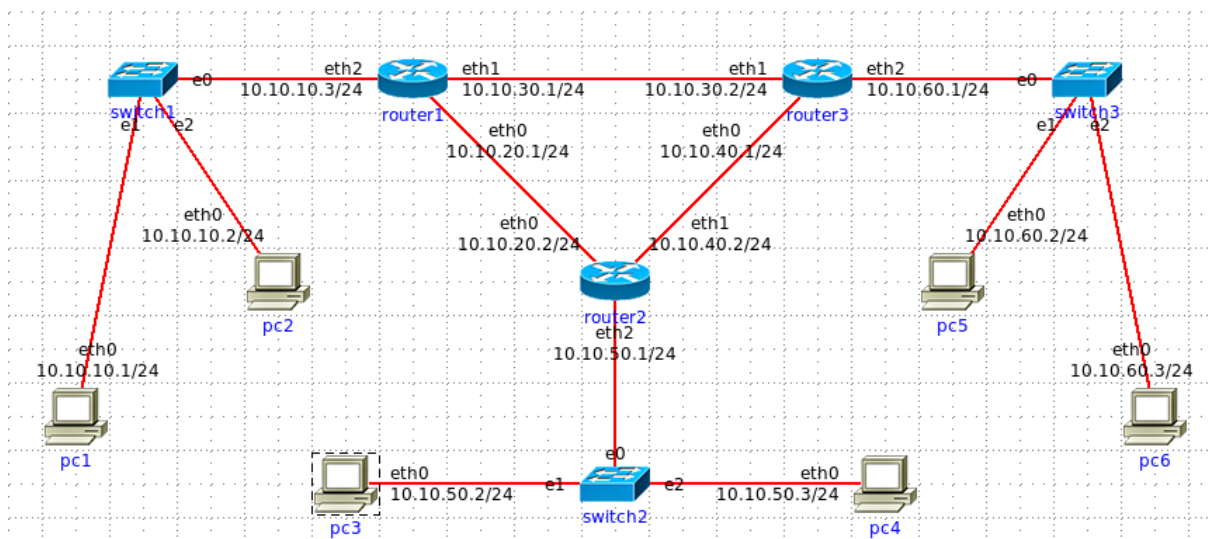


2.Laboratorijska vježba

Boris Boronjek, JMBAG:0036531473

22. U emulatoru/simulatoru IMUNES konstruirajte mrežu koja sadrži tri podmreže povezane usmjeriteljima (Slika 3.16). Konfigurirajte statičko usmjeravanje između svih podmreža (dakle, bez korištenja protokola za usmjeravanje). Sučeljima računala i usmjeritelja dodijelite IP-adrese iz raspona 10.10.10.0/24, 10.10.20.0/24, 10.10.30.0/24, 10.10.40.0/24, 10.10.50.0/24 i 10.10.60.0/24. Ispišite tablice usmjeravanja svih računala i usmjeritelja.



pc1:

Odredište	Sljedeći skok	Mrežno sučelje
default	10.10.10.3	eth0
10.10.10.0/24	link#2	eth0
10.10.10.1	link#2	lo0
localhost	link#1	lo0

pc2:

Odredište	Sljedeći skok	Mrežno sučelje
default	10.10.10.3	eth0
10.10.10.0/24	link#2	eth0
10.10.10.2	link#2	lo0
localhost	link#1	lo0

pc3:

Odredište	Sljedeći skok	Mrežno sučelje
default	10.10.50.1	eth0
10.10.50.0/24	link#2	eth0
10.10.50.2	link#2	lo0
Localhost	link#1	lo0

pc4:

Odredište	Sljedeći skok	Mrežno sučelje
default	10.10.50.1	eth0
10.10.50.0/24	link#2	eth0
10.10.50.3	link#2	lo0
Localhost	link#1	lo0

pc5:

Odredište	Sljedeći skok	Mrežno sučelje
default	10.10.60.1	eth0
10.10.60.0/24	link#2	eth0
10.10.60.2	link#2	lo0
localhost	link#1	lo0

pc6:

Odredište	Sljedeći skok	Mrežno sučelje
default	10.10.60.1	eth0
10.10.60.0/24	link#2	eth0
10.10.60.3	link#2	lo0
localhost	link#1	lo0

router1:

Odredište	Sljedeći skok	Mrežno sučelje
10.10.10.0/24	link#4	eth2
10.10.10.3	link#4	lo0
10.10.20.0/24	link#2	eth0
10.10.20.1	link#2	lo0
10.10.30.0/24	link#3	eth1
10.10.30.1	link#3	lo0
10.10.50.0/24	10.10.20.2	eth0
10.10.60.0/24	10.10.30.2	eth1
localhost	link#1	lo0

router2:

Odredište	Sljedeći skok	Mrežno sučelje
10.10.10.0/24	10.10.20.1	eth0
10.10.20.0/24	link#2	eth0
10.10.20.2	link#2	lo0
10.10.40.0/24	link#3	eth1
10.10.40.2	link#3	lo0
10.10.50.0/24	link#4	eth2
10.10.50.1	link#4	lo0
10.10.60.0/24	10.10.40.1	eth1
localhost	link#1	lo0

router3:

Odredište	Sljedeći skok	Mrežno sučelje
10.10.10.0/24	10.10.30.1	eth1
10.10.30.0/24	link#3	eth1
10.10.30.2	link#3	lo0
10.10.40.0/24	link#2	eth0
10.10.40.1	link#2	lo0
10.10.50.0/24	10.10.40.2	eth0
10.10.60.0/24	link#4	eth2
10.10.60.1	link#4	lo0
localhost	link#1	lo0

23. U emulatoru/simulatoru IMUNES konstruirajte mrežu koja sadrži tri podmreže povezane usmjeriteljima (Slika 3.16). Konfigurirajte statičko usmjeravanje (dakle, bez korištenja protokola za usmjeravanje) tako da dođe do petlje u usmjeravanju. Sučeljima računala i usmjeritelja dodijelite IP-adrese iz raspona 10.10.10.0/24, 10.10.20.0/24, 10.10.30.0/24, 10.10.40.0/24, 10.10.50.0/24 i 10.10.60.0/24. Ispišite tablice usmjeravanja svih računala i usmjeritelja. Pomoću alata Wireshark utvrdite što se tada događa s paketima koji "uđu" u petlju te komentirajte svoja zapažanja.

Tablice usmjeravanja računala su jednake kao u zadatku 22, a tablice usmjeravanja usmjeritelja su se promijenile jer smo morali stvoriti petlje u topologiji. Ovako izgledaju nove tablice usmjeravanja usmjeritelja:

router1:

Odredište	Sljedeći skok	Mrežno sučelje
10.10.10.0/24	link#4	eth2
10.10.10.3	link#4	lo0
10.10.20.0/24	link#2	eth0
10.10.20.1	link#2	lo0
10.10.30.0/24	link#3	eth1
10.10.30.1	link#3	lo0
10.10.50.0/24	10.10.30.2	eth0
10.10.60.0/24	10.10.20.2	eth1
localhost	link#1	lo0

router2:

Odredište	Sljedeći skok	Mrežno sučelje
10.10.10.0/24	10.10.40.1	eth0
10.10.20.0/24	link#2	eth0
10.10.20.2	link#2	lo0
10.10.40.0/24	link#3	eth1
10.10.40.2	link#3	lo0
10.10.50.0/24	link#4	eth2
10.10.50.1	link#4	lo0
10.10.60.0/24	10.10.20.1	eth1
localhost	link#1	lo0

router3:

Odredište	Sljedeći skok	Mrežno sučelje
10.10.10.0/24	10.10.40.2	eth1
10.10.30.0/24	link#3	eth1
10.10.30.2	link#3	lo0
10.10.40.0/24	link#2	eth0
10.10.40.1	link#2	lo0
10.10.50.0/24	10.10.30.1	eth0
10.10.60.0/24	link#4	eth2
10.10.60.1	link#4	lo0
localhost	link#1	lo0

Kada paketi uđu u petlju šalju se među usmjeriteljima u petlji dok im TTL ne istekne i tada budu odbačeni.

