Višja strokovna šola Velenje – Informatika Murska Sobota

Računalniške komunikacije in omrežja II

OMREŽNI UKAZI OS LINUX

Priročnik za laboratorijske vaje

Druga popravljena in razširjena izdaja

Pripravil: dr. **Iztok Fister**
E-mail: iztok.fister@mdi2.net

Murska Sobota
Januar, 2009

KAZALO

1 Plast omrežnega vmesnika ethernet

V tem poglavju obravnavamo orodja, s katerimi prikažemo ali preverimo karakteristike omrežnega vmesnika ethernet. Če želimo, da višje plasti protokolnega sklada TCP/IP delujejo v redu, moramo zagotoviti operativnost nižje nivojskih plasti. Orodja, ki jih opisujemo v nadaljevanju, omogočajo preverjanje pravilnega delovanja omrežnega vmesnika ethernet na OS Linux.

1.1 arp

Uporabljamo za prikazovanje in rokovanje zapisov tabele **arp**. V najenostavnejši obliki klicanje ukaza **arp** brez parametrov prikaže trenutno stanje tabele.

Prikazovanje vsebine tabele arp

Vsebino tabele **arp** na vmesniku **eth0** prikažemo z naslednjim ukazom

***arp -a –n –i eth0*,**

kjer pomenijo

-a	prikaži vse zapise,
-n	zapise prikaži v numerični obliki (brez DNS),
-i	določeni omrežni vmesnik.

Dodajanje zapisov v tabelo arp

Vsebino tabele **arp** na vmesniku **eth0** dodamo z naslednjim ukazom

***arp -s 192.168.100.17 –i eth0 –D eth0 pub*,**

kjer pomenijo

-s	dodaj novi zapis,
-i	določeni omrežni vmesnik,
-D	preberi MAC naslov iz omrežnega vmesnika eth0,
pub	objavi naslov v omrežju.

Brisanje zapisov iz tabele arp

Zapis v tabeli **arp** na vmesniku **eth0** brišemo z naslednjim ukazom

***arp –i eth0 -d 192.168.100.17*,**

kjer pomenijo

-i	določeni omrežni vmesnik,
-d	briši zapis z IP naslovom.

1.2 arping

Skoraj nepoznani ukaz **arping** deluje podobno kot ukaz **ping**, vendar na plasti omrežnih vmesnikov. Medtem ko z ukazom ping testiramo dostopnost do IP naslova s pošiljanjem paketov ICMP, z ukazom arping testiramo dostopnost do IP naslova, ki ga nosi gostitelj v lokalnem omrežju, s pošiljanjem paketov ARP.

Prikaz dostopnosti IP naslova na lokalnem omrežju ethernet

Dostop do gostitelja, ki je na lokalno omrežje ethernet priključen z vmesnikom **eth0,** prikažemo z naslednjim ukazom

***arping –I eth0 -c 2 192.168.100.17*,**

kjer pomenijo

- ***-I*** določeni omrežni vmesnik **eth0**,
- ***-c*** prekini izvajanje ukaza, ko dobiš predpisano število odgovorov.

Odkrivanje dvojnih IP naslovov v lokalnem omrežju ethernet

Ukaz **arping** lahko odkriva že uporabljene IP naslove v lokalnem omrežju. Ta postopek, ki ga imenujemo tudi odkrivanje dvojnih IP naslovov, prikazuje naslednji ukaz

***arping –D –q –I eth0 -c 2 192.168.100.17*,**

kjer pomenijo

- ***-D*** način odkrivanja dvojnih IP naslovov,
- ***-q*** tihi način delovanje, t.j. ne prikazuj ničesar,
- ***-I*** določeni omrežni vmesnik **eth0**,
- ***-c*** prekini izvajanje ukaza, ko dobiš predpisano število odgovorov.

Vsak odgovor, ki označuje, da je IP naslov že uporabljen, povzroči izhod iz ukaza arping z izhodno kodo večjo od nič (običajno rc = 1). Izhodno kodo določenega programa vidimo z ukazom

echo $?

