcia **add** pre pridanie, **replace** pre nahradenie, resp. pridanie a **del** pre vymazanie cesty cez rozhranie (**via** **nazov/IP**)
 - Akcia **get** pre zistenie cesty pre danú adresu
 - Objekt **neigh** pre správu a zobrazenie objektov v okolí (ARP pri IPv4)
 - Akcia **show** pre zobrazenie objektov v okolí (pri špecifikovaní **dev** **nazov** len konkrétne zariadenie)
 - Akcia **add** pre pridanie, **replace** pre nahradenie, resp. pridanie a **del** pre vymazanie cesty pre zariadenie (**IP** **lladr** **dev**)
 - Objekt **help** pre všeobecnú pomocnú informáciu, prípadne akcia **help** pri konkrétnom objekte (**addr**, **link**, **neigh**, ...)

Sieťové príkazy

- Štatistická informácia o socketoch v systéme - príkaz **ss**
 - Zobrazuje sa informácia o type socketu (**Netid**), o jeho stave (**State**), o počte prijatých (**Recv-Q**) a odoslaných (**Send-Q**) packetoch a o adresách (**Local Address:Port**, **Peer Address:Port**)
 - Normálne sa zobrazia nepočúvajúce pripojenia (non-listening), je možné zobrazíť všetky (**-a**) alebo len počúvajúce (**-l**)
 - Je možné špecifikovať typ socketov, ako napríklad TCP (**-t**), UDP (**-u**), Unix (**-x**), Raw (**-w**), IPv4 (**-4**), IPv6 (**-6**)
 - Tiež je možné si pozrieť štatistiku (**-s**), filtrovať výstup podľa kritérií (**dst, src, state, or, and, ..**) alebo zobrazíť aj proces, ktorý daný socket používa (**-p**)
- Sťahovanie súborov pomocou protokolov HTTP, HTTPS, FTP, FTPS priamo v termináli - príkaz **wget**
 - Sťahuje sa zo zadanej adresy do aktuálneho adresára, pričom je možné pokračovať v prerušenom sťahovaní (**-c**), vypnúť výstup na obrazovku (**-q**), spustiť sťahovanie na pozadí (**-q**)
 - Tiež je možné zmeniť názov výstupného súboru (**-O**), stiahnuť do špecifického adresára (**-P**), sťahovať viacero súborov naraz (**-i**) alebo stiahnuť webové stránky (**-m**, pre lokálne prehliadanie je vhodné pridať **-k** a **-p**)
 - Pri FTP sťahovaní je často potrebné špecifikovať aj meno a heslo (**--ftp-user=login, --ftp-password=heslo**)
- Presun dát zo serveru alebo na server - príkaz **curl**
 - Komplexný príkaz pracujúci s webovými lokalitami, ktorý umožňuje ich prehliadanie, stiahnutie, testovanie, atď.
 - Podpora viacerých protokolov, má veľké množstvo prepínačov, nutnosť inštalácie
 - Napr. prepínače **-f**, **-s**, **-S** a **-L** znamenajú prešun chyby na návratovú hodnotu, umlč výstup z príkazu, ukázanie chyby, aj v prípade umlčaného výstupu a sleduj akékoľvek presunutie na inú adresu