24. Započnite simulaciju i s računala pc provjerite dostupnost (naredba ping) poslužitelja server i očitajte TTL vrijednost iz ispisa. Zatim, s poslužitelja server provjerite dostupnost računala pc (naredba ping) i očitajte tu TTL vrijednost iz ispisa. Kojim putem idu paketi u jednom, a kojim putem u drugom slučaju? Ukratko objasnite zašto se međusobno razlikuju?

Kada sa računala pc pingamo server prvo dobivamo TTL=60, a nakon toga se povećava na TTL=61, a kada pingamo PC preko servera TTL=61. TTL se poveća na 61 jer se ažurira tablica servera i pronađe se brža ruta, te se preko te brže rute počinju slati podaci.

25. U emulatoru/simulatoru IMUNES, pomoću alata Wireshark snimite paket koji pripada protokolu RIP, proučite njegov sadržaj, te ga ukratko komentirajte.

Na računalu pc7(eth0) otvorio sam wireshark i pronašao paket koji pripada RIP protokolu. Sadržaj tog protokola je:

Command: Response (2) -> ovo je RIP odgovor

Version: RIPv2 (2) -> verzija protokola

IP Address: 10.0.0.0, Metric: 2

IP Address: 10.0.1.0, Metric: 3

IP Address: 10.0.2.0, Metric: 2

IP Address: 10.0.3.0, Metric: 2

IP Address: 10.0.4.0, Metric: 1

IP Address: 10.0.5.0, Metric: 1

IP Address: 10.0.6.0, Metric: 2

IP Address: 10.0.7.0, Metric: 3

IP Address: 10.0.8.0, Metric: 2

Paket koji pripada RIP protokolu sadrži odgovor u kojem nam govori metriku, tj. broj skokove potreban do svake mreže.

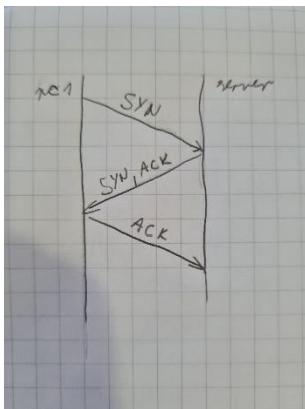
28. Učitajte u IMUNES mrežu Ping/ping.imn. Pomoću alata Wireshark snimite proizvoljan TCP-promet koji pripada jednoj vezi te odredite segmente koji se razmjenjuju u fazama uspostave veze i raskida veze (za generiranje TCP-prometa iskoristite alat netcat). a. Skicirajte razmjenu segmenata za te dvije faze, uz navođenje korištenih TCP-zastavica. b. Za uhvaćeni promet, odredite koje se adrese i vrata koriste na izvoru i odredištu. Imaju li svi segmenti istu četvorku {izvorišna IP-adresa, izvorišna vrata, odredišna IP-adresa, odredišna vrata}? c. Skicirajte razmjenu nekoliko TCP-segmenata u fazi trajanja veze, uz navođenje korištenih TCP-zastavica. d. Utvrdite na koji se način koriste potvrde u TCP-vezi. Komentirajte. e. Snimite proizvoljan TCP-promet (koji pripada jednoj vezi) i utvrdite veličine prozora. Mijenja li se veličina prozora često u tijeku trajanja TCP-veze? Objasnite.