V primeru, da je omrežni vmesnik ethernet v stanju neaktiven, ukaz vrne izhodno kodo rc = 2.

1.3 ip link

Ukaz **ip link** omogoča prikaz informacij o omrežni plasti, aktivira in deaktivira omrežni vmesnik, spreminja stanje omrežnega vmesnika in velikost MTU.

Prikaz karakteristik omrežnega vmesnika

Karakteristike omrežnega vmesnika **eth0** prikažemo z ukazom

ip link show dev eth0.

Ukaz **ip link show** daje zelo podobne rezultate ukazu **ifconfig**.

Spreminjanje karakteristik omrežnega vmesnika

Stanje omrežnega vmesnika opišemo z zastavicami prikazanih v Tabeli 1.

Tabela 1: Stanja omrežnih vmesnikov

Zastavica	Mogoče stanje	Opis stanja
arp	on|off	Postavi ali odpravi protokol ARP na omrežnem vmesniku.
pomisc	on|off	Postavi ali odpravi promiskuitetni način delovanja vmesnika.
allmulti	on|off	Sprejemaj ali ignoriraj vse multicast pakete z omrežja.
multicast	on|off	Postavi ali odpravi multicast način delovanja vmesnika.
dynamic	on|off	Spremeni zastavico DYNAMIC na omrežnem vmesniku.

Ukazi za spreminjanje karakteristik omrežnega vmesnika so naslednji

ip link set dev eth0 arp off
ip link set dev eth0 multicast off promisc on
ip link set dev eth0 allmulti on dynamic off

Deaktiviranje omrežnega vmesnika

Omrežni vmesnik **eth0** deaktiviramo z naslednjim ukazom

ip link set dev eth0 down

Ukaz je analogija ukazu ***ifconfig <interface> down***.

Aktiviranje omrežnega vmesnika

Omrežni vmesnik **eth0** aktiviramo z naslednjim ukazom

ip link set dev eth0 up

Ukaz je analogija ukazu **ifconfig <interface> up**.

Spreminjanje velikosti MTU

Velikost MTU omrežnega vmesnika **eth0** spremenimo z naslednjim ukazom

ip link set dev eth0 mtu 1412

Ukaz je analogija ukazu ***ifconfig <interface> mtu***.

1.4 ip neighbor

Ukaz **ip neighbor** omogoča prikaz, dodajanje in brisanje zapisov tabele **arp**, ter brisanje celotne tabele **arp**. Ukaz je ekvivalent ukaza **arp**.

Prikaz vsebine tabele arp na omrežnem vmesniku eth0

Vsebino tabele **arp** prikažemo z ukazom

ip neighbor show dev eth0

Ukaz je enakovreden ukazu ***arp –n –i eth0*.**

Dodajanje stalnega zapisa v tabelo arp

Stalni zapis (angl. permanent entry) dodamo v tabelo **arp** omrežnega vmesnika **eth0** z ukazom

***ip neighbor add 192.168.100.1 lladdr 00:c0:7b:7d:00:c8 dev eth0 nud permanent*,**

kjer pomenijo
lladd - MAC naslov omrežnega vmesnika (angl. Link Line Address),
dev – ime omrežnega vmesnika,
nud – odkrivanje dosegljivosti sosedov (angl. Neighbor Reachability Detection)
s stanji: **permanent**, **noarp**, **reachable** in **stale**.

Spreminjanje zapisa v tabeli arp

Zapis v tabeli **arp** spremenimo z ukazom

***ip neighbor change 192.168.99.254 lladdr 00:80:c8:27:69:2d dev eth0*,**

kjer pomenita
lladd - MAC naslov omrežnega vmesnika (angl. Link Line Address),
dev – ime omrežnega vmesnika.

Brisanje zapisa v tabeli arp

Zapis v tabeli **arp** zbrišemo z ukazom

***ip neighbor del 192.168.99.254 dev eth0*.**

Brisanje celotne tabele arp

Celotno tabelo **arp** zbrišemo z ukazom

***ip neighbor flush*.**

1.5 mii-tool

Ukaz **mii-tool** omogoča prikaz hitrosti omrežnega vmesnika, ki je priključen na omrežje ethernet. Informacije, ki jih omenjeni ukaz sporoča, prikazuje Tabela 2.

