

Síť a komunikace

SÍŤ A KOMUNIKACE	
hostname	vypíše jméno počítače, -s krátké jméno počítače (bez názvu domény), -I všechny IP adresy počítače, -d jméno DNS domény, -y jméno NIS domény
hostname <jmeno_pocitace>	nastaví jméno počítače (trvalé nastavení se provede v <i>/etc/sysconfig/network</i>)
hostnamectl (platí od RHEL 7)	[status] vypíše jméno počítače, typ hardwaru, ID počítače, boot ID, jméno operačního systému, verzi jádra a architekturu procesoru, hostname vypíše pouze jméno počítače
hostnamectl set-hostname hostname <jmeno_pocitace> (platí od RHEL 7)	nastaví trvalé jméno počítače (upraví <i>/etc/hostname</i>), --transient nastaví dočasné jméno počítače
domainname	vypíše jméno NIS domény počítače, -I všechny IP adresy počítače, -d jméno DNS domény
domainname <nisdomena>	nastaví jméno NIS domény počítače (trvalé nastavení se provede v <i>/etc/sysconfig/network</i>)
dnsdomainname domainname -d	vypíše jméno DNS domény počítače
hostid	vypíše číselný identifikátor počítače
cat /etc/services	vypíše známé síťové služby včetně jejich protokolu a čísla portu
cat /etc/protocols	vypíše známé síťové protokoly
cat /etc/resolv.conf	vypíše seznam dostupných DNS serverů pro překlad doménových jmen na IP adresy
cat /etc/hosts	vypíše seznam IP adres a k nim přiřazených jmen počítačů, včetně jejich aliasů, používaných k lokálnímu překladu jmen počítačů na odpovídající IP adresy, aniž by bylo nutné se dotazovat na DNS server
host [<IP_adresa jmeno_pocitace>] [<DNS_server>]	vypíše jméno nebo IP adresu vzdáleného počítače a případně i IP adresu použitého DNS serveru, -t <typ> určí typ dotazu (implicitně „A“, „AAAA“ a „MX“), -v podrobný výpis; pro vyhledávání DNS záznamů lze použít zadaný DNS server místo serverů uvedených v <i>/etc/resolv.conf</i> \$ host prompt.cz (vypíše IPv4 a IPv6 adresu vzdáleného počítače a jméno poštovního serveru domény)
nslookup [<IP_adresa jmeno_pocitace>] [<DNS_server>]	vypíše jméno nebo IP adresu vzdáleného počítače a případně i IP adresu použitého DNS serveru, -query=<typ> určí typ dotazu (implicitně „A“ a „AAAA“); pro vyhledávání DNS záznamů lze použít zadaný DNS server místo serverů uvedených v <i>/etc/resolv.conf</i> ; bez argumentu se spustí v interaktivním režimu
dig [<jmeno_pocitace>] [@<DNS_server>]	vypíše IP adresu vzdáleného počítače a IP adresu použitého DNS serveru, -x <IP_adresa> vypíše jméno vzdáleného počítače, -t <typ> určí typ dotazu (implicitně „A“); pro vyhledávání DNS záznamů lze použít zadaný DNS server místo serverů uvedených v <i>/etc/resolv.conf</i>
whois [<jmeno_domeny>]	zobrazí informace o registraci internetové domény (jméno domény, organizaci, která doménu zaregistrovala, datum registrace, datum vypršení platnosti, vlastníka a kontakty) \$ whois prompt.cz (zobrazí informace o registraci internetové domény)
arp [<IP_adresa jmeno_pocitace>]	zobrazí záznamy v ARP tabulce (mapování IP adres na jejich odpovídající MAC adresy pro síťová zařízení, se kterými systém nedávno komunikoval v rámci stejné podsítě), -n nepřeloží jména počítačů pomocí DNS, -i <zarizeni> určí síťové rozhraní, -s <IP_adresa> <MAC_adresa> přidá záznam do ARP tabulky, -d <IP_adresa> odstraní záznam z ARP tabulky
ping [<IP_adresa jmeno_pocitace>]	zjistí dostupnost počítače v síti odesláním paketů „ICMP echo request“ a přijetím paketů „ICMP echo reply“, přičemž vypíše počet odeslaných, přijatých a ztracených paketů spolu s dobou odezvy pro každý paket, -b použije broadcast adresu, -c <n> určí počet paketů, které mají být odeslány cílovému hostiteli, -i <n> určí interval mezi odesláním každého paketu v řádu sekund (implicitně 1 s), -I <zarizeni> určí síťové rozhraní pro odeslání paketů, -n nepřeloží jména počítačů pomocí DNS, -W <n> určí časový limit pro odpověď v řádu sekund \$ ping -c 5 prompt.cz (zjistí dostupnost počítače v síti odesláním pěti „ICMP echo request“ paketů)

SÍŤ A KOMUNIKACE	
route [<prikaz>] [<cil>] [<specifikace ...>]	zobrazí IP směrovací tabulku jádra, -n nepřeloží jména počítačů pomocí DNS, add přidá statickou trasu, del odstraní statickou trasu, -net určí síť, -host určí počítač, netmask určí masku sítě, gw určí bránu (gateway), dev určí síťové rozhraní; (trvalé nastavení se provede v <i>/etc/sysconfig/network-scripts/route-*</i>) # route add default gw 192.168.1.1 (přidá výchozí trasu přes bránu „192.168.1.1“) # route add -host 192.168.100.10 gw 192.168.100.1 (přidá trasu k počítači „192.168.100.10“ přes bránu „192.168.100.1“) # route add -net 192.168.100.0/24 gw 192.168.100.1 dev eth1 (přidá trasu do sítě „192.168.100.0/24“ přes bránu „192.168.100.1“ a síťové rozhraní „eth1“) # route del -net 192.168.100.0/24 gw 192.168.100.1 dev eth1 (odstraní trasu do sítě „192.168.100.0/24“ přes bránu „192.168.100.1“ a síťové rozhraní „eth1“)
