Analiza načina udaljenog pristupa računalnim mrežama

Slade, Miho

Undergraduate thesis / Završni rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Transport and Traffic Sciences / Sveučilište u Zagrebu, Fakultet prometnih znanosti**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:119:855732

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-28**

Repository / Repozitorij:

Faculty of Transport and Traffic Sciences - Institutional Repository

SVEUČILIŠTE U ZAGREBU

FAKULTET PROMETNIH ZNANOSTI

Miho Slade

ANALIZA NAČINA UDALJENOG PRISTUPA RAČUNALNIM MREŽAMA

ZAVRŠNI RAD

Zagreb, 2017.

Sveučilište u Zagrebu

Fakultet prometnih znanosti

ZAVRŠNI RAD

ANALIZA NAČINA UDALJENOG PRISTUPA RAČUNALNIM MREŽAMA

ANALYSIS OF REMOTE ACCESS TO COMPUTER NETWORKS

Mentor: doc. dr. sc. Ivan Grgurević

Student: Miho Slade, 0135237599

Zagreb, rujan 2017.

Analiza načina udaljenog pristupa računalnim mrežama

SAŽETAK

U završnom radu objašnjeni su načini udaljenog pristupa u računalnim mrežama. Rad započinje s uvodnim predstavljanjem osnovnih značajki računalnih mreža, njihovim razvojem i podjelom na vrste prema različitim kriterijima. Nakon toga, opisane su dvije temeljne arhitekture mreža i standardni protokolni složaj korišten u komunikaciji dvaju računala. Zatim slijedi aspekt sigurnosti u mrežama. Kroz model informacijske sigurnosti nastoje se predstaviti zahtjevi koje računalna komunikacija mora ispuniti da bi se smatrala vjerodostojnom i sigurnom. Potom je objašnjena kriptografija te metode, postupci i protokoli u službi sigurnosti. Nakon toga, čitatelj bi trebao steći teoretsku podlogu potrebnu za razumijevanje načina udaljenog pristupa koji su temelj ovog završnog rada. Načini udaljenog pristupa VNC, VPN, SSH definiraju se te se zatim obavlja pregled i testiranje odabranih programskih alata i rješenja. Naposljetku se provodi usporedna analiza na osnovu iskustva testiranja aplikacija. U svrhu usporedne analize uspostavljeni su relevantni parametri za usporedbu, npr. vrsta sučelja, interoperabilnost, korišteni protokoli, sigurnost, arhitektura, itd. Sukladno tome provodena je analiza te su utvrđene prednosti, nedostatci, sličnosti i razlike načina i programskih alata za udaljeni pristup.

KLJUČNE RIJEČI: računalne mreže, Internet, računalna sigurnost, protokoli, udaljeni pristup

SUMMARY

The final paper explains methods of remote access in computer networks. It begins with introduction of basic computer network features, their development and division into types according to different criteria. Subsequently, the paper describes two basic network architectures and a standard protocol arrangement that is used in communication between computers. Afterwards it follows the security aspects in computer networks. With an information security model, the paper seeks to present the requirements that computer communications must meet to be considered credible and safe. It continues with cryptography and methods, procedures and protocols in the service of security. After that, the reader should acquire the theoretical background needed to understand the remote access methods that are the basis of this final paper. Remote access methods VNC, VPN, SSH are defined and then selected remote access software tools and solutions are reviewed and tested. Finally, a comparative analysis is carried out based on the experience of application testing. For this purpose, relevant parameters for comparison are established, such as interface type, interoperability, used protocols, security, architecture, etc. According

to the parameteres, the paper analyzes and finally determines advantages, disadvantages, similarities and differences of methods and program tools for remote access.

KEY WORDS: computer networks, Internet, cybersecurity, protocols, remote access

Sadržaj

1. Uvod 1
2. Značajke računalnih mreža 3
2.1 Vrste računalnih mreža 3
2.2 Internet, intranet, extranet 4
2.4 Arhitektura računalnih mreža 5
2.5 OSI model 5
3. Sigurnosni kriteriji u funkciji računalnih mreža 8
3.1 Model CIA 8
3.2 Kriptografija 9
3.4 Simetrična kriptografija 10
3.5 Asimetrična kriptografija 12
3.6 Hash funkcije 13
3.7 Digitalni potpis 15
3.9 IPsec 16
3.10 Firewall 19
4. Načini udaljenog pristupa računalnoj mreži (VNC, VPN, SSH) 20
4.1 Virtual Network Computing (VNC) 20
4.2 Remote Desktop Protocol (RDP) 22
4.3 Secure Shell (SSH) 23
4.4 Virtual Private Network (VPN) 26
5. Pregled i testiranje načina udaljenog pristupa 29
5.1 RealVNC *Enterprise* 29
5.2 Remote Desktop Connection 32
5.3 TeamViewer 34
5.4 FreeSSHd 37
5.5 PuTTy 39
5.6 Tera Term 44
5.7 TTY emulator 48
5.8 Vypr VPN 49
5.9 TunnelBear 52
5.10 Hide.me 54
6. Usporedna analiza načina udaljenog pristupa računalnim mrežama 56
6.1 Usporedna analiza *remote desktop* programskih alata 58

6.2 Usporedna analiza SSH programskih alata 59
6.3 Usporedna analiza VPN programskih alata 60
7. Zaključak 62
Literatura 63
Popis kratica i akronima 67
Popis slika 72
Popis tablica 73

1. Uvod

Informacija je centralna vrijednost današnjeg modernog društva. Imperativ svakog korisnika postojećih tehnologija je brzo i efikasno dohvatiti potrebnu informaciju. Gubitak iste ili nemogućnost pristupa može izazvati štetu koja je nerijetko značajna, posebno u poslovnom okruženju. Korisnici često nemaju fizički pristup računalima odnosno izvorima informacija. Međutim, računala su povezana u mreže te postoje načini za pristupanje istima. Razlog postojanja potrebe za udaljenim pristupom može biti pružanje tehničke podrške, administracija i konfiguriranje računala, ostvarenje sigurne komunikacije i dr. Računalu se može pristupiti na tri načina: pristup konzolnom sučelju operacijskog sustava, pristup radnoj površini preko grafičkog sučelja te pristup korištenjem posredničkog poslužitelja. Bitna karakteristika udaljenog pristupa je da se komunikacija odvija javnom mrežom poput Interneta te da npr. razmjena datoteka u lokalnoj mreži ne može biti smatrana udaljenim pristupom. Pred javnu mrežu zato se postavljaju zahtjevi koje treba ispuniti u vidu sigurnosti, pouzdanosti i raspoloživosti. Za potrebe rada identificirani su standardni načini udaljenog pristupa. Cilj je objasniti princip rada tih načina i tehnologije koje stoje iza njih, testirati popularna programska rješenja te im dokazati praktičnost primjene u privatnom i poslovnom životu. Potom se nastoji odrediti parametre kojima se može kvalitetno usporediti funkcionalnosti koje pružaju načini i programi za udaljeni pristup. Svrha završnog rada je analizom podataka u tablicama usporedbe uočiti bitne razlike u načinima udaljenog pristupa te potom doći do objektivno najboljeg programskog rješenja za svaki od načina u skladu s općim korisničkim zahtjevima.

Tematika rada razložena je u sedam poglavlja:

1. Uvod
2. Značajke računalnih mreža
3. Sigurnosni kriteriji u funkciji računalnih mreža
4. Načini udaljenog pristupa računalnoj mreži (VNC, VPN, SSH)
5. Pregled i testiranje načina udaljenog pristupa
6. Usporedna analiza načina udaljenog pristupa računalnim mrežama
7. Zaključak

Uvodno poglavlje daje osnovnu sliku o radu te definira cilj i strukturu rada.

U drugom poglavlju dane su definicije potrebne za razumijevanje osnovnih pojmova vezanih za računalne mreže. Izvedene su podjele mreža s obzirom na

različite kriterije te objašnjene dominantne arhitekture: klijent-server i *peer to peer*. Na kraju poglavlja objašnjen je OSI model kao temelj za razumijevanje mrežne komunikacije.

U trećem poglavlju fokus je na sigurnosnom aspektu u računalnim mrežama. Predstavljene su kriptografske metode i postupci koji se implementiraju u svrhu postizanja sposobnosti ispunjenja sigurnosnih zahtjeva povjerljivosti, integriteta, raspoloživosti i neporecivosti. Kao ključne teme ovog poglavlja ističu se simetrična i asimetrična kriptografija, *hash* funkcije, digitalno potpisivanje i IPsec protokol.

U četvrtom poglavlju, ulazi se u glavnu temu rada. Proučeni su načini za udaljeni pristup u računalnim mrežama: VNC, VPN i SSH. Za svaki od načina objašnjeni su principi rada, protokoli i mogućnosti.

Peto poglavlje namijenjeno je pregledu i testiranju dostupnih programskih rješenja za načine udaljenog pristupa. Predstavljene su značajke svih rješenja te dane upute za korisnu primjenu u stvarnom životu uz brojne ilustracije.

U šestom poglavlju se kroz komparativnu analizu nastoje utvrditi prednosti i nedostatci svakog od načina udaljenog pristupa ovisno o definiranim parametrima.

U zaključnom pogavlju, na temelju činjenica iznesenih u prethodnim poglavljima i izvršene usporedne analize, dan je autorov komentar koji objedinjuje sve prikupljene spoznaje.

2. Značajke računalnih mreža

Računalna mreža je sustav koji je nastao povezivanjem dva i više računala. Za dva računala se kaže da su povezana ako mogu međusobno komunicirati. Svrha ovog umrežavanja je omogućiti dijeljenje resursa i podataka te stvaranje distribuirane obrade podataka [1].

Razvoj računalnih mreža uvjetovala je potreba za razmjenom velike količine podataka (računalnih programa, multimedijskog sadržaja i slično) između korisnika računala. Prvotno se razmjena obavljala snimanjem sadržaja na CD i DVD medije, ali izostaje način kojim bi se sadržaj prebacio na veće udaljenosti u kratkom vremenu [1].