Tabela 1: Okrajšave hitrosti omrežnih vmesnikov

Hitrost vmesnika	Opis
10baseT-HD	10 Mbitov half duplex
10baseT-FD	10 Mbitov full duplex
100baseTx-HD	100 Mbitov half duplex
100baseTx-FD	100 Mbitov full duplex

Odkrivanje statusa omrežnega vmesnika z ukazom mii-tool

Hitrost in način delovanja omrežnega vmesnika prikazuje naslednji ukaz

***mii-tool*.**

Postavitev hitrosti in način delovanja omrežnega vmesnika z ukazom mii-tool

Hitrost in način delovanja omrežnega vmesnika postavimo z naslednjim ukazom

***mii-tool --force 10baseT-FD*.**

Če želimo, da začne omrežni vmesnik delovati z novimi nastavitvami, ga je potrebno ponovno zagnati, kar naredimo z ukazom

***mii-tool --restart*.**

2 Omrežna plast IP

2.1 ifconfig

Spreminjanje IP naslovov in prehodov

Ko se računalnik naloži in priključi na ethernet omrežje s privzeto konfiguracijo, lahko to v istem hipu spremenimo. Trenutno konfiguracijo omrežnega vmesnika pogledamo z ukazom

ifconfig eth0

Vsebino usmerjevalne tabele prikažemo z ukazom

route –n

Proces spreminjanja IP naslova omrežnega vmesnika vsebuje tri korake:

1. aktivni vmesnik deaktiviramo,
2. konfiguriramo novi IP naslov vmesnika,
3. dodamo nov privzeti prehod.

Omrežni vmesnik deaktiviramo z ukazom

ifconfig eth0 down

Stranski učinki deaktiviranja omrežnega vmesnika z ukazom **ifconfig** so naslednji:

- izgubimo vse konfigurirane IP naslove na deaktiviranem vmesniku,
- prekinemo vse povezave na deaktiviranem vmesniku,
- iz usmerjevalne tabele odstranimo vse prehode, ki imajo izvor ali ponor na IP naslovu deaktiviranega vmesnika,
- ugasnemo omrežni vmesnik.

Omrežni vmesnik, ki je priključen na omrežje 192.168.99.0, aktiviramo z ukazom

ifconfig eth0 192.168.99.14 netmask 255.255.255.0 up

Stranski učinki aktiviranja omrežnega vmesnika z ukazom **ifconfig** so naslednji:

- priključimo omrežni vmesnik,
- konfigurirani IP naslov priredimo omrežnemu vmesniku,
- v usmerjevalno tabelo vpišemo lokalni, omrežni in broadcast prehod.

Pri deaktiviranju omrežnega vmesnika izgubimo tudi privzeti prehod, ki ga je potrebno ponovno postaviti. Privzeti prehod konfiguriramo z naslednjim ukazom

route add default gw 192.168.99.254

IP naslov privzetega prehoda običajno kaže na usmerjevalnik, ki je priključen na Interenet. Če želimo komunicirati z gostitelje na omrežju, ki ni na Internetu, to običajno rešujemo z dodajanjem statičnih prehodov. Statični prehod dodamo z ukazom

route add –net 192.168.98.0 netmask 255.255.255.0 gw 192.168.99.1

zbrišemo pa z ukazom

route del –net 192.168.98.0 netmask 255.255.255.0 gw 192.168.99.1

Spreminjanje maksimalne prenosne enote MTU

Včasih se pojavi potreba po spremembi maksimalne prenosne enote MTU. MTU lahko spremenimo z ukazom

ifconfig eth0 mtu 1412

Spreminjanje zastavic omrežnih vmesnikov

Vsaka naprava na sistemu ima zastavice, ki označujejo stanje v katerem je lahko le-ta. Ta stanja (Tabela 1) lahko spreminjamo s pomočjo ukaza **ifconfig**.