traceroute [<IP_adresa jmeno_pocitace>]	vypíše cestu po síti k vzdálenému počítači, -m <n> určí max. počet skoků – max. hodnotu „time-to-live“ (implicitně 30), -n nepřeloží jména počítačů pomocí DNS, -i <zarizeni> určí síťové rozhraní pro odesílání paketů, -w <n> určí časový limit pro odpověď v řádu sekund (implicitně 5 s) \$ traceroute prompt.cz
tracpath [<IP_adresa jmeno_pocitace>]	vypíše cestu po síti k vzdálenému počítači, -m <n> určí max. počet skoků – max. hodnotu „time-to-live“ (implicitně 30), -n nepřeloží jména počítačů pomocí DNS, -b vypíše jména počítačů i jejich IP adresy
mtr [<IP_adresa jmeno_pocitace>]	zobrazí interaktivním a dynamickým způsobem cestu po síti k vzdálenému počítači, včetně procenta ztráty paketů, počtu odeslaných paketů a doby odezvy pro každý skok, -m <n> určí max. počet skoků – max. hodnotu „time-to-live“ (implicitně 30), -n nepřeloží jména počítačů pomocí DNS, -I <zarizeni> určí síťové rozhraní pro odesílání paketů; interaktivní volby: p pozastaví aktuální zobrazení, q ukončí program
ip [<objekt>] [<prikaz>] [<cil>] [<specifikace ...>]	zobrazí či nastaví síťové parametry daného objektu, addr { add del show } přidá, odstraní či zobrazí IP adresy síťového rozhraní, link { add del set show } přidá, odstraní, nastaví či zobrazí vlastnosti síťového rozhraní, neigh { add del change show } přidá, odstraní, změní či zobrazí záznamy v ARP tabulce, route { add del change show } přidá, odstraní, změní či zobrazí záznamy ve směrovací tabulce, -s zobrazí statistiku zatížení síťového rozhraní; (trvalé nastavení se provede v <i>/etc/sysconfig/network-scripts/</i>) \$ ip link show (vypíše vlastnosti všech síťových rozhraní – jejich stav, MAC adresu a další síťové parametry) \$ ip -s link show enp3s0 (vypíše vlastnosti daného síťového rozhraní, včetně informací o počtu odeslaných a přijatých paketů a bytů) # ip link set down enp1s6 (deaktivuje síťové rozhraní) \$ ip addr show (vypíše vlastnosti všech síťových rozhraní – jejich stav, MAC adresu, IP adresu, masku sítě a další síťové parametry) # ip addr add 192.168.0.100/24 dev eth0 (přidělí síťovému rozhraní IP adresu) # ip addr del 192.168.0.100/24 dev eth0 (odebere síťovému rozhraní IP adresu) \$ ip neigh show (zobrazí ARP tabulku) \$ ip route show (zobrazí IP směrovací tabulku jádra) # ip route add default via 192.168.1.1 (přidá výchozí trasu přes bránu „192.168.1.1“) # ip route add 192.168.100.10 via 192.168.100.1 (přidá trasu k počítači „192.168.100.10“ přes bránu „192.168.100.1“) # ip route add 192.168.100.0/24 via 192.168.100.1 dev eth1 (přidá trasu do sítě „192.168.100.0/24“ přes bránu „192.168.100.1“ a síťové rozhraní „eth1“) # ip route del 192.168.100.0/24 via 192.168.100.1 dev eth1 (odstraní trasu do sítě „192.168.100.0/24“ přes bránu „192.168.100.1“ a síťové rozhraní „eth1“)
ifconfig [<zarizeni>]	vypíše vlastnosti všech aktivních či zadaných síťových rozhraní – jejich stav, MAC adresu, IP adresu, masku sítě a další síťové parametry, -a vypíše i neaktivní síťové rozhraní

SÍŤ A KOMUNIKACE	
ifconfig <zarizeni> <specifikace ...>	nastaví vlastnosti daných síťových rozhraní, up aktivuje zařízení, down deaktivuje zařízení, hw ether <MAC_adresa> určí MAC adresu, netmask <maska_site> určí masku sítě, mtu <n> určí maximální přenosovou jednotku; (trvalé nastavení se provede v <code>/etc/sysconfig/network-scripts/ifcfg-*</code>) # ifconfig eth0 down; ifconfig eth0 up (deaktivuje a aktivuje síťové rozhraní) # ifconfig eth0 192.168.0.10 netmask 255.255.255.0 (nastaví síťovému rozhraní statickou IP adresu a masku sítě) # ifconfig eth0 hw ether 00:11:09:D6:DC:3C (nastaví síťovému rozhraní danou MAC adresu)
ifup <zarizeni>	aktivuje síťové rozhraní # ifup eth0
ifdown <zarizeni>	deaktivuje síťové rozhraní # ifdown eth1