Sieťové príkazy

- Pre zistenie odozvy zariadení použitých pri komunikácii z počítača do cieľa - príkaz **tracert**
 - Prvý riadok je informácia o postupnom vykonávaní (cieľ, počte zariadení, ktoré sa navštevujú a veľkosť UDP packetu), potom nasleduje informácia o jednotlivých zariadeniach (názov a IP adresa, potom čas pre jednotlivé odpovede, štandardne sú to tri odpovede), prípadne * pokiaľ dané zariadenie nevráti žiadnu odpoveď alebo uplynula maximálna doba čakania na odozvu
 - Je možné zmeniť čas čakania na odozvu (-w, reálne čísla), počet testovacích packetov (-q), počet zariadení v ceste (-m, maximálne 255), začiatkové zariadenie pre testovanie (-f), nezobrazovať hostname (-n), zmeniť port (-p) či zmeniť veľkosť packetu (zadáva sa na konci za adresu)
- Dotazovanie konkrétneho servera - príkaz **ping**
 - Príkaz potrebuje názov ale IP adresu cieľa, ktorému bude posielať testovacie ICMP packety a následne vypíše čas od poslania packetu až po prijatia odozvy a po stlačení CTRL + C sa zobrazí súhrnná štatistika o komunikácii
 - Je možné limitovať počet poslaných packetov (-c), typ internetového protokolu (IPv4 -4, IPv6 -6), časový interval medzi posielaním packetov (-i, reálne čísla), zmeniť veľkosť packetu (-s), limitovať čas posielania na sekundy (-w), prehrať zvuk, ak sa podarilo dosiahnuť cieľ (-a) alebo zaplaviť daný server množstvom packetov (-f, vyžaduje root oprávnenia, používajte s rozumom)

Sieťové príkazy

- Šifrovanie a dešifrovanie - príkaz **gpg**
 - GPG vyhovuje štandardu OpenPGP a používa PGP (Pretty Good Privacy), obsahuje viaceré zaujímavé prepínače, pričom bude uvedený len nutný základ
 - Prepínače **--dearmor** a **-o** znamenajú zbalenie alebo rozbalenie ľubovoľného vstupu do/z brnenia OpenPGP ASCII (rozšírenie GnuPG na OpenPGP a vo všeobecnosti nie je veľmi odporúčané) a ulož výstup do súboru
- Pripojenie sa na vzdialený server pomocou protokolu SSH (Secure Shell)- príkaz **ssh**
 - Je potrebné minimálne zadať IP adresu alebo názov hostiteľa, pričom pokiaľ sa nezadá používateľ (login@host), tak sa berie login aktuálne prihláseného používateľa
 - Je možné nastaviť port, ktorý je iný ako 22 (**-p**), poslať príkaz na spustenie na inom serveri (za pripojenie je potrebné špecifikovať príkaz na spustenie) alebo povoliť spúšťanie grafických aplikácií, ak to umožňujú oboje strany (**-X**)
 - Pomocou príkazu **scp** je možné kopírovať súbory na iný server alebo z iného serveru
 - V súbore **~/.ssh/known_hosts** sa nachádzajú RSA kľúče používané na pripojenie na server

Sieťové príkazy

- Pre získanie informácie DNS - príkaz **dig**
 - Príkaz potrebuje názov či názvy cieľov, ktoré hľadá a z výstupu je hlavná sekcia ANSWER
 - Výstup je možné meniť pomocou **+nocmd**, **+nocommend**, **+noedns**, **+noquestion**, **+noanswer**, **+noauthority**, **+nostats** pre vypnutie príslušnej časti výstupu, **+short** pre stručný výstup a **+noall** a následne dané časti pre zapnutie príslušných náležitostí
 - Je možné pridať vlastný DNS pomocou @, špecifikovať typ záznamu (a pre adresu, **cname** pre alias domény, **txt** pre textové záznamy, **mx** pre mail server, **ns** pre autoritatívny „name“ server, **any** pre všetky možnosti)
 - Pre opačný postup (z IP adresy získať názvy) je potrebné pridať prepínač **-x** a pomocou prepínaču **-f** je možné spustiť dávkové spracovanie pre záznamy v danom súbore a vytvorením súboru **.digrc** v domovskom adresári je možné manažovať preferencie pri zobrazovaní informácií
- Získanie DNS informácie zjednodušeným spôsobom - príkaz **host**
 - Príkaz potrebuje názov alebo IP cieľa, pre ktoré hľadá, pričom sa dokáže dostať na výstup podobný s **dig**
 - Je možné vybrať príslušný typ záznamu (**-t** a typ záznamu) či všetky záznamy (**-a**, **-v**), zobrazíť SOA (*start of authority*) záznam (**-C**), vybrať verziu protokolu IP (**-4**, **-6**), špecifikovať čas čakania (**-w**, pre nekonečné čakanie **-w**) či nastaviť počet opakovaní poslania UDP packetov na zistenie informácie (**-T**)