Tabela 1: Stanja omrežnih vmesnikov

Zastavica	Ukaz	Opis
UP	-	Naprava deluje.
BROADCAST	[-]*broadcast*	Naprava pošilja promet vsem gostiteljem v omrežju.
RUNNING	-	???
MULTICAST	*multicast*	Naprava pošilja in sprejema pakete multicast.
ALLMULTI	[-] *allmulti*	Naprava sprejema vse pakete multicast v omrežju.
PROMISC	[-] *promisc*	Naprava sprejema ves promet v omrežju.

Zastavica PROMISC je zelo uporabna pri spremljanju prometa v omrežju.

2.2 ip address

Ukaz **ip address** prikaže IP naslove pridružene omrežnim vmesnikom, doda IP naslove, briše IP naslove ali odstrani vse IP naslove na določeni napravi.

Prikazovanje informacij omrežnih vmesnikov

IP naslove omrežnih vmesnikov prikaže ukaz

ip address show

Če želimo prikazati IP naslov omrežnega vmesnika **eth0**, uporabimo ukaz

ip address show dev eth0

Dodajanje IP naslovov omrežnim vmesnikom

IP naslov omrežnemu vmesniku dodamo z ukazom

ip address add 192.168.99.37/24 brd + dev eth0

Brisanje IP naslovov omrežnim vmesnikom

IP naslov omrežnemu vmesniku zbrišemo z ukazom

ip address del 192.168.99.37/24 brd + dev eth0

Odstranitev vseh IP naslovov omrežnih vmesnikov

Vse konfigurirane IP naslove na vseh omrežnih vmesnikov odstranimo z ukazom

ip address flush

Če želimo odstraniti vse IP naslove omrežnega vmesnika **eth0**, uporabimo ukaz

ip address flush dev eth0

3 Usmerjanje IP

3.1 route

Usmerjanje in razumevanje usmerjanja v omrežjih IP je ena izmed osnov, ki jih potrebujemo, če želimo izkoristiti fleksibilnost omrežja in storitev IP. V omrežjih IP ni dovolj samo, da naslovimo gostitelja na tem omrežju, ampak potrebujemo tudi smer, po kateri pridemo do njega. Eden izmed ključnih elementov pri oblikovanju omrežij IP je statično usmerjanje, t.j. za vsak paket IP se usmerjevalnik neodvisno odloča, po kateri poti ga bo poslal do naslovnika. V tem poglavju pregledamo, s katerimi orodji v OS Linux pregledamo vsebino usmerjevalnih tabel, kako to vsebino spreminjamo, kako dodajamo nove usmerjevalne poti ali le-te brišemo. Pri tem uporabljamo dobro znan ukaz **route**, ki ga lahko nadomestimo z ukazom paketa iproute2 **ip route**. Na koncu obravnavamo tudi filtriranje paketov z ukazom **ip rule**.

Prikazovanje vsebine usmerjevalne tabele

Vsebino usmerjevalne tabele prikažemo z ukazom

***route –n*,**

kjer pomeni

-n	zapise prikaži v numerični obliki, t.j. brez razreševanja.

Prikazovanje vsebine usmerjevalnega pomnilnika (angl. Routing Cache)

Vsebino usmerjevalnega pomnilnika, ki ga uporablja jedro OS (angl. kernel) kot tabelo namenjeno hitremu iskanju ustreznih prehodov, prikažemo z ukazom

***route –Cen*,**

kjer pomenijo

-C	prikaži vsebino usmerjevalnega pomnilnika,
-e	uporabi obliko izpisa ukaza **netstat**,
-n	zapise prikaži v numerični obliki, t.j. brez razreševanja.

Dodajanje statičnega prehoda

Statični prehod dodamo z ukazom

***route add –net 10.38.0.0 netmask 255.255.0.0 gw 192.168.100.1*,**

kjer pomenijo

-net	dodaj prehod do omrežja,
10.38.0.0	naslov IP ponornega omrežja (angl. destination network),
netmask	maska IP omrežja/podomrežja,
255.255.0.0	omrežje = 10, podomrežje = 38,
gw	prehod,
192.168.100.1	naslov IP prehoda.