iwconfig [<zarizeni>] [<specifikace ...>]	nastaví či vypíše vlastnosti bezdrátového síťového rozhraní, essid <jmeno_site> určí jméno sítě, ap <AP_adresa> určí adresu přístupového bodu, ke kterému se má rozhraní připojit, mode <rezim> určí provozní režim zařízení („Managed“ = klient, „Master“ = access point), key <klic> určí šifrovací klíč # iwconfig eth1 essid AP_profik ap 00:60:1D:01:23:45 key 0123-4567-89 mode Managed (nastaví bezdrátové rozhraní „eth1“ pro připojení k síti „AP_profik“, s adresou přístupového bodu „00:60:1D:01:23:45“, pomocí šifrovacího klíče „0123-4567-89“ pro zabezpečené připojení, a v provozním režimu „Managed“)
iwlist [<zarizeni>] [<parametr>]	vypíše podrobné informace o bezdrátových síťových rozhraních a sítích, scan vypíše dostupné sítě, včetně adresy AP, frekvence, režimu, šifrování a kvality
nmcli [<objekt> <prikaz> [<argument ...>]]	vypíše informace o síťových zařízeních a konfiguraci sítě, con { add del show mod reload up down } vytvoří, smaže, zobrazí, upraví, načte, aktivuje či deaktivuje profil síťového připojení, dev { con dis status show wifi } připojí, odpojí, zobrazí stav síťového zařízení či vypíše seznam dostupných přístupových bodů Wi-Fi, radio wifi [on off] zobrazí či nastaví stav Wi-Fi; profil síťového připojení je kolekce nastavení, která lze konfigurovat pro dané zařízení, přičemž každý profil má jméno nebo ID, které ho identifikuje \$ nmcli dev status (zobrazí stav všech síťových zařízení) \$ nmcli dev show enp3s0 (zobrazí podrobné informace o daném síťovém zařízení) \$ nmcli con show (zobrazí všechny profily síťového připojení) \$ nmcli con show --active (zobrazí pouze aktivní profily síťového připojení) \$ nmcli con show enp3s0 (zobrazí podrobné informace o daném profilu síťového připojení) # nmcli con add con-name static ifname enp3s0 type ethernet ipv4.method manual ipv4.address 192.168.15.105/24 ipv4.gateway 192.168.15.1 ipv4.dns 192.168.15.1 (vytvoří nový profil síťového připojení „static“ s danou IP adresou, maskou sítě, výchozí bránou a DNS) # nmcli con mod static +ipv4.address 192.168.15.106/24 (upraví profil síťového připojení přidáním další IP adresy) # nmcli con mod static +ipv4.routes 192.168.15.0/24 10.10.10.1 (upraví profil síťového připojení přidáním další statické trasy) # nmcli con up static (aktivuje profil síťového připojení) # nmcli con mod static connection.id primary (přejmenuje profil síťového připojení na „primary“) # nmcli con mod enp3s0 autoconnect no (zakáže automatické spuštění původního profilu síťového připojení při startu systému) # nmcli con reload (načte změny v konfiguračních souborech) # nmcli con down enp3s0 (deaktivuje profil síťového připojení) # nmcli con del enp3s0 (odstraní profil síťového připojení) # nmcli dev dis enp3s0 (odpojí dané síťové zařízení) \$ nmcli radio wifi on (povolí Wi-Fi připojení) \$ nmcli dev wifi list (zobrazí seznam dostupných sítí Wi-Fi) \$ nmcli dev wifi connect WiFi01 (připojí se k síti Wi-Fi určené daným SSID)

SÍŤ A KOMUNIKACE	
whatmask [<maska_site IP_adresa/maska_site>]	vypíše počet použitelných IP adres v dané síti; je-li uvedena s maskou sítě i IP adresa, vypíše také adresu sítě, broadcast adresu a první a poslední použitelnou IP adresu \$ whatmask /24 \$ whatmask 255.255.255.0 (vypíše počet použitelných IP adres v síti) \$ whatmask 192.168.15.100/24 \$ whatmask 192.168.15.100/255.255.255.0 (vypíše počet použitelných IP adres v síti, včetně adresy sítě, broadcast adresy a první a poslední použitelné IP adresy)
ethtool [<zarizeni>]	vypíše nastavení síťové karty, -S zobrazí statistiku síťového provozu, -s <zarizeni> <parametr ...> změní nastavení síťové karty - duplex {full half} nastaví plný nebo poloviční duplexní režim, speed <n> nastaví rychlost v Mb/s # ethtool -s eth0 duplex full speed 100 (nastaví síťovou kartu pro provoz v plně duplexním režimu o rychlosti 100 Mb/s)
ifstat [<zarizeni>]	zobrazí statistiku síťového provozu - velikost přijatých a odeslaných dat na všech či daných síťových rozhraních (vypíše pouze rozdíl od předchozího zobrazení)
iftop	zobrazí interaktivním a dynamickým způsobem síťový provoz mezi vzdálenými počítači - zdrojovou a cílovou adresu, rychlost, jakou byla data odesílána a přijímána v předchozích 2, 10 a 40 sekundových intervalech a celkový souhrn, -i <zarizeni> určí síťové rozhraní (implicitně první v pořadí); interaktivní volby: n nepřeloží jména počítačů pomocí DNS, p zobrazí porty, N nepřeloží čísla portů na názvy služeb, P pozastaví aktuální zobrazení, q ukončí program