Zbog toga se počinje s povezivanjem računala u mreže. Računala u mreži nazivaju se čvorovima te su međusobno spojena kanalima koji predstavljaju odgovarajući prijenosni medij putem kojeg se obavlja prijenos informacija (bakar, optika, zrak) [1].

2.1 Vrste računalnih mreža

Postoji više različitih vrsta računalnih mreža. Dijele se prema svrsi i veličini. U smislu svrhe, mnoge se mreže mogu smatrati mrežama opće svrhe tj. služe za sve, od slanja datoteka do pristupa Internetu. Međutim, neke vrste mreža služe posebnoj svrsi, naprimjer:

- SAN (engl. *Storage Area Network*) - mreža za spajanje računala na spremišta podataka,
- WLAN (engl. *Wireless LAN*) – bežična lokalna mreža,
- EPN (engl. *Enterprise Private Network*) – privatna mreža tvrtke i
- VPN (engl. *Virtual Private Network*) – virtualna privatna mreža (o ovom vrsti umrežavanja biti će više riječi u sljedećim poglavljima).

Prema veličine računalne mreže dijele se na:

- PAN (engl. *Personal Area Network*),
- LAN (engl. *Local Area Network*),
- MAN (engl. *Metropolitan Area Network*) i
- WAN (engl. *Wide Area Network*).

PAN ili osobna mreža je mreža organizirana oko pojedinca unutar jedne građevine (mali ured ili rezidencija). U PAN su povezani jedan ili više terminalnih uređaja (računala, telefona, televizora) te se mreža prostire na svega nekoliko metara. Ukoliko više pojedinaca koristi istu mrežu unutar rezidencije, PAN mrežu se može nazvati HAN (engl. *Home Area Network*, kućna mreža) [2].

Lokalna mreža (LAN) je računalna mreža u kojoj su računala smještena na manjim udaljenostima (unutar doma, ureda, ili blisko smještenih zgrada). Značajka lokalnih mreža je da su one najčešće u cijelosti u vlasništvu i pod upravljanjem onih koji ih koriste (osobe, tvrtke, institucije). LAN je vrlo koristan za dijeljenje resursa, može se izgraditi relativno jeftinim hardverskim elementima kao što su *hub*[1], mrežni adapter i *Ethernet* kabel. Najmanji LAN može koristiti samo dva računala, a u veće LAN mreže mogu biti povezane tisuće računala. LAN se uglavnom oslanja na žične veze (mogu biti i bežične), a karakteristike LAN-a su velika brzine prijenosa, mala vjerojatnost greške i poboljšana sigurnost [2], [3].

Metropolitanska mreža (MAN) značajno je veća od LAN mreže. MAN obuhvaća šire područje poput grada, kampusa ili manje regije. MAN se često koristi kako bi se povezalo više LAN mreža u jednu veliku mrežnu cjelinu. Kada je ova mreža posebno dizajnirana za kampus naziva se i CAN (engl. *Campus Area Network*) [2].

WAN mreža zauzima još veće područje poput države ili svijeta. Sastoji se od više manjih mreža (MAN i LAN). Primjer WAN mreže je globalna računalna mreža Internet [2].

2.2 Internet, intranet, extranet

Internet je svjetska mreža računala dostupna svima koji poznaju IP adresu odredišnog računala na koji se žele spojiti. IP adresa je jedinstveni skup brojeva koji definiraju lokaciju računala. Međutim, najčešće korisnici u *browser* unose simbolički naziv željenog poslužitelja. Prije nego što se može pristupiti imenovanom računalu ime treba prevesti u odgovarajuću IP adresu. Korisnikov preglednik pristupat će DNS (engl. *Domaine Name System*) poslužitelju kako bi pronašao uneseni naziv i vratio IP adresu odredišta [4].

Intranet je privatna mreža poduzeća, ustanove ili organizacije. Mreža se sastoji od ograničenog broja računala koja su međusobno povezana i kontrolirana na definiran način. Intranet postavlja organizacija kako bi osigurala sigurnu i neometanu vezu između svojih članova te efikasnije dijeljenje informacija [4].

Extranet je dio intraneta te se također smatra privatnom mrežom. Extranetom upravlja ista organizacija. Cilj extraneta je pružiti ovlaštenim subjektima iz vanjskog svijeta siguran pristup Intranetu [4].

[1] *Hub* je centralni uređaj za povezivanje računala u zvjezdastu topologiju (način povezivanja računala u lokalnim mrežama u obliku zvijezde sa središnjim čvorom) i dijeli se na aktivne i pasivne *hubove*. Pasivni *hub* ni na koji način ne obrađuje podatake. Aktivni *hub* obnavlja signal i održava potrebnu snagu signala. Neki *hubovi* također mogu preuzeti ulogu mrežnih mostova, usmjerivača ili skretnica.

2.4 Arhitektura računalnih mreža

U svakodnevnom korištenju terminalnih uređaja odnosno računala spojenih na mrežu mogu se prepoznati dvije arhitekture: klijentsko-serverska i *peer to peer.*

Klijent-server je oblik arhitekture koji je često korišten. Poslužitelj (engl. *server*) je moćno računalo jakih performansi kojemu je zadaća pohrana podataka i osiguranje određene usluge računalima spojenima na njega. Klijent je korisnikovo računalo (terminalni uređaj) koje korisniku omogućuje pristup podacima na udaljenom poslužitelju [5].

Kod klijentsko-serverske arhitekture klijent šalje poruku zahtjeva poslužitelju. Kada poslužitelj primi klijentov zahtjev, on će u svojoj memoriji potražiti zahtjevane podatke te ih poslati klijentu u poruci odgovora [5].

U slučaju da jedan poslužitelj pruža više usluga velikom broju korisnika može doći do tzv. *bottlenecka* odnosno smanjenja efikasnosti pružanja usluge zbog zagušenja zahtjevima [5].

Za razliku od klijentsko-serverskog modela, *peer to peer* model ne razlikuje čvorove (računala) u mreži kao klijenta ili *servera*. Svako se računalo ponaša i kao klijent i kao *server* ovisno o tome da li u nekom trenutku šalje poruke zahtjeva ili odgovora [5].

Da bi čvor postao dio *peer to peer* arhitekture prvo se mora priključiti mreži. Nakon priključenja u *peer to peer* sustav čvor može drugim čvorovima slati zahtjeve i odgovore [5].

Dva su načina utvrđivanja koji čvor pruža koje usluge, prema izvoru [5]:

1. Kada čvor ulazi u *peer-to-peer* sustav, mora registrirati usluge koje će pružati. Registracija se obavlja u centralnom serveru na mreži. Kada čvor zahtjeva neku uslugu, mora kontaktirati centralni *server* kako bi provjerio koji čvor će pružiti željenu uslugu. Ostatak komunikacije obavlja se između čvor koji zahtjeva i čvora koji pruža uslugu.

2. Čvor koji želi određene usluge mora emitirati zahtjev za uslugama svim ostalim čvorovima u *peer-to-peer* sustavu. Čvor koji pruža traženu uslugu poslat će odgovor čvoru koji zahtjeva uslugu.

2.5 OSI model

Međunarodna organizacija za standardizaciju (engl. *International Organization for Standardization*, ISO) razvila je OSI (engl. *Open Systems Interconnection*) model. OSI model dijeli mrežnu komunikaciju u sedam slojeva [6]:

7. Aplikacijski sloj,

6. Prezentacijski sloj,

5. Sesijski sloj,

4. Transportni sloj,

3. Mrežni sloj,

2. Podatkovni sloj i

1. Fizički sloj.

Niži slojevi (od sloja 1 do sloja 4) brinu se za prijenos podataka dok viši slojevi (od sloja 5 do sloja 7) sadrže podatke na razini aplikacije. Komunikacija između slojeva funkcionira na jednostavnom principu. Svaki sloj nakon obavljanja svoje zadaće podatke prosljeđuje sljedećem po redu sloju [7].

Smišljeno projektirane procedure koje prate razmjenu podataka nazivaju se protokoli. Protokol se realizira u vidu procesa koji se treba obaviti da bi se očuvao integritet prijenosa podataka. Svaki proces obavlja se na jednom od slojeva [6].

Fizički sloj definira električke, mehaničke, funkcionalne i proceduralne specifikacije za aktivaciju, održavanje, deaktivaciju fizičkog *linka* između krajnjih sustava. *Fast Ethernet*[2], RS232[3], ATM[4] neki su od protokola ovoga sloja [7].

Podatkovni sloj podatke dobivene od fizičkog sloja provjerava kako bi utvrdio da li su se dogodile pogreške u prijenosu. Bitove podataka formira u protokolne podatkovne jedinice okvire. Podatkovni sloj također upravlja shemama fizičkog adresiranja kao što su MAC adrese za *Ethernet* mreže, kontrolirajući pristup bilo kojim mrežnim uređajima na fizički medij. Budući da je podatkovni sloj jedan od najsloženijih slojeva u OSI modelu, često je podijeljen u dva dijela, podsloj *Media Access Control* i podsloj *Logical Link Control* [8].

Za mrežni sloj vežu se funkcije rutiranja i logičkog adresiranja. Kada prethodno formirani okvir dospije u mrežni sloj vrši se daljnji postupak enkapsulacije. Okvirima podatkovnog sloja dodaje se zaglavlje mrežnog sloja te se tako formiraju protokolne podatkovne jedinice mrežnog sloja – paketi. U zaglavlju paketa nalazi se IP (engl. *Internet Protocol*) adresa izvorišta i odredišta. Na temelju toga vrši se usmjeravanje paketa najoptimalnijim putem kroz mrežu [7].

U transportnom sloju, paketima se dodaje zaglavlje transportnog sloja, stvaraju se protokolne podatkovne jedinice transportnog sloja – segmenti. Adresiraju se *portovi*

[2] U računalnom umrežavanju, Fast Ethernet je skupni pojam za niz Ethernet standarda koji nose promet pri nominalnoj brzini od 100 Mbit / s

[3] Standard RS232 obično se koristi u računalnim serijskim priključcima.