Statični prehod do gostitelja preko omrežnega vmesnika **eth0** dodamo z ukazom

***route add –host 205.254.211.184 dev eth0*,**

kjer pomenijo

-host	dodaj prehod do gostitelja,
205.254.211.184	naslov IP ponornega gostitelja,
dev	omrežni vmesnik,
eth0	eth0.

Privzeti prehod dodamo z ukazom

***route add default gw 192.168.100.1*,**

kjer pomenijo

default	dodaj privzeti prehod,
gw	prehod,
192.168.100.1	naslov IP prehoda.

Brisanje statičnega prehoda

Privzeti prehod brišemo z ukazom

***route del default gw 192.168.98.254*,**

kjer pomenijo

default	privzeti prehod,
gw	prehod,
192.168.98.254	naslov IP privzetega prehoda,

statični pa z ukazom

***route del –net 10.38.0.0 netmask 255.255.0.0 gw 192.168.100.1*,**

kjer pomenijo

-net	zbriši prehod do omrežja,
10.38.0.0	naslov IP ponornega omrežja,
netmask	maska IP omrežja/podomrežja,
255.255.0.0	omrežje = 10, podomrežje = 38,
gw	prehod,
192.168.100.1	naslov IP privzetega prehoda,

3.2 ip route

Prikazovanje vsebine usmerjevalnih tabel

Vsebino usmerjevalne tabele prikažemo z ukazom

***ip route show*.**

Vsebino lokalne usmerjevalne tabele prikažemo z ukazom

***ip route show table local*.**

Z ukazom **route** ne moremo pregledati vsebino lokalne usmerjevalne tabele. Gornji ukaz daje podrobnejše informacije o omrežjih IP, na katera je gostitelj priključen neposredno in na način, kako obravnava gostitelj posebne naslove IP kot so Broadcast ali lokalni konfigurirani naslovi IP.

Prikazovanje vsebine usmerjevalnega pomnilnika (angl. Routing Cache)

Vsebino usmerjevalnega pomnilnika, ki ga uporablja jedro OS (angl. kernel) kot tabelo namenjeno hitremu iskanju ustreznih prehodov, prikažemo z ukazom

***ip route show cache*.**

Ukaz lahko vsebuje tudi IP naslov omrežnega vmesnika

***ip route show cache 192.168.100.17*.**

Dodajanje statičnega prehoda

Statični prehod dodamo z ukazom

***ip route add 10.38.0.0/16 via 192.168.100.1*,**

kjer pomenijo

10.38.0.0/16	naslov IP ponornega omrežja v kanonični obliki,
via	smer (prek),
192.168.100.1	naslov IP usmerjevalnika.

Statični prehod prek določenega gostitelja prepovemo z ukazom

***ip route add prohibit 209.10.26.51*,**

kjer pomenijo

prohibit	prepovedan prehod,
209.10.26.51	naslov IP usmerjevalnika.

Ukaz predstavlja ekvivalent ukaza **ipchains REJECT**. Statični prehoda prek določenega gostitelja lahko prepovemo tudi za določenega gostitelja. Ukaz v tem primeru je naslednji

***ip route add prohibit 209.10.26.51 from 192.168.99.35*,**

kjer pomenijo

prohibit	prepovedan prehod,
209.10.26.51	naslov IP usmerjevalnika,
from	za pakete, ki prihajajo od gostitelja,
192.168.99.35	naslov IP izvornega gostitelja.

Dodajanje privzetega prehoda

Privzeti prehod dodamo z ukazom

ip route add default via 192.168.99.254,

kjer pomenijo

default	privzeti prehod,
via	smer (prek),
192.168.99.254	naslov IP usmerjevalnika.

Dodajanje prehoda NAT

Ukaz prepiše ponorni naslov paketov IP z zasebnim (inbound processing). NAT (angl. Network Address Translation) zapis dodamo z ukazom

ip route add nat 205.254.211.17 via 192.168.100.17**,**

kjer pomenijo

nat	zapis tipa NAT,
205.254.211.17	javni ponorni naslovi IP gostiteljev,
via	smer (prek),
192.168.100.17	**za**sebni ponorni naslovi IP gostiteljev.