tcpdump [<vyraz>]	zobrazí provoz v síti, -i <zarizeni> na daném rozhraní (implicitně první v pořadí), port <port> na daném portu, tcp udp icmp pro daný protokol, host <počítač> mezi daným počítačem, ether host <MAC_adresa> mezi danou MAC adresou, -n nepřeloží jména počítačů pomocí DNS, -r <soubor> čte pakety ze souboru, -w <soubor> zapisuje pakety do souboru, -X zobrazí data každého paketu v hexadecimálním a ASCII formátu, -v podrobný výpis # tcpdump -i eth0 -nv port 22 (zobrazí síťový provoz na zařízení „eth0“ a portu 22) # tcpdump -nv ether host 00:02:3F:09:FA:F1 (zobrazí síťový provoz na zařízení s danou MAC adresou) # tcpdump -X host prompt.cz (zobrazí síťový provoz mezi daným počítačem v hexadecimálním a ASCII formátu)
netstat ss	zobrazí seznam otevřených soketů, včetně použitých protokolů, lokálních a vzdálených adres a stavů připojení, -a všechna současná připojení, -l naslouchající porty, -t TCP porty (společně s volbou „-a“ nebo „-l“), -u UDP porty (společně s volbou „-a“ nebo „-l“), -e uživatele a i-uzly, -p PID a jméno programu spojeného s každým připojením, -i tabulku síťových rozhraní (platí pouze pro příkaz „netstat“), -r IP směrovací tabulku jádra (platí pouze pro příkaz „netstat“), -s souhrnné statistiky pro každý protokol, -n zobrazí čísla portů místo názvů služeb (a IP adresy počítačů a UID uživatelů místo jejich jmen při použití příkazu „netstat“) # netstat -tupan (vypíše aktivní TCP a UDP síťová připojení, včetně naslouchajících portů a programů, které jsou s nimi spojeny)
ncat [<hostitel>] [<port>]	čte a zapisuje data napříč sítěmi, -e <příkaz> provede daný příkaz, -l <port> začne poslouchat na daném portu, -n nepřeloží jména počítačů pomocí DNS, -u použije UDP spojení místo výchozího TCP, -z skenuje otevřené porty bez odesílání dat, -v podrobný výpis \$ ncat -zv prompt.cz 80 (skenuje dostupnost portu 80 na vzdáleném hostiteli) \$ ncat -l 1234 -e /bin/bash (začne poslouchat na portu 1234 a otevře shell) \$ ncat 192.168.124.80 1234 (připojí se k počítači na portu 1234 s možností vzdáleného spouštění příkazů) \$ ncat -l 1234 > data.txt (začne poslouchat na portu 1234 a zapisuje přijatá data do souboru) \$ cat data.txt ncat 192.168.124.80 1234 (přeneše data na vzdálený počítač na portu 1234)

SÍŤ A KOMUNIKACE	
nmap [<sken>] [<hostitel ...>]	skenuje dostupnost portů na počítači s cílem identifikovat běžící služby, -sS provede TCP SYN sken (nejpoužívanější, nedojde k úplnému navázání TCP spojení, je odeslán paket SYN a přijat SYN/ACK – port otevřen nebo RST – port zavřen), -sT provede TCP connect sken (dojde k úplnému navázání TCP spojení), -sU provede UDP sken, -sn pouze zjistí dostupnost počítače v síti, v lokální síti zobrazí i jeho MAC adresu, -Pn nezjišťuje dostupnost počítače v síti, -O zjistí typ operačního systému, -sV zjistí verzi služby, -D <IP_adresa>[,<IP_adresa>...] určí seznam podstrčených počítačů, aby to vypadalo, že cílový hostitel je kromě skutečné zdrojové IP adresy skenován více systémy současně, -iL <soubor> čte cílové hostitele ze souboru, -n nepřeloží jména počítačů pomocí DNS, -p <port> určí rozsah portů (implicitně 1000 nejběžnějších portů), „-“ určí všech 65535 portů, -v podrobný výpis <pre>\$ nmap prompt.cz (skenuje nejběžnější TCP porty na cílovém hostiteli) # nmap -p - -sS -sU localhost (skenuje všechny TCP a UDP porty na lokálním hostiteli) # nmap -sn 192.168.15.0/24 (vypíše všechny dostupné počítače v síti) # nmap -sS -sV 147.229.28.4 > scan.txt (spustí TCP SYN sken s detekcí verze služby a výsledek uloží do daného souboru) # nmap -sS -Pn -p 1-1023 192.168.0.247 (spustí TCP SYN sken bez požadavku na ping a určí rozsah portů) # nmap -sS -sU -iL hosts (spustí TCP SYN a UDP sken na hostitelích uvedených v daném souboru) # nmap -sS -O -D 192.168.0.1,192.168.0.2 192.168.0.3 (spustí TCP SYN sken s detekcí OS na „192.168.0.3“ předstíraný z daných IP adres)</pre>
service iptables <prikaz>	start spustí firewall, stop zastaví firewall, restart restartuje firewall, status vypíše nastavení firewallu, save uloží nově vytvořená pravidla firewallu do <code>/etc/sysconfig/iptables</code>, aby byla zachována i po restartu systému
iptables-save	exportuje nastavená (i ještě neuložená) pravidla firewallu z paměti na STDOUT <pre># iptables-save > iprules</pre> (uloží nová pravidla firewallu do daného souboru)
