Protokola TCP in UDP

Komunikacije v avtomatiki – Laboratorijske vaje

28. oktober 2013

Povzetek

V tej nalogi se boste ukvarjali za analizo protokolov TCP in UDP. Naloge so razdeljene v več segmentov, vsak segment vsebuje nekaj vprašanj. Na ta vprašanja odgovorite, in jih zapišite v dokument (npr., uporabite LibreOffice in njegov urejevalnik Writer). Da boste lažje argumentirali svoje odgovore, shranite slike Wiresharka (npr., s programom "ksnapshot", ki ga prikliče pritisk na tipko PrintScreen), iz katerih so razvidni podatki, ki ste jih uporabili za odgovor, in jih priložite v dokument. Nekateri deli te vaje črpajo iz knjige *Computer Networks – A top down approach* [2], ki je na voljo tudi v fakultetni knjižnici, ter prosojnic s predavanj prof. Stanislava Kovačiča. Snov za te vaje najdete v knjigi [2] v poglavjih 3.5 ter 3.7. Ker se določeni deli vaje dotikajo podrobnosti delovanja protokola, spodbujamo uporabo literature na spletu za iskanje odgovorov. S predavanj veste, da pakete TCP imenujemo segmenti, medtem, ko za pakete UDP uporabljamo uveljavljeno ime UDP datagrami. V nalogi bomo uporabljali to izrazoslovje.

1 Zajem TCP segmentov pri prenosu datoteke na strežnik

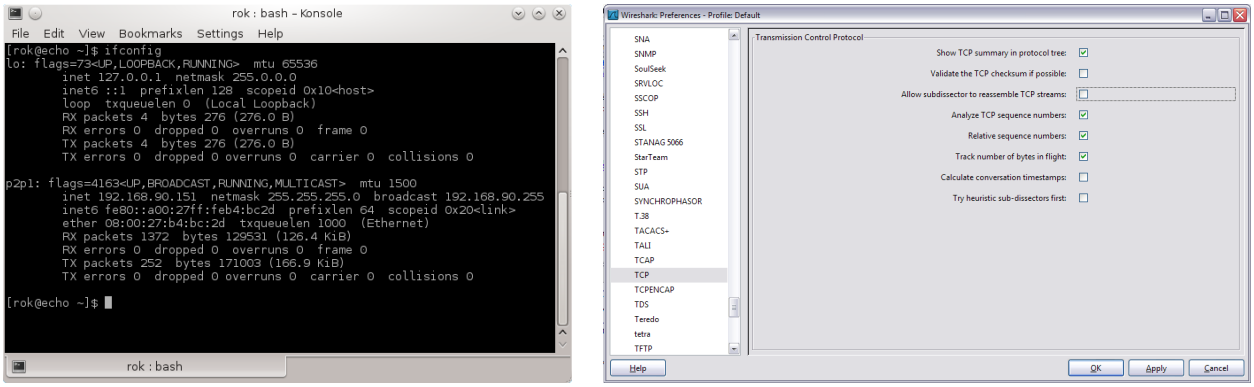
Preden začnemo z analizo protokola TCP, boste z Wiresharkom zajeli TCP segmente pri prenosu velike datoteke na oddaljeni strežnik. To boste storili preko spletne strani, ki smo jo postavili po vzoru [2] ravno za namen demonstracije delovanja protokola TCP. Spletna stran omogoča izbiro datoteke, ki se nahaja na vašem disku, ter pošiljanje te datoteke s protokolom HTTP na strežnik. Med pošiljanjem datoteke boste imeli aktiviran Wireshark, s katerim boste zajeli vse pakete, ki jih vaš računalnik pošlje in prejme.

Najprej ugotovite naslov IP klienta (vašega računalnika) ter naslov IP strežnika:

- Naslov IP vašega računalnika najlažje poiščete tako, da odprete linux terminal (npr., "konsole") in vnesete ukaz `ifconfig`, ki vam izpiše konfiguracijo vaših mrežnih vmesnikov (glej Sliko 1a). Med izpisanimi mrežnimi vmesniki izberite aktivnega (tistega, kateremu boste prisluškovali z Wiresharkom) in si prepišite njegov naslov IPv4.
- Preverite še naslov IP strežnika, na katerega boste dokument naložili. To storite z ukazom `ping`. V konzolo vpišite `ping -c 5 vision.fe.uni-lj.si`. Ob odgovoru s strežnika vam bo program izpisal njegov naslov IP. Tudi ta naslov si prepišite na list.