V tem primeru vsak paket IP, ki pride do usmerjevalnika z ponornim naslovom IP 205.254.211.17 ta prevede v zasebni naslov IP 192.168.100.17. Lokalno usmerjevalno tabelo z zapisi NAT prikažemo z ukazom

ip route show table local | grep nat**.**

Brisanje statičnega prehoda

Statični prehod brišemo z ukazom

ip route del 10.38.0.0/16 via 192.168.100.1 dev eth0**,**

kjer pomenijo

10.38.0.0/16	ponorno omrežje v kanonični obliki,
via	smer (prek),
192.168.100.1	naslov usmerjevalnika IP,
dev	omrežni vmesnik,
eth0	ime **eth0**.

Spreminjanje statičnega prehoda

Statični prehod spremenimo z ukazom

***ip route change default via 192.168.99.113 dev eth0*,**

kjer pomenijo

default	privzeti prehod,
via	smer (prek),
192.168.99.113	naslov IP usmerjevalnika,
dev	omrežni vmesnik,
eth0	ime **eth0**.

Statični prehod lahko spremenimo tako, da ga zbrišemo in ponovno definiramo. Ta postopek avtomatizira ukaz ip route change, ki spremeni naslov IP novega prehoda.

Brisanje celotnih usmerjevalnih tabel

Usmerjevalno tabelo brišemo v celoti z ukazom

***ip route flush*,**

celotno lokalno tabelo z ukazom

***ip route flush table local*,**

celoten usmerjevalni pomnilnik pa z ukazom

***ip route flush cache*.**

3.3 ip rule

Ukaz **ip rule** je orodje za upravljanje podatkovne baze usmerjevalnih politik (angl. Routing Policy DataBase, krajše RPDB) pod OS Linux.

Prikazovanje vsebine RPDB

Vsebino podatkovne baze usmerjevalnih politik RPDB prikažemo z ukazom

***ip rule show*.**

Dodajanje filtrirnih pravil v RPDB

Dodajanje pravil v RPDB je enostavno, t.j. pravilo izbira paket IP glede na njegove karakteristike. Karakteristike, ki jih uporabljamo kot selekcijske kriterije so: IP naslov izvora ali ponora, tip storitve (ToS) in omrežni vmesnik na katerega pošljemo paket.

Velika prednost RPDB je porazdelitev različnih vrst prometa različnim dobaviteljem glede na karakteristiko paketa. OS Linux podpira več usmerjevalnih tabel. Poleg dveh splošno znanih usmerjevalnih tabel (local in main) jedro OS podpira dodatnih 252 usmerjevalnih tabel. Sistem z več usmerjevalnimi tabelami omogoča fleksibilno infrastrukturo na vrhu katere lahko implementiramo kompleksno usmerjevalno politiko. Jedro OS Linux tradicionalne usmerjevalne tabele (inbound traffic) kombinira s podatkovno bazo usmerjevalnih politik (outbound traffic). Vsaka usmerjevalna tabela lahko vsebuje poljubno število zapisov. Vrstni red uporabe usmerjevalnih tabel določa uporabnik.

***ip rule add reject from 192.168.99.35 table 8*.**

kjer pomenijo

reject	zavrzi paket,
from	ki prihaja od,
192.168.99.36	gostitelja z danim IP naslovom,
table	pravilo shrani v tabelo,
8	številka tabele.

Dodajanje pravil NAT

Ukaz uporabljamo za prepisovanje izvornega naslova IP paketov med usmerjevalnim procesom (outbound processing). Pri tem ostane ponorni IP naslov paketa nespremenjen. Novo pravilo NAT dodamo z ukazom

***ip rule add nat 205.254.211.17 from 192.168.100.17*,**

kjer pomenijo

nat	paket NAT,
205.254.211.17	javni naslov izvornega IP gostitelja,
from	prevedi iz,
192.168.100.17	zasebnega naslova IP.