iptables-restore	importuje pravidla firewallu ze STDIN do paměti <pre># iptables-restore < iprules</pre> (načte pravidla firewallu z daného souboru)

SÍŤ A KOMUNIKACE	
	nastaví pravidla firewallu, která řídí příchozí a odchozí síťový provoz, -t <tabulka> určí tabulku, kde má být pravidlo aplikováno; k filtrování paketů slouží tabulka „filter“ (implicitní), jež obsahuje vestavěné řetězce „INPUT“ pro příchozí pakety určené pro lokální systém, „OUTPUT“ pro odchozí pakety vytvářené lokálním systémem a „FORWARD“ pro pakety, které systémem procházejí; pro překlad IP adres veřejné sítě a přesměrování portů slouží tabulka „nat“ s vestavěnými řetězci „PREROUTING“ pro úpravu příchozích paketů, jakmile dorazí, „INPUT“ pro úpravu příchozích paketů určených pro lokální systém, „OUTPUT“ pro úpravu paketů vytvářených lokálním systémem před směrováním a „POSTROUTING“ pro úpravu odchozích paketů, když opouští systém; tabulka „mangle“ se používá pro speciální úpravy paketů a obsahuje všechny výše uvedené vestavěné řetězce; tabulka „raw“ se používá pro konfiguraci výjimek ze sledování připojení a poskytuje vestavěné řetězce „PREROUTING“ a „OUTPUT“; a tabulka „security“ se používá pro síťová pravidla Mandatory Access Control (MAC) s vestavěnými řetězci „INPUT“, „OUTPUT“ a „FORWARD“, -I <retezec> [<cislo_pravidla>] přidá pravidlo na začátek výčtu daného řetězce či na uvedenou pozici, -A <retezec> přidá pravidlo na konec výčtu daného řetězce, -D <retezec> <cislo_pravidla> odstraní pravidlo z daného řetězce, -L [<retezec>] vypíše pravidla daného řetězce, jinak všechna; s volbou -n zobrazí IP adresy a porty v číselném formátu, s volbou -v vypíše počet paketů a bytů u každého pravidla včetně protokolu a rozhraní, s volbou --line-numbers čísluje pravidla daného řetězce (vhodné pro další použití s volbou „-l“ či „-D“), -F [<retezec>] odstraní pravidla z daného řetězce, jinak všechna, -P <retezec> <cíl> nastaví výchozí politiku řetězce (implicitně je vše povoleno), -N <retezec> vytvoří uživatelsky definovaný řetězec, obvykle sloužící pro podrobnější specifikaci pravidel (u těchto řetězců nelze nastavit výchozí politiku), -X <retezec> odstraní uživatelsky definovaný řetězec; specifikace pravidla zahrnuje: -i <zarizeni> vstupní zařízení, -o <zarizeni> výstupní zařízení, -s <adresa> zdrojová adresa, -d <adresa> cílová adresa, -p <protokol> typ protokolu, -m <modul> rozšíření pravidla (state --state <typ_spojeni> určí typ spojení – NEW nové spojení, ESTABLISHED existující spojení, RELATED nové spojení navazující na již existující komunikaci, INVALID neplatné spojení, kdy pakety nelze identifikovat; time určí čas spojení --timestart <hh:mm>, --timestop <hh:mm>, --monthdays <den_v_mesici>, --weekdays <den_v_tydnu>; iprange --src-range --dst-range <IP_adresa>-<IP_adresa> určí rozsah zdrojových nebo cílových IP adres; limit --limit <n>/{s m h d} určí časový údaj, --limit-burst <n> určí počet paketů), --sport <port> zdrojový port, --dport <port> cílový port; a konečně -j <cíl> určí, jak s paketem naložit – pro tabulku „filter“ ACCEPT = přijmout, DROP = zahodit, LOG = logovat pakety, REJECT = oznámit nedostupnost, pro tabulku „nat“ SNAT --to <IP_adresa> = změnit zdrojovou IP adresu, DNAT --to <IP_adresa> = změnit cílovou IP adresu, REDIRECT --to-ports <port> = přesměrovat port; správné nastavení firewallu striktně závisí na konkrétním pořadí pravidel uvedených v /etc/sysconfig/iptables <pre># iptables -nvL --line-numbers (vypíše pravidla firewallu včetně podrobných informací) # iptables -P INPUT DROP (zahodí všechny příchozí pakety) # iptables -I INPUT -s 147.229.28.4 -j DROP (zahodí pakety přicházející z dané IP adresy) # iptables -A INPUT -p tcp --dport 22 -j DROP (zahodí pakety přicházející na daný port) # iptables -A INPUT -p tcp --dport 443 -j REJECT (odešle informaci o nedostupnosti služby) # iptables -I OUTPUT -d '!' 147.229.28.4 -j DROP (povolí jen pakety odcházející na danou IP adresu) # iptables -A OUTPUT -o eth0 -d 192.168.0.0/24 -j ACCEPT (povolí jen pakety odcházející z daného zařízení do lokální sítě) # iptables -A OUTPUT -d upc.cz -p tcp --dport 80 -j DROP (zakáže zobrazení dané URL) # iptables -A FORWARD -i eth0 -o eth1 -p tcp --dport '!' 