[4] ATM (engl. *Asynchronous Transfer Mode*) tehnika je prijenosa u telekomunikacijama koja se zasniva na asinkronom vremenskom multipleksiranju

kako bi se mogli razlikovati procesi na krajnjem računalu. Adresa izvorišnog i odredišnog *porta* sadržana je u zaglavlju segmenta. Ovaj sloj omogućuje pouzdan i transparentan prijenos, kontrolu toka od kraja do kraja i kontrolu pogrešaka od kraja do kraja. Najvažniji protokoli ovog sloja su TCP (engl. *Transmission Control Protocol*) za spojni pouzdani prijenos i UDP (engl. *User Datagram Protocol*) za nespojni nepouzdani prijenos[5] [7].

Sesijski sloj uspostavlja, upravlja i raskida konekciju između dva računala koja međusobno komuniciraju. Sinkronizira dijalog između prezentacijskih slojeva različitih računala, upravlja transferom podataka, omogućuje kakvoću usluge te javlja o problemima u sesijskom, prezentacijskom i aplikacijskom sloju [7].

Prezentacijski sloj omogućuje da poruka generirana od aplikacijskog sloja jednog računala bude razumljiva aplikacijskom sloju drugog računala. Prevodi u neki zajednički format. Primjeri protokola su PICT, JPEG, MPEG[6] [7].

Aplikacijski sloj je jedinstven po tome što iznad njega nema višeg sloja. Najbliži je korisniku. Aplikacijski sloj uspostavlja dostupnost između komunikacijskih partnera, sinkronizira dogovore za oporavak u slučaju pogrešaka i kontrolira integritet podataka. Neki od protokola aplikacijskog sloja su DNS (engl. *Domaine Name System*)[7], SMTP (engl. *Simple Mail Transfer Protocol*)[8], HTTP (engl. *Hypertext Transfer Protocol*)[9] [7].

[5] Spojni pouzdan prijenos kod TCP-a uspostavljanjem logičke veze između procesa na krajnjim računalima te mehanizmima potvrde, retransmisije i očuvanja redoslijeda byteova dok se kod UDP ne uspostavlja logička veza te nema ovih mehanizama.

[6] PICT je Apple-ov format za grafičke datoteke, JPEG (eng. *Joint Photographic Experts Group*) je komprimirani slikovni format s gubicima izveden iz bitmape. MPEG (eng. *Moving Picture Experts Group*) je standard za kompresiju audio-video signala.

[7] DNS je hijerarhijsko raspoređeni sustav imenovanja za računala, servise ili bilo koje sredstvo spojeno na Internet ili privatnu mrežu.

[8] SMTP je uobičajeni protokol za prijenos e-pošte na Internetu.

[9] HTTP je glavna i najčešća metoda za prijenos informacija na *webu*.

3. Sigurnosni kriteriji u funkciji računalnih mreža

Sigurnosni kriteriji ili standardi *cyber* sigurnosti su tehnike kojima se nastoji zaštiti *cyber* okruženje korisnika ili organizacije. *Cyber* okruženje uključuje korisnike, mreže, uređaje, softver, procese, pohranjene i poslane informacije, aplikacije, servise i sustave koji mogu biti spojeni direktno ili indirektno na mrežu. Glavni cilj je smanjiti potencijalne rizike, prevenirati ili ublažiti posljedice *cyber* napada.

U sljedećim podpoglavljima čitatelja se upoznaje sa modelima, znanstvenim disciplinama, funkcijama, protokolima, mehanizmima i postupcima u službi očuvanja sigurnosti i vjerodostojnosti mrežnih komunikacija.

3.1 Model CIA

CIA (engl. *Confidentiality Integrity Availability*, hrv. povjerljivost, integritet, raspoloživost) je model dizajniran sa ciljem vođenja politike informacijske sigurnosti unutar organizacija [9].

Povjerljivost predstavlja skup svih pravila kojima se ograničava pristup osjetljivim informacijama osobama kojima one nisu namijenjene za pregledavanje i manipuliranje, odnosno osobama koje nemaju odgovarajuća ovlaštenja [9].

Ljudski faktor u organizaciji smatra se najvećim izvorom rizika za sigurnost informacija. Ljudi svojim namjernim i nenamjernim djelovanjem uzrokuju proboje u informacijske sustave. Studije, provodene na zaposlenicima stotina američkih kompanija, pokazale su da u 52% slučajeva odgovornost za proboj sigurnosti informacijskog sustava snose ljudske pogreške [10].

Kako bi se postigla povjerljivost odnosno privatnost, ljudski napori moraju biti usmjereni u trening i poboljšanje svojih sposobnosti, eliminiranje navika loših za sigurnost i upoznavanje sa novim tehnologijama i ranjivostima tih tehnologija u kontekstu sigurnosti. Neznanje i neinformiranost eventualno će dovesti do curenja povjerljivih podataka, onemogućavanja pružanja usluga, katastrofalnih događaja i sličnih neželjenih pojava koje će za rezultat imati ogromne novčane gubitke kao i prijetnju za sigurnost osoba.

Ključna mjera za osiguravanje povjerljivost je i enkripcija podataka. Za pristup podacima obično je implementiran mehanizam autentifikacije kroz korisničko ime i šifru, biometriju, tokene i slično [9].

Integritet je pojam kojim se definira održavanje konzistentnosti, točnosti i vjerodostojnosti podataka tijekom cijelog životnog ciklusa. Kako bi se navedeno ostvarilo, cilj mora biti onemogućiti slanje i modificiranje podataka neautoriziranim osobama. Kao i kod povjerljivosti, kriptografske metode imaju značajnu ulogu u

osiguravanja integriteta podataka. Metode korištene u zaštiti integriteta podataka najčešće su provlačenje podataka kroz *hash* funkciju i digitalno potpisivanje [9].

Raspoloživost se najbolje postiže strogim održavanjem svog hardvera, vršenjem popravaka odmah kada za tim postoji potreba te održavanjem pravilnog funkcioniranja okruženja operativnog sustava bez softverskih sukoba uz redovito ažuriranje samog softvera. Pristup informacijama mora biti omogućen ovlaštenim osobama kada im je to potrebno [9].

Informacije imaju vrijednost samo ako im pravi ljudi mogu pristupiti u pravo vrijeme. Stoga ne iznenađuje sve veća učestalost DDOS (engl. *Distributed Denial Of Service*) napada na popularna *web* odredišta. Cilj ovakvih napada je onemogućiti legitimnim korisnicima pristup resursima *web* stranice, rezultat je financijski gubitak za vlasnika stranice zbog nemogućnosti isporučivanja usluge za određeno vrijeme. Osim DDOS[10] napada postoje i druge prijetnje za gubitak raspoloživost poput nestanka električne energije, eksplozije, požara, potresa i slično.

Uz povjerljivost, integritet, raspoloživost za sigurnosni model još je važan uvjet neporecivosti obavljene transakcije. Neporecivost je uvjerenost da netko ne može zanijekati nešto. Obično se odnosi na to da se strani koja je potpisala određeni ugovor onemogući poricanje autentičnosti potpisa.

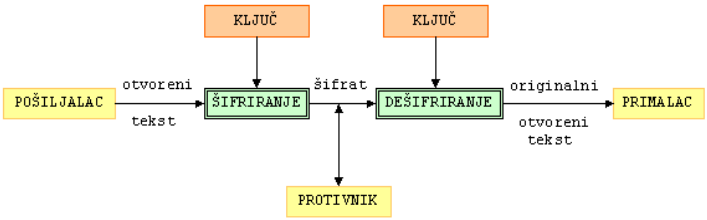
3.2 Kriptografija

Kriptografija je znanstvena disciplina koja se bavi proučavanjem metoda za slanje poruka u takvom obliku da ih samo onaj kome su namijenjene može pročitati. Sama riječ kriptografija je grčkog podrijetla i mogla bi se doslovno prevesti kao *tajnopis.*

Zadatak kriptografije je omogućiti dvjema osobama, pošiljatelju i primatelju poruke, da preko nesigurnog komunikacijskog kanala komuniciraju na način da treća strana, koja se u literaturi spominje kao protivnik, a moguće je da nadzire isti komunikacijski kanal ne dobije saznanje o sadržaju njihovih poruka [11].

Poruku koju pošiljatelj želi poslati primatelju zove se otvoreni tekst (engl. *plaintext*). Pošiljatelj transformira otvoreni tekst koristeći unaprijed dogovoreni ključ. Taj postupak se naziva šifriranje, a dobiveni rezultat šifrat (engl. *ciphertext*) [11].

[10] DDOS (engl. *Distributed Denial-of-Service*) običajeni način na koji se sprječava pristup računalnom sustavu ili sustavima je kroz preopterećivanje računalne mreže slanjem mnogostrukih zahtjeva prema poslužitelju



Slika 1 Sigurna komunikacija preko nesigurnog kanala

Izvor: [11]

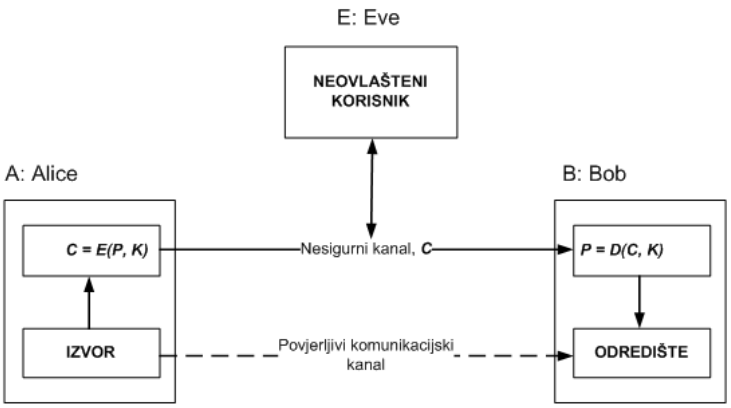
Dakle, prilikom komunikacije, pošiljatelj će primatelju slati poruke u obliku šifrata. Protivnik nadzire komunikacijski kanala i presreće šifrat kojeg ne može razumjeti. Na taj način štiti se otvoreni tekst poruke. Za razliku od protivnika, primatelj poruke poznaje unaprijed dogovoreni ključ kojim je poruka šifrirana te istu može jednostavno dešifrirati i pročitati (vidljivo sa slike 1) [11].