Sedaj boste zajeli pogovor med prenosom datoteke na strežnik:

- Zaženite spletni brskalnik, ter si z naslova `http://vision.fe.uni-lj.si/classes/KA-vaje/alice.txt` prenesite na disk ASCII kopijo *Alice v čudežni deželi*.
- Nato se postavite na spletno stran `http://vision.fe.uni-lj.si/classes/KA-vaje/vaje/v2srvr/TCPWS.html`, kjer se vam odprejo navodila za pošiljanje dokumenta.
- Uporabite gumb *Browse* za izbiro datoteke `alice.txt`. Pozor: Zaenkrat še ne pritisnite tipke *Prenesi*.
- Zdaj zaženite Wireshark in pričnite zajem podatkov z vašega mrežnega vmesnika (*Capture → Start, OK*).



(a) (b)

Slika 1: Primer iskanja lastnega naslova IP (a) in primer izklopa sestavljanja paketov (b).

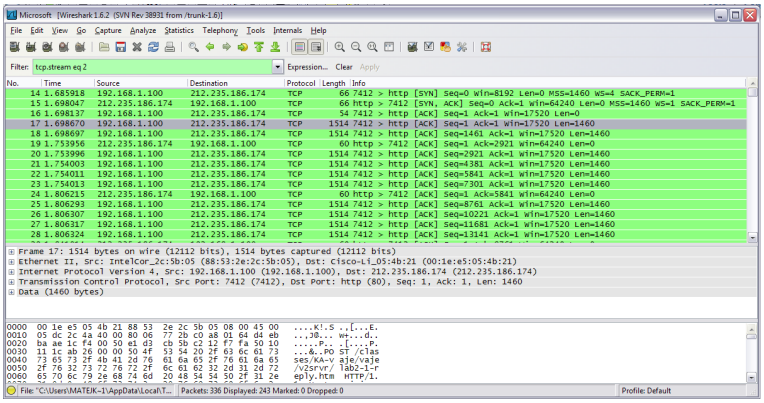
- Postavite se nazaj na spletno stran za prenos datoteke in pritisnite gumb *Prenesi*. Prikaže se vam kratko sporočilo, ki vam pove, da ste uspešno prenesli datoteko.
- Ustavite zajem podatkov v Wiresharku.
- Za vsak slučaj si shranite izpis na disk, da ga boste lahko obdelovali naprej doma, oziroma na naslednjih vajah. To storite preko *File* → *Save as*. Pozor: nekateri paketi so lahko v Wiresharku obarvani črno. Na teh paketih je bila detektirana napaka. Če je teh paketov preveč, potem je to znak, da Wireshark sestavlja pakete in narobe prikazuje njihovo dolžino. V tem primeru je vaš izpis lahko neuporaben za vajo. Preverite primernost vašega posnetka. Poiščite TCP segment, ki je bil poslan iz vašega IP naslova na IP naslov strežnika, in katerega značka `Len` ni enaka 0 (odčitate lahko kar iz okna izpisa paketov). Če je ta številka večja od 1460 bajtov, potem Wireshark sporoča napačno velikost paketa. V tem primeru si prenesite posnetek izpisa Wireshark iz `http://vision.fe.uni-lj.si/classes/KA-vaje/vaje/2012/WiresharkTCP_vision_klient_192.168.33.102.pcap` in uporabite ta posnetek za preostanek vaje. Posnetek naložite v Wireshark preko *File* → *Load* ali tako, da datoteko ročno prenesete v Wireshark (angl., drag-and-drop). Vidite, da so podatki v najdaljših segmentih v tem posnetku krajši od 1460 bajtov. Razlog je v tem, da mrežna kartica, preko katere smo posneli dostop, omejevala velikost paketa IP na 1500 bajtov. Ker je glava IP dolga 40 bajtov, ostane za celoten TCP segment (glava+podatki) 1460 bajtov. Naslov IP klienta v tem primeru je bil `192.168.33.102`.
- Čestitamo! Sedaj imate vse pripravljeno za začetek vaje.

2 Visokonivojski pogled na TCP (10%)

Preden se posvetimo podrobnostim protokola TCP, si poglejmo celotno sliko prenesenih paketov. Najprej izklopimo prikaz dodatnih paketov TCP, ki jih Wireshark tvori umetno, ko sestavi večje zaporedje krajših paketov (angl., reassembling). To storite tako, da preko zavihka *Edit* → *Prefereces* → *Protocols* → *TCP* izklopite *Allow subdissector to reasemble* (glej Sliko 1b). Tukaj je na voljo več nastavitev. Opazite lahko *Analyze TCP sequence numbers* in *Relative sequence numbers*. Te bomo pustili odkljukane, zavedati pa se morate, da s temi nastavitvami Wireshark reinterpretira značke `Seq` v TCP segmentih in jih *prikaže* od 0 dalje – po protokolu TCP je zares vrednost `Seq` v prvem segmentu nastavljena na naključno vrednost!