80 -j DROP (povolí přesměrování paketů pouze na port 80) # iptables -A INPUT -p tcp --dport 50:55 -m iprange --src-range 192.168.0.1-192.168.0.10 -j ACCEPT (povolí zdrojovým adresám rozsah portů 50-55) # iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s --limit-burst 2 -j ACCEPT (omezí počet požadavků „ping“ na 2 za 1s) # iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-ports 3250 (přesměruje cílový port 80 na 3250) # iptables -t nat -A PREROUTING -p tcp --dport 80 -i eth0 -j DNAT --to 192.168.0.2:8080 (změní cílovou IP adresu a port služby) # iptables -A INPUT -j LOG (loguje všechny pakety, které nevyhovují žádnému z nastavených pravidel do /var/log/messages) # iptables -D INPUT 5 (odstraní pravidlo z řetězce „INPUT“ nacházející se ve výčtu na pátém místě)</pre>

SÍŤ A KOMUNIKACE	
firewall-cmd [<specifikace>] (platí od RHEL 7)	nastaví pravidla firewallu, která řídí příchozí a odchozí síťový provoz, --get-default-zone vypíše výchozí zónu pro síťová připojení a rozhraní, --set-default-zone=<zona> nastaví výchozí zónu pro síťová připojení a rozhraní, --get-active-zones vypíše právě aktivní zóny společně se síťovými rozhraními a zdroji používanými v těchto zónách, --get-zones vypíše všechny dostupné zóny, --list-all-zones vypíše podrobné informace o všech zónách, --zone=<zona> určí zónu (pokud není uvedena, použije se výchozí zóna), --list-all vypíše podrobné informace o zóně, --get-services vypíše všechny dostupné služby, --list-services vypíše služby přidané do zóny, --list-ports vypíše porty přidané do zóny, --add-source=<IP_adresa>[/<maska_site>] směruje veškerý provoz přicházející z dané IP adresy nebo sítě do zóny, --remove-source=<IP_adresa>[/<maska_site>] odebere ze zóny pravidlo směřující veškerý provoz pocházející z dané IP adresy nebo sítě, --add-interface=<rozhraní> směruje veškerý provoz přicházející z daného síťového rozhraní do zóny, --change-interface=<rozhraní> změní síťové rozhraní pro zónu, --add-service=<sluzba> přidá službu do zóny, --remove-service=<sluzba> odebere službu ze zóny, --add-port=<port>/<protokol> přidá port do zóny, --remove-port=<port>/<protokol> odebere port ze zóny, --add-rich-rule=<pravidlo> přidá do zóny vlastní pravidlo firewallu, které není pokryto základní firewall syntaxí, --remove-rich-rule=<pravidlo> odebere ze zóny vlastní pravidlo firewallu, --query-rich-rule=<pravidlo> ověří, zda bylo do zóny přidáno vlastní pravidlo firewallu, --list-rich-rules vypíše všechna vlastní pravidla firewallu pro danou zónu, --permanent provede trvalé nastavení (zapiše změny do <i>/etc/firewalld/</i>), --reload načte trvalé nastavení, --runtime-to-permanent převede aktuální nastavení z paměti na trvalé <pre># firewall-cmd --add-service=http --permanent (povolí službu http ve výchozí zóně) # firewall-cmd --add-port=8080/tcp --permanent (povolí TCP port 8080 ve výchozí zóně) # firewall-cmd --zone=internal --add-source=192.168.0.0/24 --permanent (směruje veškerý provoz přicházející ze sítě 192.168.0.0/24 do vnitřní zóny) # firewall-cmd --zone=internal --list-all --permanent (vypíše podrobné informace o vnitřní zóně) # firewall-cmd --add-rich-rule='rule family=ipv4 source address=183.131.80.130 reject' --permanent (blokuje veškerý provoz z dané IP adresy ve výchozí zóně) # firewall-cmd --add-rich-rule='rule family=ipv4 source address=192.168.0.15 port port=8080 protocol=tcp accept' --permanent (povolí port 8080 pro danou IP adresu ve výchozí zóně) # firewall-cmd --permanent --zone=work --add-rich-rule='rule family=ipv4 source address=192.168.0.0/26 forward-port port=80 protocol=tcp to-port=8080' (přesměruje port 80/TCP na port 8080/TCP pro danou síť v pracovní zóně) # firewall-cmd --reload (načte změny v nastavení firewallu)</pre>
ssh [[<uživatel>@]<hostitel>] [<příkaz>]	uskuteční šifrované přihlášení k existujícímu účtu na vzdáleném počítači či provede daný příkaz na vzdáleném počítači namísto spuštění interaktivního přihlašovacího shellu, jehož výstup se zobrazí na terminálu lokálního počítače, -l <uživatel> přihlásí se pod daným uživatelem, -i <soubor> určí soubor se soukromým klíčem (implicitně <i>~/.ssh/id_rsa</i>), -p <port> určí nestandardní port, -o <volba> určí volbu, která přepíše výchozí konfiguraci, -J <mezipostup> určí mezipostup, -X povolí přesměrování X11, -v podrobný výpis <pre>\$ ssh 192.168.0.20 (přihlásí se na vzdálený počítač při použití stejného uživatelského účtu na obou systémech) \$ ssh norton@prompt.cz \$ ssh -l norton prompt.cz (přihlásí se na vzdálený počítač při použití rozdílných uživatelských účtů na obou systémech) \$ ssh -o PubkeyAuthentication=no norton@192.168.0.20 (zakáže použití ssh klíče při přihlášení uživatele a vyzve ho k zadání hesla) \$ ssh -J norton@192.168.15.107 kuba@192.168.124.5 (přihlásí se na vzdálený počítač přes mezipostup při použití rozdílných uživatelských účtů na obou systémech) \$ ssh 192.168.0.20 "uname -a; who -b" (spustí na vzdáleném počítači dané příkazy, jejichž výstup zobrazí na terminálu lokálního počítače) \$ for server in centos{1..2}.example.com; do ssh \$server 'bash -s' < script.sh; done (spustí lokální skript na vzdálených počítačích)</pre>

SÍŤ A KOMUNIKACE	
ssh-keygen	vytvoří pár autentizačních klíčů – soukromý a veřejný, které slouží pro bezpečnou identifikaci uživatele během SSH připojení, aniž by musel zadat své přístupové jméno a heslo; soukromý klíč se standardně ukládá v <code>~/.ssh/id_rsa</code> , veřejný klíč bývá uložen v <code>~/.ssh/id_rsa.pub</code> a jeho obsah je třeba vložit do <code>~/.ssh/authorized_keys</code> hostitele; program rovněž vyzve uživatele k zadání ověřovací fráze (řetězec libovolných znaků včetně mezer, sloužící k ochraně soukromého klíče před zneužitím), která, pokud není prázdná, je vyžadována pro identifikaci na začátku každého připojení, -b <pocet_bitu> určí počet bitů v klíči (implicitně 3072 pro rsa), -C <komentar> přidá komentář, -t <klic> určí typ klíče – „rsa“, „dsa“, „ecdsa“ nebo „ed25519“ (implicitně „rsa“), -f <soubor> určí soubor s klíči (implicitně <code>~/.ssh/id_rsa</code> a <code>~/.ssh/id_rsa.pub</code>), -l zobrazí délku klíče v bitech a šifrovací algoritmus, -p změní ověřovací frázi, -v podrobný výpis \$ ssh-keygen -lf ~/.ssh/id_rsa awk '{print \$1}' (zobrazí délku soukromého ssh klíče v bitech)
ssh-copy-id [[<uzivatel>@]<hostitel>]	zkopíruje veřejný SSH klíč uživatele z lokálního počítače do <code>~/.ssh/authorized_keys</code> na vzdáleném počítači, -i <soubor> určí soubor s veřejným klíčem (implicitně <code>~/.ssh/id_rsa.pub</code>) \$ ssh-copy-id -i ~/.ssh/id_dsa.pub 94.112.152.47 (zkopíruje veřejný SSH klíč přihlášeného uživatele na vzdálený počítač)
ssh-agent [<prikaz>]	umožní bezpečné přihlášení na základě SSH klíčů, aniž by bylo nutné před každým připojením zadávat ověřovací frázi (využití zejména při vzdáleném spouštění příkazů na větším počtu počítačů pomocí skriptu); ssh-agent se tedy spustí před začátkem operace, příkazem „ssh-add“ se mu předá soukromý klíč a zadá se pouze jednou ověřovací fráze \$ ssh-agent sh <- ' \$ ssh-add <- ' > <passphrase> <- '
ssh-add [<soubor>]	předá dočasně soukromý SSH klíč a ověřovací frázi programu „ssh-agent“ (implicitně <code>~/.ssh/id_rsa</code> , <code>~/.ssh/id_dsa</code> , <code>~/.ssh/id_ecdsa</code> a <code>~/.ssh/id_ed25519</code>)
scp [[<uzivatel>@]<hostitel>:]<zdroj ...> [[<uzivatel>@]<hostitel>:]<cil>	uskuteční šifrovaný přenos dat mezi vzdálenými počítači přes SSH připojení, -i <soubor> určí soubor se soukromým klíčem (implicitně <code>~/.ssh/id_rsa</code>), -P <port> určí nestandardní port, -p zachová atributy souboru, -r rekurzivně, -l omezí rychlost přenosu v kB/s, -C použije kompresi, -v podrobný výpis \$ scp ~/.ssh/id_rsa.pub 192.168.0.20:~/.ssh/authorized_keys (zkopíruje obsah souboru „id_rsa.pub“ z lokálního počítače do souboru „authorized_keys“ na vzdáleném počítači) \$ scp -rv 192.168.0.20:/home/kuba/data . (zkopíruje adresář „data“ ze vzdáleného počítače do pracovního adresáře lokálního počítače) \$ scp kuba@192.168.0.20:soubor.txt 192.168.0.21: (zkopíruje soubor „soubor.txt“ z domovského adresáře uživatele na jednom vzdáleném počítači do domovského adresáře uživatele na jiném vzdáleném počítači)
sftp [[<uzivatel>@]<hostitel>]	uskuteční interaktivní šifrovaný přenos dat mezi vzdálenými počítači přes SSH připojení, -i <soubor> určí soubor se soukromým klíčem (implicitně <code>~/.ssh/id_rsa</code>), -P <port> určí nestandardní port; interaktivní příkazy: pwd vypíše cestu k pracovnímu adresáři, cd <adresar> vstoupí do daného adresáře, ls vypíše obsah pracovního adresáře, get <soubor> kopíruje vzdálený soubor na lokální počítač, put <soubor> kopíruje lokální soubor na vzdálený počítač, ! <prikaz> spustí daný příkaz na lokálním počítači, help ? nápověda, bye quit exit ukončí program