Suprotno dešifriranju, kriptoanaliza ili dekriptiranje je znanstvena disciplina koja se bavi proučavanjem postupaka za čitanje skrivenih poruka bez poznavanja ključa. Kriptologija je pak grana znanosti koja obuhvaća kriptografiju i kriptoanalizu [11].

Kriptografski algoritam ili šifra je matematička funkcija koja se koristi za šifriranje i dešifriranje. Općenito, radi se o dvije funkcije, jednoj za šifriranje, a drugoj za dešifriranje. Te funkcije preslikavaju osnovne elemente otvorenog teksta (najčešće su to slova, bitovi, grupe slova ili bitova) u osnovne elemente šifrata, i obratno. Funkcije se biraju iz određene obitelji funkcija u ovisnosti o ključu. Skup svih mogućih vrijednosti ključeva nazivamo prostor ključeva. Kriptosustav se sastoji od kriptografskog algoritma, te svih mogućih otvorenih tekstova, šifrata i ključeva [11].

3.4 Simetrična kriptografija

Simetrična kriptografija je najstariji oblika kriptografije, stara gotovo koliko i ljudska komunikacija. Naziva se i kriptografijom tajnog ključa jer se podatak kriptira i dekriptira istim tajnim ključem. U simetričnim kriptosustavima ključ kriptiranja KE jednak je ključu dekriptiranja KD. Prema tome zajednički ključ se može označiti jednim simbolom K.



Slika 2: Simetrična kriptografija

Izvor: [11]

Sa slike 2 vidljiva je komunikacija između dva sudionika A i B. Sudionik A šalje sudioniku B poruku kriptiranu tajnim ključem K. Sudionik B poruku dekriptira istim ključem kako bi saznao sadržaj poruke. Neovlašteni korisnik koji prisluškiva komunikacijski kanal tako ne može pročitati poruku.

Za proces kriptiranja u simetričnoj kriptografiji potrebno je znati algoritam kriptiranja i tajni ključ. Nekad su se algoritmi držali u tajnosti, ali se pokazalo da skrivanje algoritma ne doprinosi sigurnosti. Svi suvremeni simetrični algoritmi javno su obznanjeni. Zbog toga ih je u potpunosti moguće testirati i provjeriti njihovu otpornost na napade, odnosno moguće ih je analizirati (kriptoanaliza).

Sigurnost simetričnih algoritama ovisi o sigurnosti samog algoritma i dužini ključa. Najpoznatiji simetrični algoritam je DES (engl. *Data Encryption Standard*), kojeg je razvio IBM 1977. godine. Bio je standardni simetrični algoritam sve do 2000. godine kada ga je zamijenio AES (engl. *Advanced Encryption Standard*), koji rukuje ključevima dužine 128, 192 i 256 bita. Glavni razlog zbog kojeg je DES zamijenjen AES-om je taj što DES ima dužinu ključa od 56 bita [11].

Kratki ključevi mogu značajno kompromitirati sigurnost kriptiranih poruka zbog toga što napadač može pokušati dekriptirati poruku isprobavanjem svakog mogućeg ključa/kombinacije. Unutar računala, kriptografski ključ je prikazan kao niz binarnih znamenaka. Svaka binarna znamenka može biti 0 ili 1. Dakle, ako je ključ duljine jedan bit, postoje dvije moguće kombinacije: 0 i 1. Ako je ključ duljine dva bita, postoje četiri moguće kombinacije: 00, 01, 10 i 11. Ako je ključ duljine tri bita, postoji osam mogućih kombinacija: 000, 001, 010, 011, 100, 101, 110 i 111. Očito je da se svakim dodatnim bitom ključa udvostručuje broj kombinacija.

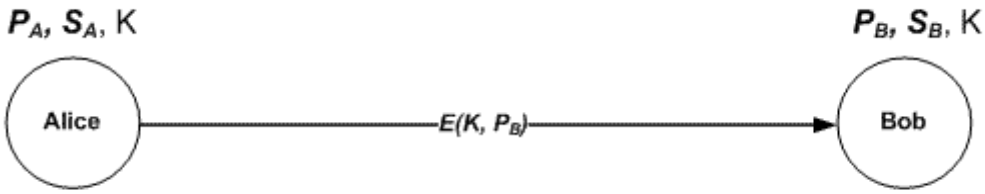
Osnovni nedostatak simetričnih algoritama, odnosno simetričnih kriptosustava, jest upravljanje ključevima, točnije njihova distribucija. Prije same sigurne komunikacije subjekti (sudionici komunikacije) moraju razmijeniti ključeve [11].

Pošto se sigurnost svih zaštićenih (kriptiranih) informacija oslanja na sigurnosti ključa, sigurna razmjena ključeva može postati vrlo ozbiljan problem. Još je veći problem ako se komunikacija odvija na većoj udaljenosti, a u samoj komunikaciji sudjeluje više subjekata [11].

Generiranje i upravljanje velikim brojem ključeva kod simetrične kriptografije najčešće je nepraktično, a razmjena ključeva je nesigurna. Napori uloženi u rješavanje problema razmjene tajnih ključeva doveli su do pojave asimetričnih algoritama odnosno asimetrične kriptografije [12].

3.5 Asimetrična kriptografija

Godine 1976. Whitfield Diffie i Martin Hellman u svojoj su publikaciji "*New Directions in Cryptography*" predstavili ideju razmjene tajnog ključa temeljenu na asimetričnoj kriptografiji, u kojoj je predočeno postojanje para ključeva: ključ za kriptiranje P i ključ za dekriptiranje S, umjesto jednog, istog ključa za kriptiranje i dekriptiranje K. Jedan ključ je javno dostupan svima, naziva se javni ključ, i on služi za kriptiranje. Drugi ključ je poznat samo njegovom vlasniku, naziva se privatni ključ, i služi za dekriptiranje poruka. Poruku je moguće i kriptirati privatnim ključem te se onda dekripcija vrši javnim ključem [12].



Slika 3: Asimetrična kriptografija

Izvor: [12]

Na slici 3 vidljiv je proces razmjene tajnog ključa K između Alice i Boba asimetričnim kriptiranjem. Alice šalje tajni ključ K kriptiran Bobovim javnim ključem PB, a Bob ga dekriptira svojim privatnim ključem SB. Ključ K služi kao ključ kriptiranja u daljnjoj komunikaciji između Alice i Boba [12].

Mnogi protokoli poput SSH, OpenPGP[11], S/MIME[12] i SSL/TLS[13] oslanjaju se na asimetričnu kriptografiju za šifriranje i digitalno potpisivanje. Također je koriste i

[11] OpenPGP (engl. *Open Pretty Good Privacy*) je najčešće korišten standard za šifriranje e-pošte.

računalni programi za potrebe uspostavljanja sigurne veze preko nesigurne mreže poput Interneta [12].

Da bi asimetrično šifriranje ispunilo zahtjeve za povjerljivosti, integritetom, autentičnosti i neporecivosti, korisnici i sustavi moraju biti sigurni da je javni ključ autentičan i da pripada određenoj osobi ili entitetu koji se smatra vjerodostojnim te da neka treća zlonamjerna strana nije preuzela njihov identitet. Najčešći je pristup rješavanju ovih zahtjeva za sigurnošću, korištenje infrastrukture javnog ključa (engl. *Public Key Infrastructure*, PKI) [13].

RSA (Rivest-Shamir-Adleman) je najčešće korišteni asimetrični algoritam objavljen 1978. godine. RSA se koristi za kriptiranje i potpisivanje podataka. Ugrađen je u SSL/TLS protokol koji se koristi za pružanje sigurne komunikacije preko računalne mreže. Proces kriptiranja i potpisivanja provodi se kroz niz modularnih umnožavanja [13].

ECC (engl. *Eliptic Curve Cryptography*, hrv. kriptografija eliptičkih krivulja) je algoritam koji pruža slične funkcionalnosti kao RSA. ECC se implementira u manje uređaje kao što su mobiteli, zahtjeva manje računalne snage nego RSA, smanjena je potrošnja energije. ECC enkripcijski sustav temelji se na ideji korištenja točaka na krivulji za definiranje javnog/privatnog ključa [13].

3.6 Hash funkcije

Message digest je reprezent sadržaja poruke prikazan u obliku numeričkog slijeda fiksne duljine, a izračunava se pomoću *hash* funkcije. Ako se *message digest* kriptira formirat će se digitalni potpis [14].

Kao što je definirano *message digest* je strogo fiksne duljine te ne ovisi o duljini poruke koja je varijabilna. Da bi poruka postala *message digest* mora ući u *hash* funkciju. *Hash* funkcija je transformacija koja mora zadovoljiti dva kriterija, prema [15]:

1. Ne smije biti moguće iz rezultata *hash* funkcije inverzom dobiti sadržaj originalne poruke (kojoj pripada taj *message digest*) osim provjeravanjem svih mogućih poruka,

2. Ne smije biti moguće pronaći dvije poruke čiji je rezultat *hash* funkcije identični *message digest.*

Message digest šalje se zajedno sa izvornom porukom. Primatelj može provjeriti da li je integritet poruke povrijeđen u prijenosu na način da sam generira *message*

[12] S/MIME (engl. *Secure Multipurpose Internet Mail Extension*) je standardu koji nudi dodatni sloj zaštite za e-poštu kroz digitalno potpisivanje i šifriranje

[13] TLS (engl. *Transport Layer Security*) i njegov prethodnik SSL(engl. *Secure Sockets Layer*) su kriptografski protokoli koji omogućuju sigurnu komunikaciju putem Interneta za Internet bankarstvo, e-mail, Internet fax, *instant messaging* itd.

digest primljene poruke i usporedi ga sa pošiljateljevim. Svaka izmjena poruke tijekom prijenosa rezultirat će promijenjenim *message digestom* [14], [15].