Príklady na precvičenie

- Vytvorte skript, ktorý bude načítavať texty z klávesnice, až kým sa nezadá slovo „end“. Skript bude pracovať v štyroch režimoch (používateľ sa prepne medzi režimami pomocou znakov a-d).
 - V režime a sa vypíšu všetky súbory a adresáre v adresári zadanom pomocou vstupného textu, ktorých názov má aspoň 5 znakov, začína na písmená a,r,t,s a v názve neobsahujú číslce
 - V režime b sa vygeneruje zadaný počet náhodných čísel z rozsahu $<1,100>$, pričom je počet zadaný pomocou vstupného textu. Následne sa vypíše minimum z týchto čísel
 - V režime c sa texty budú ukladať do skrytého súboru, pričom pri skončení skriptu sa vypíše top 10 zoznam najpoužívanějších slov a skrytý súbor sa vymaže. Pamätajte, že text nemusí byť len jedno slovo
 - V režime d sa spracuje zadaný text tak, aby neobsahoval číslce a interpunkčné znamienka a následne sa vypíše informácia, či sa jedná o palindróm (napr. radar, kde keď daný text čítame z oboch strán, dostaneme vždy to isté)

Príklady na precvičenie

- Vytvorte skript, pomocou ktorého budete simulovať vytvorenie špeciálnych útvarov, pričom voľba útvaru je špecifikovaná prvým argumentom
 - Prvý útvar bude na začiatku tvoriť 15 hviezdíčiek. Následne sa v ďalšej úrovni k počtu hviezdíčiek pridá náhodne vygenerovaná hodnota z rozsahu $\langle -2, 2 \rangle$, pričom najmenší možný počet hviezdíčiek je 1
 - Počet úrovní útvaru sa vygeneruje z rozsahu $\langle 20, 40 \rangle$
 - Druhý útvar bude na začiatku tvoriť 30 dolárov. Následne sa v ďalšej úrovni k počtu dolárov pridá náhodne vygenerovaná hodnota z rozsahu $\langle -3, 0 \rangle$
 - Počet úrovní útvaru je limitovaný počtom dolárov, ktorý nemôže byť menší ako 5
 - Tretí útvar bude na začiatku tvoriť 1 otáznik. Následne sa v ďalšej úrovni k počtu otáznikov pridá náhodne vygenerovaná hodnota z rozsahu $\langle 0, 3 \rangle$
 - Počet úrovní útvaru je limitovaný počtom otáznikov, ktorý nemôže byť väčší ako 25
- Pri výpise je potrebné, aby po každej úrovni bola prestávka 1 sekunda. Na konci sa ešte vypíše riadok pomlčiek s dĺžkou generovanou z rozsahu $\langle 10, 40 \rangle$

Príklady na precvičenie

- Vytvorte skript, ktorý bude predstavovať príkaz `mxuser`. V rámci tohto príkazu sa vypíše informácia o aktuálne prihlásenom používateľovi, čo znamená jeho login, UID, domovský adresár, používaný shell a zoznam skupín, do ktorých patrí. Tento príkaz pozná tieto štyri prepínače, ktoré je potrebné zadávať samostatne (t.j. `-aux` nie je podporované)
 - Prepínač `-c` so zadaným loginom používateľa bude fungovať v dvoch režimoch. Ak má používateľ `sudo` práva, tak sa spustí príkaz `adduser` s príslušným loginom. Ak používateľ nemá `sudo` práva, tak sa vypíše informácia, či taký používateľ v systéme existuje, alebo nie. Po skončení vráti hodnotu 0. Ak sa nezadá login, tak sa vypíše chyba „Nezadany login pre prepinač -c“ na štandardný chybový výstup a príkaz sa ukončí hodnotou 1
 - Prepínač `-g` umožní aktuálnemu používateľovi nastaviť zadanú skupinu ako predvolenú, za predpokladu, že používateľ má `sudo` práva. Po skončení vráti hodnotu 0. Ak sa nezadá existujúca skupina, tak sa vytvorí. Ak používateľ nemá práva `sudo`, tak sa vypíše chyba „Nedostatočne prava pre prepinač -g“ na štandardný chybový výstup a príkaz sa ukončí hodnotou 5
 - Prepínač `-f` vypíše obsah domovského súboru daného používateľa, pričom pri súboroch sa zobrazí ich typ a práva a pri adresároch sa zobrazí ich obsah, pričom sa už hlbšie nepostupuje. Po skončení vráti hodnotu 0.
 - Prepínač `-h` zobrazí pomocnú informáciu o použití príkazu `mxuser`, pričom sa vypíše aj informácia o jednotlivých prepínačoch. Po skončení sa vráti hodnota 3