ftp [<hostitel>]	uskuteční interaktivní nešifrovaný přenos dat mezi vzdálenými počítači; interaktivní příkazy: pwd vypíše cestu k pracovnímu adresáři, cd <adresar> vstoupí do daného adresáře, ls vypíše obsah pracovního adresáře, get <soubor> kopíruje vzdálený soubor na lokální počítač, mget <soub*> kopíruje více vzdálených souborů s využitím zástupných znaků, put <soubor> kopíruje lokální soubor na vzdálený počítač, mput <soub*> kopíruje více lokálních souborů s využitím zástupných znaků, ! <prikaz> spustí daný příkaz na lokálním počítači, help ? nápověda, bye quit exit ukončí program
telnet [<hostitel>] [<port>]	uskuteční nešifrované přihlášení k existujícímu účtu na vzdáleném počítači či zjistí dostupnost daného portu; bez argumentu se spustí v interaktivním režimu \$ telnet 192.168.0.20 80 (zjistí dostupnost portu 80 na vzdáleném hostiteli)
lynx [<URL>]	zobrazí textový obsah URL adresy s možností procházet odkazy a pracovat s formuláři, q ukončí program \$ lynx prompt.cz (zobrazí obsah webové stránky)

SÍŤ A KOMUNIKACE	
curl [<URL ...>]	zobrazí zdrojový kód URL adresy či zkopíruje data z nebo na danou URL adresu, -o <soubor> určí cílový soubor (implicitně STDOUT), -O stáhne obsah URL adresy do souboru v pracovním adresáři pojmenovaném podle poslední části URL cesty (za koncovým lomítkem), -F <polozka>=<obsah> specifikuje odchozí data („@“ značí zdrojový soubor), -u <uživatel>:<heslo> určí uživatelské jméno a heslo pro ověření identity, -x [<protokol>://]<hostitel>[:<port>] určí proxy server, -v podrobný výpis <pre>\$ curl https://prompt.cz</pre> (zobrazí zdrojový kód webové stránky) <pre>\$ curl -o skript https://prompt.cz/_media/wiki/lrx.sh</pre> (stáhne obsah skriptu „lrx.sh“ z webové stránky do souboru „skript“, který zároveň vytvoří) <pre>\$ curl -O https://prompt.cz/_media/wiki/lrx.sh</pre> (stáhne skript „lrx.sh“ z webové stránky do pracovního adresáře) <pre>\$ curl ipinfo.io/ip</pre> (zobrazí veřejnou IP adresu hostitele)
wget [<URL ...>]	stáhne obsah URL adresy do pracovního adresáře, -P <adresar> určí adresář pro stažení, -c pokračuje ve stahování částečně staženého souboru po přerušení přenosu, -r rekurzivně, -t <n> určí počet pokusů o stažení <pre>\$ wget https://prompt.cz/_media/wiki/lrx.sh</pre> (stáhne skript „lrx.sh“ z webové stránky do pracovního adresáře)
mail	zobrazí obsah poštovní schránky přihlášeného uživatele (/var/spool/mail/<uživatel>), -f zobrazí obsah schránky s již přečtenými zprávami (/home/<uživatel>/mbox); interaktivní příkazy: p p <n> zobrazí obsah poslední či dané zprávy, r odpoví na zprávu, d d <m-n> d * smaže současnou, dané či všechny zprávy, q ukončí program
mail <adresa>	odešle zprávu na adresu příjemce, -r <adresa> určí adresu odesílatele, -s <predmet> určí předmět, -c <adresa> určí adresu kopie (CC), -b <adresa> určí adresu skryté kopie (BCC) <pre>\$ mail root < info.txt</pre> <pre>\$ cat soubor mail -s "navrh" kuba -c root</pre> <pre>\$ echo "ahoj Kubo" mail -s "pozdrav" kuba</pre> zprávu lze rovněž poslat tímto způsobem: <pre>\$ mail <adresa> <- ' Subject: <predmet> <- ' <text> <- ' .- ' nebo Ctrl+d Cc: <adresa> <- '</pre>
wall [<zprava>]	odešle zprávu na terminály všech přihlášených uživatelů na stejném počítači
write [<uživatel>] [<terminal>]	odešle zprávu danému uživateli na stejném počítači; pokud je uživatel přihlášen na více terminálech současně, lze terminál specifikovat, jinak program zvolí terminál, na kterém byl uživatel aktivní naposledy <pre>\$ write kuba <- ' <text> Ctrl+d</pre> <pre>\$ echo "Ahoj" write kuba</pre>
talk <uživatel>[@<hostitel>] [<terminal>]	umožní komunikaci v reálném čase mezi dvěma uživateli na stejném počítači nebo na různých počítačích, pokud na obou systémech používají stejná uživatelská jména; pokud je uživatel přihlášen na více terminálech současně, lze terminál specifikovat, jinak program zvolí terminál, na kterém byl uživatel aktivní naposledy <pre>\$ talk tom@prompt.cz <- ' <text> <- ' Ctrl+c</pre>
mesg [y n]	vypíše či nastaví dostupnost terminálu přihlášeného uživatele pro příjem zpráv programu „wall“, „talk“ nebo „write“ („y“ = ano, „n“ = ne)
who -w	vypíše dostupnost terminálů přihlášených uživatelů pro příjem zpráv programu „wall“, „talk“ nebo „write“ („+“ = ano, „-“ = ne)

From:
<https://prompt.cz/> - Prompt.cz

Permanent link:
<https://prompt.cz/sit-a-komunikace>

Last update: 2025/04/24 15:59