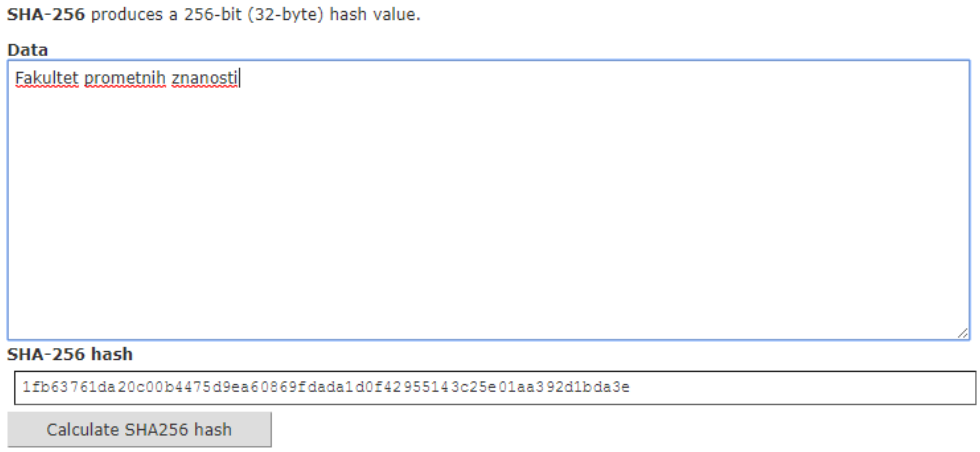
SHA (engl. *Secure Hash Algorithm*) je algoritam *hash* funkcije korišten u sigurnim komunikacijama za dokazivanje integriteta i autentičnosti poruke na strani primatelja. [16], [17].

SHA-1 verzija algoritma za rezultat daje 160-bitni otisak kada se koristi na poruci. Prihvaćen je 1995. i do nedavno je predstavljao standard. Međutim, od tada veliki razvoj i napredak u tehnologiji i kriptografiji zahtijeva bolje i pouzdanije rješenje u vidu SHA-2. SHA-1 je od siječnja 2017. prestao biti podržan od vodećih *web browsera* (Chrome, Safari, Mozilla, Opera) te je počela migracija na SHA-2 [16].

Slika 4: SHA-1 *hash* generator

Izvor: [18]

SHA-2 je set *hash* funkcija koji uključuje SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224 i SHA-512/256. Najčešće korišten je SHA-256 koji koristi 256 bita. Radi na isti način kao SHA-1 s tim da proizvodi dulji otisak. Stoga je SHA-2 sigurniji i vjerodostojniji od SHA-1 koji koristi manji broj bita [16].



Slika 5: SHA-2 (SHA-256) *hash* generator

Izvor: [18]

Iz slika 4 i 5 vidljiv je dulji otisak koji proizvodi SHA-2 što ga čini manje ranjivim na napade. Još jedna prednost SHA-2 je što je manje osjetljiv na koliziju koja bi nastala kada bi dvije različite poruke proizvele isti *hash*, što bi dovelo do zamjene dvije različite poruke te tako do potencijalne sigurnosne ugroze [17].

3.7 Digitalni potpis

Digitalni potpis je izraz koji se upotrebljava za definiranje procesa označavanja ili potpisivanja elektroničkog dokumenta. Proces je zamišljen da bude analogan ručnom potpisu s razlikom da se kod digitalnog potpisivanja koristi tehnologija asimetrične kriptografije [19].

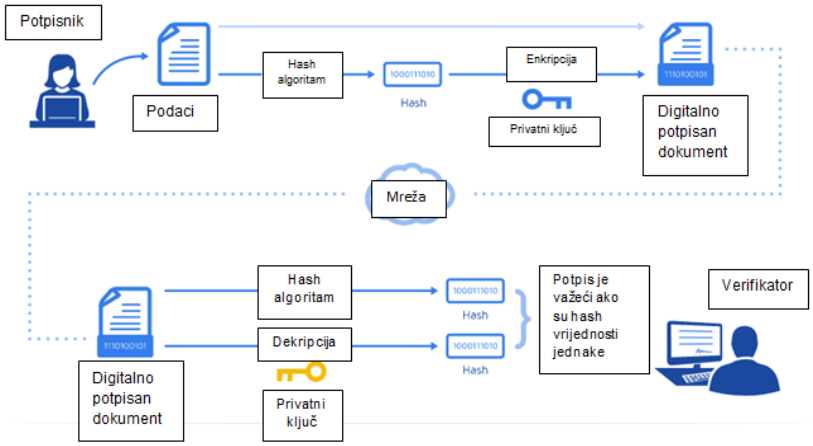
U digitalnom svijetu potrebna su dodatna sigurnosna svojstva prilikom potpisivanja dokumenata. Razlog tomu je što vjerojatnost sporova oko autentičnosti elektronskih transakcija dramatično raste zbog obavljanja istih preko mreže, na udaljenosti, bez sastanka licem u lice. Također, uvijek bi postojala sumnja od neovlaštenih modifikacija dokumenta u procesu [19].

Digitalni potpisi adresiraju ove zabrinutosti i nude bolju sigurnost od klasičnih papirnatih potpisa. U usporedbi s ostalim oblicima potpisa, digitalni su daleko najsigurniji i najpouzdaniji [19].

Svaki je digitalni potpis unikatan. Pružatelji usluga digitalnog potpisivanja koriste protokol PKI[14]. PKI zahtjeva uporabu matematičkog algoritma kojim se generiraju dva ključa: javni i privatni. Kada potpisnik elektronski potpiše dokument, potpis se kreira korištenjem potpisnikovog privatnog ključa. Matematički algoritam djeluje kao *hash* funkcija, kreira unikatni, fiksni numerički niz koji se podudara sa potpisanim dokumentom te nakon toga vrši kriptiranje. Rezultirajući kriptirani podaci čine digitalni potpis, proces digitalnog potpisivanja vidljiv je sa slike 6. Potpis je označen i točnim vremenom kada je obavljeno potpisivanje. Da bi se zaštitio integritet digitalnog potpisa, PKI zahtjeva da ključevi budu kreirani i spremljeni na siguran način. To se postiže radom u sprezi sa pouzdanim *Certification Authorityijem* (CA)[15] [20].

[14] Infrastruktura javnog ključa (engl. *Public Key Infrastructure*) podržava distribuciju i identifikaciju javnih ključeva za šifriranje, omogućujući korisnicima i računalima da sigurno razmjenjuju podatke putem mreža kao što je Internet i provjeravaju identitet druge strane.

[15] U kriptografiji, *Certification Authority* je entitet koji izdaje digitalne certifikate. Digitalni certifikat ovjerava vlasništvo javnog ključa.



Slika 6: Digitalno potpisivanje i slanje dokumenta

Izvor: [20]

Zlonamjerni korisnik koji bi se pokušao predstaviti kao netko drugi, nije u mogućnosti falsificirati digitalni potpis te osobe jer ne posjeduje njen privatni ključ. Također, osoba koja je transakciju digitalno potpisala ne može tvrditi da istu nije poslala ona (da je transakcija rezultat greške sustava u tom trenutku ili da ju je poslao netko drugi) jer je ista transakcija potpisana njenim privatnim ključem (osigurana je neporecivost izvršene transakcije). Ukoliko bi neki napadač modificirao digitalno potpisanu poruku tijekom prijenosa, primateljevo računalo je sposobno detektirati promjenu koja bi tada nastala u *hashu* poruke [21].

3.9 IPsec

IPsec (engl. *IP security*) je skup protokola koji je razvila organizacija IETF (engl. *Internet Engineering Task Force*) za zaštićenu komunikaciju preko Interneta. Protokoli funkcioniraju na trećem sloju OSI modela (mrežnom sloju). Sastavni su dio IPv6 protokola, a mogu se uključiti i unutar IPv4 sustava[16]. IPsec se koristi kod implementacije virtualne privatne mreže (VPN) [22].

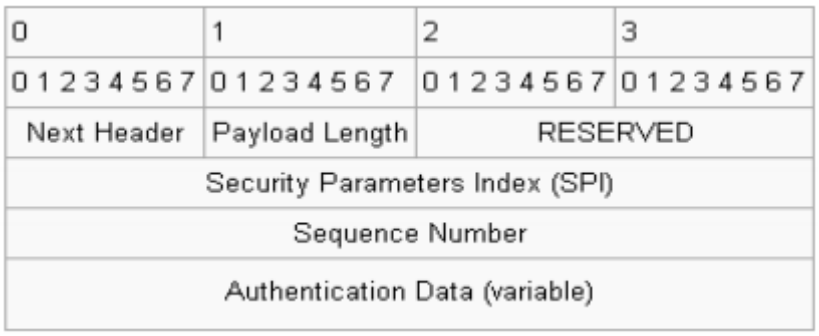
Slika 7: Standardni IP *datagram*

Izvor: [22]

U prvi dio seta protokola IPsec spadaju dva kriptografska protokola:

[16] IPv4 i IPv6 su verzije *Internet Protocola* koji definira format paketa i adresne sheme računala za mrežnu komunikaciju. IPv6 ima broje prednosti nad IPv4 poput većeg adresnog prostora (128-bitna adresna shema naspram 32bitne), bolje sigurnosti, jednostavnijeg zaglavlja itd.

AH (engl. *Authentification Header*) osigurava integritet, autentifikaciju i neporecivost. Opcionalno može poslužiti za zaštitu od *replay* napada (ponavljanja poruke). Definiran je u RFC 2402. Protokol ima vlastito zaglavlje koje se umeće između IP zaglavlja i IP podataka. Standardni IP *datagram* vidljiv je sa slike 7 dok je zaglavlje AH vidljivo sa slike 8 [22].



Slika 8: Zaglavlje AH

Izvor: [22]

Značenje polja u zaglavlju, prema [23]:

Next header (8 bita) (hrv. sljedeće zaglavlje) je vrsta dodatnog zaglavlja, pokazuje koji je protokol višeg sloja zaštićen. Vrijednost se uzima iz popisa brojeva IP protokola.