Ker poznate naslov IP vašega računalnika in strežnika, poiščite prvi paket, ki je bil poslan z vašega računalnika na strežnik. Sedaj preglejte celotni pogovor tako, da z desnim gumbom na miški kliknete na paket, nato pa izberete *Follow TCP stream*. Vrnite se nazaj v glavno okno Wiresharka, kjer se v vrstici za filtriranje paketkov nahaja izraz podoben "`tcp.stream eq 2`", kar pomeni, da Wireshark prikazuje samo paketke TCP, ki ustrezajo pogovoru med vašim računalniku in strežnikom.

Prvi trije paketki TCP ustrezajo trosmernem rokovanju (angl., three-way handshake) med vašim računalnikom in strežnikom. S tem se vzpostavi TCP seja, segmente pa prepoznate po



Slika 2: Primer izpisa v Wiresharku.

značkah `[SYN]` ter `[ACK]`. Prvim trem segmentom sledi paket, ki je tipa HTTP. Zavedati se morate, da vaš računalnik ni zares poslal celega paketa HTTP *naenkrat*, pač pa ga je razdrobil v manjše TCP segmente in te poslal enega za drugim na strežnik, kjer jih je ta sestavil nazaj v sporočilo HTTP. Odgovorite na spodnja vprašanja, in v dokument z odgovori dodajte primerne slike iz Wiresharka:

1. Navedite naslov IP ter številko TCP vrat (angl., port), ki jih uporablja klient (vaš računalnik) za prenos paketov na strežnik *vision.fe.uni-lj.si*. Na to vprašanje boste najlažje odgovorili tako, da izberete prvi paket HTTP, nato pa v srednjem oknu Wiresharka preko zavihkov poiščete ustrezno informacijo.
2. Navedite naslov IP strežnika *vision.fe.uni-lj.si*, ter na katerih vratih strežnik oddaja in prejema TCP segmente pri vašem prenosu.

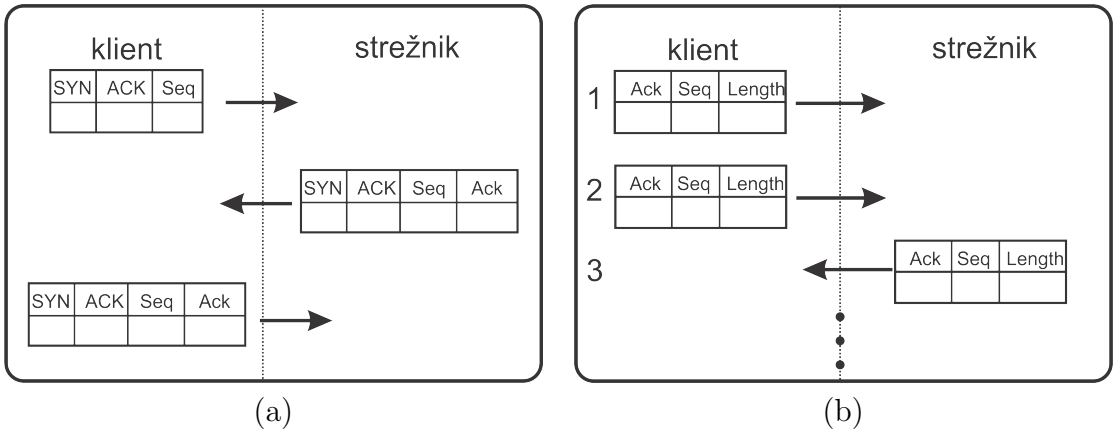
Ker se bodo vaje v naslednjem poglavju nanašale zgolj na prenosni sloj, bomo odklopili prikaz paketov HTTP (aplikacijski sloj) v Wiresharku. To storite tako, da izberete *Analyze* → *Enabled Protocols*, izbrišete kljukico pri protokolu HTTP, ter pritisnete OK. Vaš izpis v Wiresharku bi moral izgledati kot v Sliki 2.

3 Osnove protokola TCP (50%)

V tem delu si boste natančneje ogledali protokol TCP. Za kratek pregled glavnih elementov protokola si boste prebrali spletno poglavje [1]. V prvem delu se bomo ukvarjali z vzpostavitvijo TCP seje, torej s trosmernim rokovanjem. Zato si najprej preberite [1] [1] do vključno poglavja "*Establishing a connection*" – pri branju tega poglavja bodite pozorni na princip, kako se med trosmernim rokovanjem v paketih TCP postavljajo zastavice `ACK`, `SYN` in kako se določajo vrednosti sekvence `Seq` in potrditvene številke `Ack`. Kratek odstavek na temo trostranskega rokovanja najdete na Wikipediji `http://en.wikipedia.org/wiki/Three-way_handshake#Connection_establishment`. Zgornjo literaturo najprej preberite na hitro, potem pa se vračajte nazaj nanjo ob iskanju odgovorov na spodnja vprašanja.