Príklady na precvičenie

- Vytvorte skript, v ktorom sa bude spracovávať .csv súbor s 100000 ľuďmi (people-100000.csv, odkaz: <https://github.com/datablist/sample-csv-files>), pričom je potrebné vykonať tieto úlohy:
 - Vypíšte počet mužov ako aj počet žien samostatne, ktorých meno začína na písmená A až S a ich priezvisko má aspoň 5 znakov
 - Vypíšte všetky koncovky mailových adries (.com, .org, ...) v súbore, pričom k nim pridajte aj počet výskytov daných domén v súbore
 - Vypíšte všetkých ľudí, ktorý sa narodili pred rokom 1980, pričom vypíšte meno, priezvisko, telefónne číslo a ich vek (približná hodnota bude stačiť)
 - Vypíšte prvých 1000 záznamov a posledných 500 záznamov, následne vyberte všetky mená, pričom potlačte duplicity a usporiadajte mená od A po Z. Tento výpis uložte do súboru names.txt, ktorý následne skomprimujte pomocou príkazu xz

Príklady na precvičenie

- Vytvorte skript, pomocou ktorého budete simulovať hru myslím si číslo, pričom hra bude fungovať v dvoch režimoch
 - V prvom režime bude používateľ mať za úlohu uhádnuť číslo
 - Číslo sa vygeneruje z rozsahu $<1,100>$. Používateľ bude hodnoty zadávať počas vykonávania skriptu, pričom po zadaní hodnoty sa vypíše informácia menej ak je zadaná hodnota väčšia ako vygenerované číslo a viac ak je zadaná hodnota menšia ako vygenerované číslo
 - Ak používateľ uhádne číslo do štyroch pokusov, tak sa vypíše informácia „Good job, you are unstoppable!“
 - Inak pokiaľ používateľ uhádne číslo do ôsmich pokusov, tak sa vypíše informácia „Congrats!“
 - Pokiaľ používateľ neuhádne číslo do ôsmich pokusov, tak sa mu vypíše informácia „Game over, you lost!“ pomocou príkazu yes.
 - V druhom režime bude mať za úlohu počítač uhádnuť číslo
 - Číslo zadá používateľ na začiatku, pričom musí byť z rozsahu $<1,100>$. Pokiaľ číslo nie je z daného rozsahu, skript skončí hodnotou 1 a na štandardný chybový výstup sa vypíše informácia „Does not compute!“
 - Následne sa počítač bude snažiť uhádnuť dané číslo, pričom pokiaľ mu to bude trvať viac ako 6 tipov, tak sa vypíše na obrazovku „A WINNER IS YOU!“
 - Ak na uhádnutie bude potrebovať maximálne 6 tipov, tak sa vypíše informácia „All your base are belong to us“ pomocou príkazu yes

Ďalšie úlohy na precvičenie

- Ďalšie úlohy na precvičenie nájdete v súbore `priklady_na_precvicenie.pdf`
- Podmienky praktickej časti priebežného hodnotenia písanej na cvičeniach budú špecifikované na MS Teams