Payload lenght (8 bita) (hrv. duljina ispune) označava duljinu AH zaglavlja u 32-bitnim riječima, umanjenu za vrijednost dva. Iako se veličina mjeri u jedinicama od 4 okteta[17], duljina ovog zaglavlja mora biti višekratnik od 8 okteta ako se nosi u IPv6 paketu.

Reserved (16 bita) (hrv. rezervirano) je polje rezervirano za buduću upotrebu. Do tad mora biti ispunjeno nulama.

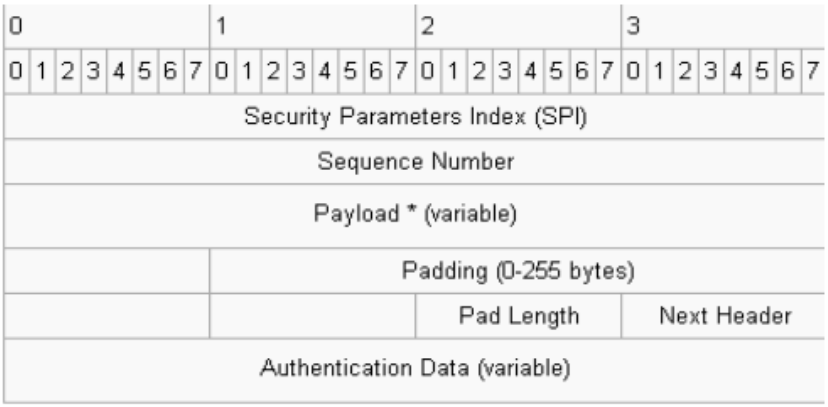
Security parameter index (32 bita) (hrv. popis sigurnosnih parametara) je proizvoljna vrijednost koja se koristi zajedno sa odredišnom IP adresom za identifikaciju sigurnosnih parametara SA (engl. *Security Association*). SA se najjednostavnije definira kao skup parametara i ključeva za kriptiranje i autentifikaciju toka podataka. Potrebna su dva SA za dvosmjerni prijenos.

Sequence number (32 bita) (hrv. sekvencijalni broj) broj koji se monotono i striktno povećava za jedan (inkrementira za 1 za svaki poslani paket). Služi za prevenciju *replay* napada. Kada je uključena detekcija *replay* napada, jednom iskorišten sekvencijski broj ne smije se ponoviti tj. prvo se mora ugovoriti nova SA kada se sekvencijalni broj inkrementira do svoje maksimalne vrijednosti. Tada se vrijednost ponovno vraća na 1.

[17] Oktet je slijed od osam bitova

Authentification data (hrv. autentifikacijski podatci) polje se koristi za autentifikaciju paketa. Varijabilne je duljine. Polje sadrži ICV (engl. *Integrity Check Value*) za provjeru autentičnosti i integriteta poruke. Može sadržavati *padding* za popunjavanje polja na 8-oktetnu granicu za IPv6 ili granicu od četiri okteta za IPv4.

ESP (engl. *Encapsulated Security Payload*) uz integritet, autentifikaciju i neporecivost osigurava i povjerljivost podataka (eng. *payload*) koji se prenose. Definiran je u RFC 2406. Protokol ima vlastito zaglavlje koje se umeće iza IP zaglavlja. Enkapsulira sve podatke višeg sloja te dodaje završni slog u kojem mogu biti sadržani autentifikacijski podaci [24].



Slika 9: ESP datagram

Izvor: [22]

Na slici 9 prikazan je ESP datagram. Značenje polja u zaglavlju je sljedeće, prema [23], [24]:

Security parameter index (32 bita) (hrv. popis sigurnosnih parametara) je polje isto kao i kod AH. Proizvoljne je vrijednosti i služi za identifikaciju SA.

Sequence number (32 bita) (hrv. sekvencijalni broj) je isto polje kao i kod AH. Služi za zaštitu od *replay* napada.

Payload data (varijabilni broj bita) (hrv. podaci) je zaštićeni sadržaj originalnog IP paketa, uključuje i podatke koje se koriste za zaštitu. *Next header* polje pokazuje vrste podataka koji se štite.

Padding (0-255 okteta) (hrv.ispuna) je ispuna za enkripciju. Proširuje *payload* na duljinu koju zahtjeva kriptografski algoritam za šifriranje odnosno blokovi fiksne duljine koje koristi algoritam. Također, služi za poravnavanje sa sljedećim poljem odnosno omogućava da se dobije zahtijevana duljina zaglavlja.

Pad lenght (8 bita) je duljina ispune u oktetima

Next header (8 bita) je polje isto kao i kod AH. Vrijednost se uzima iz iz liste brojeva IP protokola.

Authetification data (hrv. autentifikacijski podaci) je isto polje kao i u AH. Sadrži ICV.

IPsec ima dva načina rada, prema [22]:

• tuneliranje paketa

• transportni način rada

U prvom slučaju nekoliko računala (ili cijela jedna lokalna mreža) sakriva se iza jednog čvora te je kao takva nevidljiva ostatku mreže (samim time i zaštićena od napada). U drugom slučaju paketi se šalju između dva krajnja računala na mreži, pri čemu računalo koje prima paket izvršava sigurnosne provjere prije isporučivanja paketa višim slojevima. U oba slučaja moguće je izgraditi virtualne privatne mreže – VPN (engl. *Virtual Private Network*), što je i osnovna ideja zaštite IPsec protokolima.

Još jedan protokol iz seta IPsec protokola je IKE protokol (engl. *Internal Key Exchange*). IKE protokol obavlja obostranu autentifikaciju korisnika te uspostavlja SA (engl. *Security Association*) vezu. Uspostava SA veze podrazumijeva izračunavanje *keying* materijala te dogovaranje oko skupa algoritama i drugih parametara koji će štititi SA. Protokol radi tako da inicijator veze (engl. *initiator*) nudi prihvatljive parametre za zaštitu SA. Ako ih druga strana (engl. *responder*) prihvati ostvaruje se SA veza [22].

3.10 Firewall

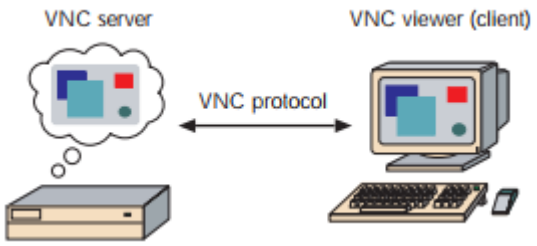
Vatrozid (engl. *firewall*) je sustav mrežne sigurnosti dizajniran za prevenciju neautoriziranog pristupa od i prema privatnoj mreži. Vatrozid može biti implementiran hardverski, softverski ili kombinirano. Mrežni vatrozidi se često koriste za onemogućavanje pristupa neovlaštenim korisnicima sa Interneta prema privatnoj mreži/intranetu. Sve odlazne i dolazne poruke u intranetu prolaze kroz vatrozid. Vatrozid ispitiva svaku poruku te blokira one koje ne zadovoljavaju određene sigurnosne kriterije. Dakle, može se reći da su osnovne zadaće vatrozida filtriranje paketa i uspostavljanje *proxya*. *Proxy* štiti adrese računala u intranetu. Odnosno vrši adresnu translaciju iz privatnih u javne adrese potrebnih za surfanje Internetom (NAT, engl. *Network Address Translation*) [25], [26].

4. Načini udaljenog pristupa računalnoj mreži (VNC, VPN, SSH)

U sljedećim podpoglavljima ulazi se u glavnu temu rada. Proučeni su načini za udaljeni pristup u računalnim mrežama: VNC, VPN i SSH. Za svaki od načina objašnjeni su principi rada, protokoli i mogućnosti primjene.

4.1 Virtual Network Computing (VNC)

VNC (engl. *Virtual Network Computing*) je sustav za pristup i upravljanje udaljenim računalom. Protokol koji se koristi je RFB (engl. *Remote Framebuffer*). On obavlja prijenos događaja s jednog računala (npr. ulaza s tipkovnice ili miša) na drugo te pritom omogućuje prikaz promjena na zaslonu. Interoperabilan je, što znači da se VNC preglednik na jednom operacijskom sustavu može povezati s VNC poslužiteljem na istom ili bilo kojem drugom operacijskom sustavu. Također, više korisnika može se povezati s istim VNC poslužiteljem istovremeno. Popularna uporaba ove tehnologije uključuje udaljenu tehničku podršku i pristup datotekama na radnom računalu s kućnog računala [27], [28].



Slika 10: VNC klijent, *server* i protokol

Izvor: [27]

VNC sustav sastoji se od tri dijela (vidljivo sa slike 10):

1. VNC server je program na računalu koji služi za pružanje kontrole klijentu.

2. VNC klijent je program koji služi za pregled, kontrolu i interakciju s poslužiteljem.

3. VNC protokol tj. RFB protokol je vrlo jednostavni protokol za prijenos slike ili poruka o događajima između poslužitelja i klijenta.

U uobičajenom načinu rada klijent ili *viewer* povezuje se s priključkom na serveru (*port* 5900). Postoji i alternativa koja podrazumijeva korištenje *web* preglednika kao klijenta te povezivanje na server preko *porta* 5800. Server se može povezati sa *viewerom* u način slušanja (engl. *listening mode*) preko *porta* 5500. Prednost ovog načina je što strana *servera* ne mora konfigurirati vatrozid za omogućavanje pristupa preko *porta* 5900 ili 5800. Konfiguraciju je dužan napravit korisnik na klijentskoj strani koji često posjeduje veće znanje o problematici nego korisnik na strani *servera* [29].

Kada se uspostavi veza, *server* će slati male kvadratiće *framebuffera* klijentu. RFB protokol tako može zauzeti veći dio *bandwitdtha*[18]. Poželjno je da obje strane imaju širokopojasnu vezu. Osim toga, razrađene su različite metode za smanjenje *overheada*[19] u komunikaciji odnosno postoje određeni načini kodiranja za efikasniji prijenos prethodno spomenutih kvadratića. VNC protokol dopušta klijentu i *serveru* dogovaranje načina kodiranja kojeg će koristiti u komunikaciji. Najjednostavniji način kodiranja kojeg podržavaju svi klijenti i *serveri* je *raw encoding.* Radi se o metodi kod koje se podaci o pikselu (najmanji grafički element slike) šalju u redoslijedu od lijeve strane prema desnoj, a nakon slanja cijele slike šalju se samo dijelovi koji se mijenjaju. Ovo kodiranje je korisno i efikasno samo ako se mali dijelovi slike mijenjaju od okvira do okvira. To je slučaj kod pomicanja miša na statičnoj pozadini ili upisa teksta u okvir. Međutim, kod velikih promjena slike, kao kod pregleda video zapisa, ova metoda postaje neuporabljiva [28], [29].