1. Slika 3a prikazuje skico prvih treh paketov, ki si jih izmenjata klient in strežnik ob vzpostavljanju TCP seje. Za vse tri segmente zapišite vrednosti `SYN`, `ACK`, `Ack` in `Seq`, ki jih odčitate iz vašega primera v Wiresharku. Na kratko razložite kako se vse tri vrednosti izračunavajo.
   - Kako se nastavi/izračuna vrednost `Seq` pri prvem paketu?
   - Kako se nastavi/izračuna vrednost `Ack` pri drugem paketu?
   - Kako se nastavi/izračuna vrednost `Ack` pri tretjem paketu?
2. V Wiresharku poiščite in prikažite kje se v paketku vidi nastavitev zastavic `SYN` in `ACK`.

[1] `http://www.cs.rpi.edu/academics/courses/spring06/netprog/c06.html`



Slika 3: Skica prvih treh paketov pri vzpostavljanju seje (a), ter skica paketov pri prenosu vsebine HTTP ukazov in podatkov (b).

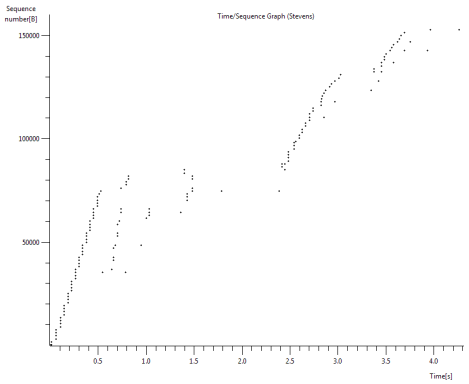
3. Postavite se na prvi paket v trosmernem rokovanju. V osrednjem oknu Wiresharka se postavite na zavihek za prenosni sloj, in preko podzavihkov poiščite značko *Maximum segment size*. Koliko znaša vrednost te značke in kaj značka pomeni? Poglejte vrednost te značke v naslednjem segmentu, ki ga strežnik pošlje nazaj klientu. Ali je vrednost enaka? Komentirajte svoj odgovor. Teorijo si lahko preberete v poglavju *Establishing a connection* [1] in `http://www.daemon.org/tcp.html`.
4. Postavite se na ustrezen paket, ki ga klient pošilja strežniku, in poiščite značko *Window size value*. V poglavju *Preventing a sender from overloading a receiver* [1] preberite, kaj pomeni ta značka in razložite njen namen, njen pomen in kaj pomeni njena vrednost v vašem primeru.
5. Med TCP segmenti poiščite prvega, ki nosi ukaze HTTP (t.j., `POST /classes/vaje...`) in se postavite nanj. Obravnavajte ta segment kot prvi paket. Sedaj si skicirajte zaporedje pošiljanja prvih šestih segmentov kakor je skicirano v Sliki 3b – pozor: vaše zaporedje segmentov je lahko precej drugače, kot tisto skicirano v sliki. Pri svoji skici izpolnite polja z značkami `Ack`, `Seq`, `Length`.
6. Razložite, kaj pomenijo značke `Seq` in `Length`. Za razlago značk si lahko pomagate z dokumentom na spletni strani `http://www.daemon.org/tcp.html`.
7. Med skiciranimi paketki poiščite potrditveni paket, ki ga strežnik pošlje v potrditev prvega klientovega paketa. S primerjanjem vrednosti `Length` klientovega paketa in vrednosti `Ack` v potrditvenem paketu razložite kako se določajo/izračunajo v zaporedju paketov vrednosti `Ack`. Teorijo si preberite pod *Ensuring Reliable Transport* v [1].
8. Še vedno sklicujoč se na vaš dokument, kako se v tretjem paketu, ki ga pošlje klient, izračuna številka pod značko `Seq`?

4 Protokol TCP in kontrola pretoka (10%)

Oglejmo si še prenos podatkov na časovno enoto s klienta na strežnik. Pretok si preprosto izrišemo z Wiresharkovim orodjem *Time-Sequence-Graph(Stevens)*.