Na serveru koristim naredbu nc -l 100

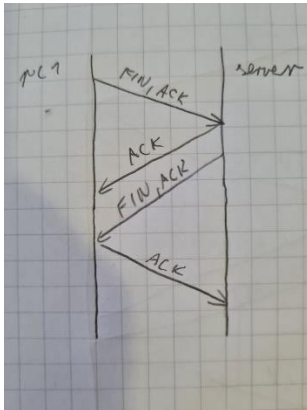
Na pc1 koristim naredbu nc 10.0.8.10 100

a)

uspostava veze



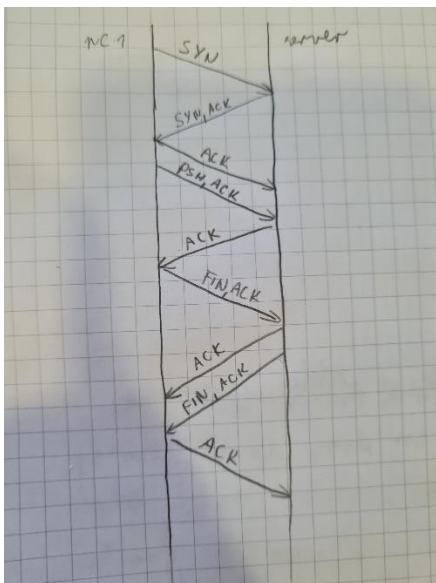
raskid veze



b)

pc1 šalje poruku serveru preko svog porta 32145 na port 100 servera, te s porta 100 prima TCP promet na port 32145. Dakle njihove adrese i portovi se ne mijenjaju.

c)



d)

Potvrde govore primatelju koliko je okteta primljeno

e)

Window size value: 1026
[Calculated window size: 65664]
[Window size scaling factor: 64]

Veličina prozora snimljenog TCP-prometa je 1026 sa faktorom od 64 i izračunatom vrijednosti prozora od 65664. Tijekom trajanja TCP-veze ne mijenja se izračunata vrijednost prozora.

29. Utvrdite mogu li se na jednom računalu pokrenuti dva procesa koji slušaju na istim vratima (npr., pokušajte dvaput pokrenuti alat netcat, istovremeno iz dvije konzole istog računala). Komentirajte.

Na istim vratima računalu preko jednog porta ne može slušati više procesa, ali ako računalu sluša na više različitih portova onda može slušati po jedan proces na svakom portu.

30. Ukoliko se alatu netcat ne zada protokol koji će koristiti, on podrazumijeva protokol TCP. Ponovite pokus iz prethodnog zadatka uz korištenje protokola UDP te komentirajte.

Kao i u zadatku 29. na istim vratima računala preko jednog porta nije moguće slušati više procesa istovremeno.

31. Primijetite da, iako su bitno različiti po svojstvima, protokoli UDP i TCP po funkcionalnosti spadaju u transportni sloj referentnog modela OSI. Objasnite zašto.

TCP i UDP su protokoli koji po funkcionalnosti spadaju u transportni sloj referentnog modela OSI jer služe za prijenos podataka između različitih računala. Razlika između tih protokola je to što oni svoju namjenu ostvaruju na različite načine.

32. Pokušajte prouzročiti gubitak TCP-segmenata. Na koji način možete utvrditi da je došlo do gubitaka? Možete li izazvati gubitke paketa bez mijenjanja karakteristika poveznica mreže?

Ako promijenimo veličinu prozora možemo utvrditi da je došlo do gubitaka bez mijenjanja karakteristika poveznica mreže.

33. Pokušajte identificirati promet koji pripada jednoj TCP-vezi za vrijeme u kojem dolazi do gubitka segmenata. Utvrdite što se tada događa s potvrdama i veličinom prozora.

Poslani segment je prevelik da ga prozor primatelja pohrani, te se segment odbacuje i javlja pogrešku pošiljatelju. Nakon toga segment se ponovno šalje, ali je još uvijek prevelik, te se ponovno ne pohranjuje.