Brisanje pravil NAT

Pravilo NAT brišemo z ukazom

***ip rule del nat 205.254.211.17 from 192.168.100.17*,**

kjer pomenijo

nat	paket NAT,
205.254.211.17	javni naslov izvornega IP gostitelja,
from	prevedi iz,
192.168.100.17	zasebnega naslova IP.

4 Diagnostika IP

4.1 ping

Ukaz predstavlja eno od najstarejših orodij. V najenostavnejši obliki ping sprašuje drugega gostitelja, ali je še živ in zapisuje čas, ki ga potrebuje diagnostični paket med poslano zahtevo in vrnjenim odgovorom. Poleg testiranja dostopnosti gostitelja pa lahko z njim pošljemo gostitelju določeno število paketov, obremenimo omrežje, zapišemo prehod, ki ga paket uporabi, postavimo TTL, določimo ToS in izvorni IP naslov.

Test dostopnosti gostitelja

Ukaz v najenostavnejši obliki zapišemo kot

***ping –n 192.168.98.254*,**

kjer z opcijo **–n** uporabljamo IP naslov gostitelja, t.j. izognemo se uporabi DNS pri razrešitvi imena gostitelja. Ko s tipko Ctrl-C ukaz *ping* prekinemo, le-ta izračuna minimalni, povprečni in maksimalni čas potreben za prenos in sprejem zahtevanega paketa, kot tudi odstotek pri prenosu izgubljenih paketov.

Število poslanih paketov gostitelju omejimo z opcijo **–c**. Če detajlnih izpisov pri pošiljanju paketov ne želimo, uporabimo opcijo **–q**. Ukaz zapišemo kot

***ping –q –c 10 –n 192.168.98.254*,**

kjer pomenijo

-q	prepreči detajlni izpis,
-c 10	ponornemu gostitelju pošlji 10 paketov,
-n	uporabi IP naslov gostitelja.

Obremenitev omrežja

Če želimo testirati, koliko paketov lahko prenese določeno omrežje in kako se zmanjša odzivnost le-tega, uporabimo ukaz

ping –c 400 –f –n 192.168.99.254,

kjer pomenijo

-c 400	ponornemu gostitelju pošlji 400 paketov,
-f	ping s poplavo paketov,
-n	uporabi IP naslov gostitelja.

Z uporabo večjih paketov lahko omrežje obremenimo še bolj. V primeru, ko v omrežje pošljemo pakete velikosti 512 bajtov, je sintaksa ukaza **ping** naslednja

ping –s 512 –c 400 –f –n 192.168.99.254.

Zapisovanje omrežnih prehodov

Ping z opcijo **–R** lahko uporabimo tudi za izpisovanje prehodov na poti od izvornega do ponornega gostitelja. Ukaz je po funkcionalnosti ekvivalenten ukazu **traceroute**. Omrežne prehode izpišemo z ukazom

ping –c 2 –n –R 192.168.99.35.

Postavitev TTL na paketu ping

Ta vrsta ukaza **ping –t** ima dvomljivo praktično vrednost. Z njim določimo maksimalno število skokov, ki jih lahko paket naredi, preden ga zavržemo. Ukaz ima naslednjo obliko

ping –c 1 –t 4 192.168.99.35,

kjer število skokov omejimo na 4.

Postavitev ToS na paketu ping

ToS (angl. Type of Service) uporabljamo na hrbteničnih Internetnih omrežjih, ki jih nudijo Internetni ponudniki storitev ISP z implementiranimi SLA (angl. Service Level Agreements). Pri takih ponudnikih lahko z uporabo ukaza **ping –Q** preverimo, ali se ponudniki ISP držijo svojega dela pogodbe. Ukaz ima naslednjo obliko

ping –c 2 –Q 8 -n 192.168.99.35,

kjer pomeni

-Q 8 vrednost ToS.