1. V spisku paketov v Wiresharku označite paket, ki prenaša podatke iz klienta na strežnik. Nato izberite *Statistics* → *TCP Stream Graph* → *Time-Sequence-Graph(Stevens)*. Videti bi morali graf, kakršnega prikazuje Slika 4.
2. Vaš graf si skopirajte v dokument z odgovori.
3. Kaj pomeni izraz "kontrola pretoka" (angl. congestion control) pri protokolu TCP in kaj je njen namen?

Dodatna naloga 1 (dodatnih 5%):
Kako deluje napad "denial-of-service" in katero lastnost protokola TCP izkorišča?



Slika 4: Primer grafa hitrosti prenosa podatkov. Vsaka pika na grafu predstavlja TCP segment in prikazuje zaporedno številko segmenta glede na čas oddaje. Pike, ki so "zložene" ena nad drugo prikazujejo sekvenco paketov, ki jih je klient poslal neposredno enega za drugim (angl., back-to-back).

5 Protokol UDP (30%)

Sedaj si oglejmo še delovanje protokola UDP v praksi. Zaženite zajem podatkov z Wiresharkom in izvedite operacijo, ki bo povzročila izmenjavo paketov UDP (datagramov). Odprite linux terminal (npr., "konsole") in vpišite ukaz "`nslookup en.wikipedia.org`", ki bo sprožil poizvedbo DNS. Poizvedbo prestrezite v Wiresharku. Ustavite zajem podatkov, ter nastavite filter v Wiresharku tako, da bo prikazoval samo UDP datagrame. Izberite enega od datagramov in v srednjem oknu Wiresharka razširite zavihek, ki prikazuje podrobnosti paketa.

1. Iz izbranega datagrama določite, koliko polj vsebuje glava paketa. Brez da bi gledali v specifikacije protokola, naštejte imena teh polj in priložite sliko izris (angl., screenshot), ki prikazuje te podatke.
2. S pomočjo najnižjega okna v Wiresharku, ki prikazuje paket v heksadecimalnem zapisu, za vsako polje v glavi datagrama povejte, koliko bajtov porabi. Priložite izris, s katerim prikažete, kako ste določili/odčitali dolžino polj.
3. Dolžino česa predstavlja vrednost polja `Length`? Preverite vašo hipotezo v Wiresharku in prikažite izpis, s katerim potrdite hipotezo.
4. Če veste koliko bajtov je namenjeno zapisu polja `Length`, kolikšna je najvišja možna številka, ki jo lahko zapišemo v polje `Length`?
5. V prejšnji alineji ste izračunali največjo možno dolžino, ki jo lahko zapišemo v polje `Length`. Sedaj na podlagi te številke in vašega vedenja, koliko bitov porabimo za zapis glave UDP datagrama, izračunajte največjo možno dolžino podatkov, ki jih teoretično lahko nosi UDP datagram. Številka, ki jo dobite, velja za datagrame, ki jih prenašamo preko IPv6[2].
6. Preverite, koliko bajtov je namenjenih za zapis izhodnega porta v UDP datagramu. Iz tega izračunajte najvišjo možno številko porta.
7. Kakšna je številka protokola za UDP? Ker mora paket IP (mrežni sloj) vedeti, ali prenaša TCP segment ali UDP datagram, se številka protokola nahaja v glavi paketa IP. V Wiresharku odprite zavihek s podatki o paketu IP in odčitajte številko protokola. To številko izpišite v decimalni ter heksadecimalni obliki. Na enak način preverite še številko protokola TCP. V obeh primerih priložite izris iz Wiresharka.
8. Poglejte na stran `http://www.netfor2.com/udpsum.htm`, ali katero drugo, in povejte preko katerih polj se izračuna vrednost polja `checksum`.
9. Izberite si paket, ki vsebuje DNS poizvedbo po imenu *en.wikipedia.org*. Izrišite si podrobnosti paketa, ki vsebujejo izvorno številko vrat (angl., port number) in ponorno številko vrat. Prav tako si izrišite sliko podrobnosti paketa, ki pride kot odgovor na to poizvedbo. Razložite povezavo med številkami vrat, ki jih vidite v obeh paketih.

[2]Dolžina UDP datagrama je manjša pri prenosu preko protokola IPv4.

Dodatna naloga 2 (dodatnih 10%):
Zajemite čim manjši paket tipa UDP. Na roke preračunajte vrednost njegovega polja `checksum`. Prikažite postopek in razložite korake pri računu.

Literatura

[1] R. Ingalls, *Csci.4220 network programming: The internet transport layer, tcp and udp*, http://www.cs.rpi.edu/academics/courses/spring06/netprog/c06.html, 2006.

[2] J.F. Kurose and K.W Ross, *Computer networking – a top-down approach*, Addison Wesley, 2009.