Jedan od vrlo važnih aspekata prilikom pristupa udaljenom računalu je sigurnost podataka. Kod VNC sustava, većina sigurnosnih ranjivosti povezana je s RFB protokolom koji je po svojoj prirodi nesiguran protokol. Razlog tomu je što je ovaj protokol fokusiran na učinkovit prijenos događaja između udaljenih računala, dok su postupci autentikacije i šifriranja potpuno zanemareni. Prema tome, prve inačice sustava bile su posebno ranjive na napade dešifriranja lozinki i otkrivanja osjetljivih podataka [29].

Osnovna metoda za dešifriranje lozinki bila je *brute force* napad koji se zasniva na isprobavanju svih mogućih kombinacija ključa dok se ne otkrije odgovarajući. Efikasniji način krađe lozinki razvijen je zahvaljujući nepravilnoj pohrani lozinki u registre (kod operacijskog sustava Windows) što je omogućilo razvoj posebnih alata za izvođenje tog napada [29].

Nesigurnost VNC sustava uzrokovao je nedostatak šifriranja komunikacije između klijenta i poslužitelja nakon uspostave veze. Zahvaljujući tomu napadač je mogao, izvođenjem *man in the middle*[20] napada, presresti poruke te dobiti pristup svim informacijama koje korisnik razmjenjuje s udaljenim sustavom. Za eliminiranje uspješnosti tih napada ključno je zaštiti komunikaciju uporabom SSH (engl. *Secure Shell*) protokola, mrežnog protokola za razmjenu podataka uporabom sigurne veze između dva mrežna uređaja. Izvedba na računalima s operacijskim sustavom Windows uključuje preusmjeravanje prometa preko računala s Unix platformom [28], [29].

[18] *Bandwidth* je širina frekvencijskog spektra

[19] *Overhead* ili kontrolne informacije

[20] U kriptografiji i računalnoj sigurnosti, *man in the middle* je napad u kojem napadač potajno odašilje i eventualno mijenja komunikaciju između dvije strane koje vjeruju da izravno komuniciraju jedna s drugom.

4.2 Remote Desktop Protocol (RDP)

Remote Desktop Protocol (RDP) je Microsoftov višekanalni protokol, omogućuje zasebne virtualne kanale za prijenos prezentacijskih podataka, komunikaciju serijskih uređaja, informacija o licenciranju, visoko enkriptiranih podataka. RDP protokol udaljenom korisniku omogućuje dodavanje grafičkog sučelja na radnu površinu drugog računala. Razvijen je na temelju ITU-T.120 skupa protokola, kompatibilan je s više vrsta LAN protokola. U biti je RDP vrlo sličan VNC-u. Kao i kod VNC-a primjenjivat će se u svrhu udaljene tehničke podrške, za izvođenje dijagnostike i rješavanje problema na udaljenom računalu [30].

Ostale važne značajke i mogućnosti RDP-a, prema [30]:

1. RDP može podržati do 64.000 nezavisnih kanala za prijenos podataka,

2. Podaci se kriptiraju koristeći RSA algoritam odnosno RC4 protočnu šifru. RC4 je dizajniran za efikasno šifriranje malih količina podataka. Administrator ima opciju korištenja 56-bitnog ili 128-bitnog ključa,

3. RDP podržava različite mehanizme za redukciju količine podataka prenesenih preko mrežne konekcije (kompresija, *cache*[21]). Time se postižu bolje performanse preko niskopropusnih veza,

4. Korisnik se može ručno odspojiti sa RDP sesije bez potrebe za odjavljivanjem (engl. *log-off*),

5. Korisnici mogu kopirati, lijepiti, brisati tekst i slike između aplikacija koje se izvršavaju na lokalnom računalu i onih koje se izvšavaju u RDP sesiji kao i aplikacija između sesija i

6. Aplikacije koje se izvršavaju unutar RDP sesije mogu printati sadržaj na printer spojen na uređaj klijenta.

RDP pruža udaljeni pristup preko *porta* 3389. RDP aplikacija pakira podatke koji se trebaju prenijeti, *Microsoft Communications Service* usmjerava podatke u RDP kanal. Operacijski sustav tada kriptira RDP podatke te im dodaje okvir kako bi se mogli dalje prenositi [31].

Terminal Server Device Redirector Driver zadužen je za upravljanje svim aktivnostima RDP protokola. Ovaj se program (*driver*) sastoji od podkomponenata kao što je RDP *driver* (Wdtshare.sys) koji upravlja korisničkim sučeljima, prijenosom, šifriranjem, kompresijom i izradom okvira. *Transport Driver* (Tdtcp.sys) odgovoran je

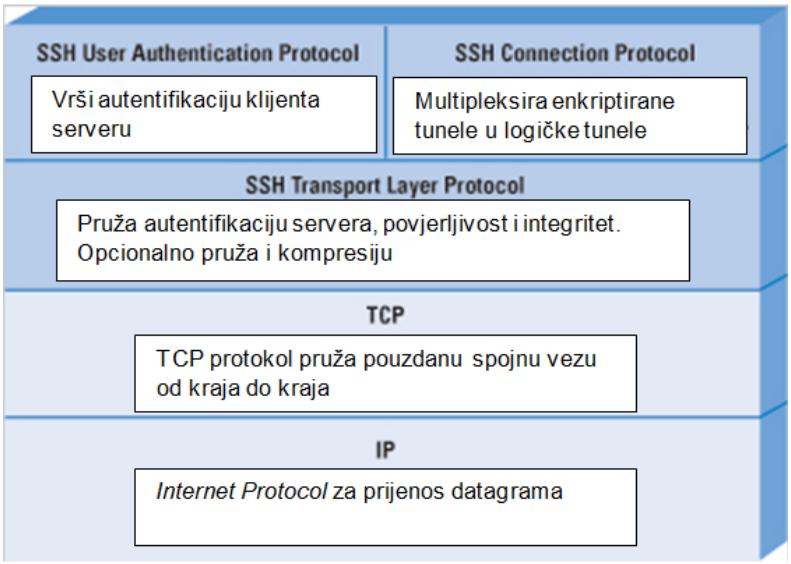
[21] *cache* je komponenta koja pohranjuje podatke kako bi u budućnosti zahtjevi za tim podacima bili brže posluženi

za pakiranje protokola na način da istom omogući da bude poslan TCP/IP mrežom [31].

4.3 Secure Shell (SSH)

SSH je protokol za sigurnu mrežnu komunikaciju dizajniran tako da bude jednostavan za implementaciju. U svojoj incijalnoj verziji SSH1, predstavljen je kao rješenje usmjereno na pružanje sigurnog udaljenog pristupa u računalnim mrežama. Osnovni razlog za razvoj ovog protokola bio je zamijeniti telnet i ostala *remote login*[22] rješenja koja nisu bila dizajnirana u smislu pružanja sigurnosti. Enkripcijom koju koristi SSH nastoji se osigurati povjerljivost i integritet podataka pri prijenosu nesigurnom, javnom mrežom kao što je Internet [32], [33].

SSH je organiziran kao tri protokola koja se izvršavaju iznad TCP protokola (vidljivo sa slike 11): *Transport Layer Protocol*, *User Authentication Protocol* i *Connection Protocol*.



Slika 11: Protokolni složaj SSH

Izvor: [32]

Transport Layer Protocol (RFC 4253): zadaća ovog protokola je inicijalna razmjena ključeva (engl. *Initial Key Exchange*), enkripcija, autentifikacija servera, kompresija te provjera integriteta. Također, protokol će organizirati ponovnu razmjenu ključeva nakon prijenosa jednog gigabajta podataka ili nakon jednog sata [34].

Slika 12 prikazuje slijed događaja u SSH *Transport Layer Protocolu*. Prvo, klijent uspostavlja TCP vezu s poslužiteljem. Kada se uspostavi veza, klijent i poslužitelj razmjenjuju pakete. Paketi su u sljedećem formatu [32]:

[22] *remote login* je stari protokol za pristup udaljenom računalu u svrhu upravljanja i konfiguracije.

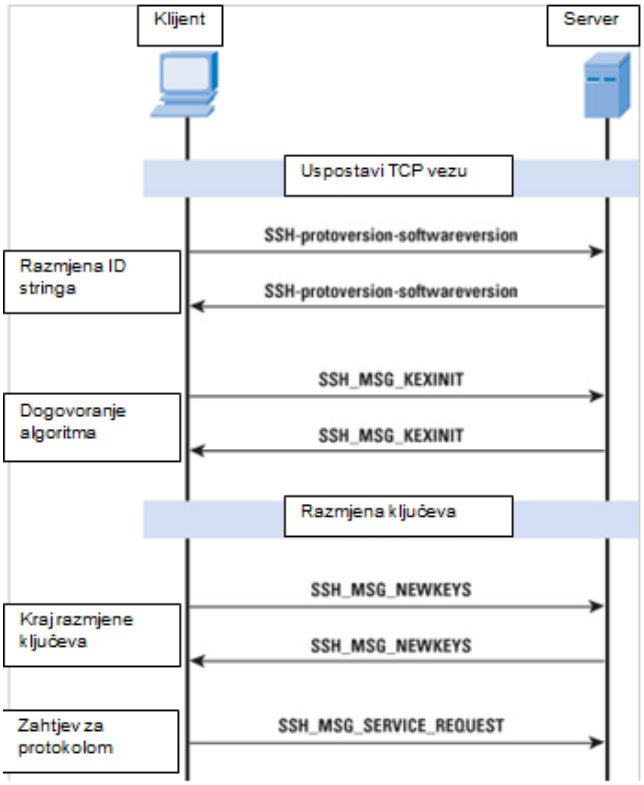
Packet lenght je duljina paketa u bitovima, ne uključujući duljinu *packet lenght* i MAC polja

Padding lenght je duljina *random padding* polja

Payload predstavlja koristan sadržaj paketa. Može biti komprimiran ili nekomprimiran.