4.2 traceroute

Ukaz izpiše prehode, preko katerih potuje paket IP od izvornega gostitelja, da bi dosegel ponornega gostitelja. Prav tako prikaže vse skoke na povezavi od lokalnega do oddaljenega gostitelja in določi vse vmesne usmerjevalnike. Privzeti tip paketa, ki ga kreira **traceroute**, je UDP. Prvi paket naslavlja vrata udp/33435 in vsak naslednji paket povečuje številko vrat. To omogoča ukazu **traceroute**, da ugotovi, kateri odgovor ustreza kateremu izhodnemu paketu. Najenostavnejša uporaba ukaza **traceroute** je naslednja

***traceroute –n 192.168.99.35*.**

Ukaz **traceroute** podpira veliko opcij. V nadaljevanju si poglejmo samo nekaj najpomembnejših. Primer:

***traceroute –S 1 –m 30 –n 192.168.99.35*,**

kjer pomenijo

-S 1 postavi začetno vrednost TTL=1,

-m 30 postavi končno vrednost TTL=30 (prekini, po 30 skokih),
-n uporabljaj IP naslove ne imena gostiteljev.

4.3 netstat

Ukaz **netstat** prikaže podrobnosti o omrežju, na katerega je računalnik priključen. Te vključujejo tudi usmerjevalne tabele, maskirne tabele NAT, statistike omrežnih vmesnikov in statistike aktivnih vtičnic. Če ukaz uporabimo brez parametrov, prikaže vse aktivne Internetne povezave.

Prikaz statusa IP vtičnic

Status vtičnic prikazuje naslednji ukaz

netstat --inet --numeric-hosts

kjer pomenijo

--inetd vsi višjenivojski IP protokoli, t.j. **--tcp** in **--udp**,
--numeric-hosts uporabi numerični izpis gostiteljev.

Prikaz glavne usmerjevalne tabele

Ena izmed najpomembnejših nalog ukaza **netstat** je prikaz usmerjevalne tabele. Najpogostejša je uporaba ukaza v obliki

netstat –rn,

kjer pomenijo

-r prikaz usmerjevalne tabele,
-n uporabi numerični izpis gostiteljev.

Prikaz usmerjevalne tabele v pomnilniku

Usmerjevalno tabelo v pomnilniku prikažemo z ukazom

netstat –rnC.

Prikaz statistike omrežnega vmesnika

Statistiko omrežnega vmesnika prikažemo z ukazom

netstat –i.

Ukaz

netstat -ie

prikaže iste rezultate kot ***ifconfig***.

Prikaz maskirne tabele

Za računalniki, ki izvajajo maskiranje, t.j. običajno dvodomni usmerjevalniki s filtriranjem, je pomembno orodje, kako pogledati vsebino maskirne tabele NAT. To prikažemo z naslednjim ukazom

***netstat –Mn*.**

4.4. nslookup

S temi ukazi postavljamo vprašanja strežniku DNS, ki določa informacije o omrežnih gostiteljih. Če iščemo gostitelja preko IP-naslova, ukaz vrne ime gostitelja in obratno, če iščemo gostitelja preko imena, nam ukaz vrne IP-naslov gostitelja. Vsi trije ukazi za določanje strežnika DNS uporabljajo datoteko */etc/resolv.conf*.

4.5 tcpdump

Ukaz **tcpdump** lahko zavzame in shrani tok paketov za kasnejšo analizo. Pogosto se znajdemo v situaciji, ko ne moremo uporabiti grafičnega analizatorja kot npr. **ethereal**. Na srečo pa lahko z ukazom **tcpdump** kreiramo podatkovne datoteke in jih pozneje obdelamo z orodji kot **ethereal**. Pakete s **tcpdump** filtriramo z zelo kompleksnimi izrazi, ki analizirajo glavo IP. Vsebino paketov v heksadecimalni obliki izpišemo z opcijo **–x**. Ukaz tcpdump v normalni obliki

tcpdump –n.

4.6 ethereal

Ethereal je paketni dekoder, ki prikaže promet skozi celotno omrežje. Ta opcija izkorišča sposobnost Ethernet vmesnika, ki v *promiskuitetnem* načinu omogoča odkrivanje vhodnih in izhodnih podatkov v omrežju. Program običajno uporabljamo pri analizi TCP zmogljivostih.