Random padding je polje ispune koje se dodaje kada je dogovoren enkripcijski algoritam.

Message Authentication Code (MAC) je polje koje sadrži određenu vrijednost ako je prethodno dogovorena autentifikacija poruka. Služi za provjeru sigurnosne nepovredivosti paketa. Sa slike je vidljivo da se cijeli paket (osim MAC polja) kriptira ako su dogovoreni algoritmi te ako je izračunata MAC vrijednost.



Slika 12: Razmjena paketa kod Transport Layer protokola

Izvor: [32]

Nakon uspostavljene TCP veze sljedeći korak je *Identification String Exchange.* Klijent šalje serveru paket sa identifikacijskim *stringom*[23] koji je u formatu:

SSH-protoversion-softwareversion SP comments CR LF

[23] *String* je pojam koji označava slijed znakova

Server će odgovoriti sa vlastitim identifikacijskim *stringom*. Ovi se *stringovi* koriste u Diffie-Hellman razmjeni ključeva [32].

Sljedeći korak je dogovaranje algoritma. Obje strane šalju poruku SSH_MSG_KEXINIT koja sadrži listu podržanih algoritama u redoslijedu prema preferencijama pošiljatelja [32].

Nakon dogovora algoritma slijedi razmjena ključeva (Diffie Hellman). Potreban je samo jedan paket u oba smjera. Poruka SSH_MSG_NEWKEYS označava kraj postupka [32].

Posljednji korak je zahtjev za uslugom. Klijent šalje paket SSH_MSG_SERVICE_REQUEST kojim zahtjeva *Authentication Protocol* ili *Connection Protocol* [32].

User Authentication Layer (RFC 4252): zadaća ovog sloja je autentifikacija klijenta. Za tu svrhu pruža brojne autentifikacijske metode npr. standardnom lozinkom korisnika, javnim ključem, jednokratnom lozinkom, GSSAPI (engl*. Generic Security Services Application Program Interface*) metodom koja je IETF standard i omogućuje korištenje vanjskih mehanizama autentifikacije poput Kerberosa 5[24] i NTLM (engl. NT *Lan Manager*)[25].

Autentifikacijski sloj poznaje tri vrste poruka:

SSH_MSG_USERAUTH_REQUEST: autentifikacijski zahtjev koji šalje klijent,

SSH_MSG_USERAUTH_SUCCESS: autentifikacija je uspješno završena,

SSH_MSG_USERAUTH_FAILURE: neuspjela autentifikacija. Server je ili odbio zahtjev ili ga je prihvatio, ali traži dodatne potvrde identiteta.

Connection Layer (RFC 4254) je sloj/protokol koji definira kanale, zahtjeve kanala i globalne zahtjeve pomoću kojih se pružaju SSH usluge. Jedna SSH konekcija može omogućiti više kanala u isto vrijeme. Svakim se kanalom podaci prenose u oba smjera [34].

Neki od standardnih vrsta kanala su, prema [34]:

shell: kanal za siguran udaljeni pristup, SFTP i SCP prijenose

direct-tcpip: za *forwardiranje* veze od klijenta prema serveru (lokalno *forwardiranje*)

forwarded-tcpip: za *forwardiranje* veze od servera prema klijentu (udaljeno *forwardiranje*)

[24] Kerberos je protokol za provjeru autentičnosti u računalnim mrežama koji radi na temelju "ulaznica" kako bi omogućio čvorovima koji komuniciraju preko nesigurne mreže da dokažu svoj identitet jedan drugome na siguran način.

[25] NTLM je Microsoftov protokol koji korisnicima pruža povjerljivost, autentifikaciju i integritet.

4.4 Virtual Private Network (VPN)

Veliki napredak u tehnologiji transporta i komunikacija zadnjih desetljeća vodio je do širenja interesa i ekspanzije međunarodnih poslovnih subjekata. Definira se novi pojam multinacionalnih korporacija. Takve organizacije svoje poslovne objekte imaju diljem svijeta. Podrazumijeva se, da u današnje vrijeme, na svakoj od tih lokacija postoji LAN mreža u vlasništvu te kompanije. Za ostvarenje dobrih poslovnih rezultata bitno je da ti međunarodni uredi ili objekti mogu međusobno komunicirati na brz, pouzdan i prije svega siguran način [35].

Za postizanje prethodno navedenih zahtjeva u komunikaciji, kompanije su iznajmljivale telekomunikacijske linije kako bi proširile svoju privatnu mrežu na veliko geografsko područje (WAN). Očite su prednosti WAN mreže u vlasništvu kompanije naspram javne mreže Interneta u gotovo svakom aspektu. Problem nastaje zbog potrebe za značajnim financijskim ulaganjima kompanije u održavanje komunikacijske infrastrukture [35].

VPN je dizajniran kako bi riješio ovu problematiku. Cjenovno je efikasno rješenje za udaljeni pristup, a da pritom ne riskira eventualnu kompromitaciju osjetljivih podataka. Način na koji VPN povezuje udaljena računala putem javne mreže ekvivalentan je korištenju iznajmljenih linija [38].

Postoje dvije česte vrste VPN-a [35]:

Remote Access VPN koja se još zove i *Virtual private dial up network* (VPDN), koristi se za kriptiranu vezu od korisnika prema LAN-u. Zaposlenici kompanije na različitim udaljenim lokacijama rabe ovu vrstu VPN za siguran pristup privatnoj mreži.

Site to site VPN se izvodi postavljanjem posebne opreme i uporabom visoke razine enkripcije. *Site to site* VPN izgrađen između ureda iste kompanije naziva se intranet, a između kompanije i klijenta extranet.

Posebna oprema potrebna za izgraditi VPN veze uključuje, prema [35]:

VPN koncentrator: služi za uspostavljanje VPN tunela i rukovanje velikim brojem istodobnih veza,

VPN omogućen/optimiziran *router*: ovo je tipični *router* koji delegira promet na mreži, ali s dodatnom značajkom usmjeravanja prometa pomoću protokola specifičnih za VPN,

VPN-omogućen vatrozid: ovo je uobičajeni *firewall* koji štiti promet između mreža, ali s dodanom značajkom upravljanja prometom koristeći protokole specifične za VPN-ove i

VPN klijent: program koji služi kao sučelje za tuneliranja prometa za višestruke veze.

Tuneliranje

Tuneliranje uključuje uspostavljanje i održavanje logičke mrežne veze. Na toj vezi, paketi konstruirani u specifični VPN protokolski format enkapsuliraju unutar nekog baznog (engl. *base*) ili nositeljskog (engl. *carrier*) protokola te se zatim prenose od VPN klijenta prema VPN poslužitelju gdje se na prijamnoj strani vrši postupak inverzan enkapsulaciji.

Tehnologije kojima se obavlja postupak tuneliranja su, prema [36]:

• GRE (engl . *Generic Routing Encapsulation*)

• ATMP (engl . *Ascend Tunnel Management Protocol*)

• *Mobile* IP – za mobilne korisnike

• IPSec (engl . *Internet Protocol Security Tunnel Mode*)

• PPTP (engl . *Point-to-Point Tunneling Protocol*)

• L2F (engl. *Layer 2 Forwarding*) • L2TP (engl . *Layer 2 Tunneling Protocol*)

Protokoli

Internet Protocol Security (IPsec) je jedan set protokola koji se koristi upravo kod VPN za osiguranje povjerljivosti. Pruža napredne sigurnosne značajke poput snažnih algoritama za kriptiranje i opsežne autentifikacije. Ima dva načina rada: tunelski i transportni. Tunelski kriptira zaglavlje i payload svakog paketa dok transportni kriptira samo *payload*. IPsec set protokola detaljnije je objašnjen u prijašnjem poglavlju.

Point to Point Tunneling Protocol (PPTP) je jedan od najstarijih VPN protokola koji je još uvijek u uporabi. Prisutan je od Windows 95 OS-a pa sve do najnovijih verzija. Koristi TCP *port* 1723. PPTP je razvijen Microsoftovom inicijativom kako bi enkapsulirao drugi protokol pod nazivom PPP (engl. *Point-to-Point Protocol*). Od svih VPN protokola, PPTP je jedan od najčešćih, najlakši je za postavljanje i računalno najbrži. Zbog toga je PPTP koristan kod aplikacija u kojima je potrebna što veća brzina, kao što su audio ili video *streaming*. Međutim, PPTP je također podložan ozbiljnim sigurnosnim ranjivostima. Njegovi temeljni protokoli za provjeru autentičnosti, MS-CHAP v1 i v2[26], dokazano su nesigurni. Iz tog razloga, PPTP se ne preporučuje [37].

Layer 2 Forwarding (L2F) je protokol za tuneliranje razvijen od Ciscoa. Opisan je u RFC 2341. Sličan je Microsoftovom PPTP protokolu. L2F djeluje na mrežnom sloju, a neovisan je o prijenosnom mediju. Omogućava *dial-up* pristup poslužiteljima.

[26] MS-CHAP je Microsoftova verzija CHAP (engl. *Challenge-Handshake Authentication Protocol*). Postoje dvije verzije koje se koriste za autentifikaciju u VPN.

Osnovna funkcija L2F protokola je osiguranje mehanizma tuneliranja za okvire prijenosnog sloja ili protokole viših slojeva. Enkapsulirani paketi se prenose preko WAN spojeva do L2F poslužitelja, gdje se obavlja inverzna enkapsulacija, i prosljeđivanje u mrežu. L2F ne definira klijente i funkcionira samo u obavezno (engl. *compulsory*)[27] definiranim tunelima. Cisco i Microsoft odlučili su spojiti svoja dva protokola u jedan, *naziva Layer 2 Tunneling Protocol* (L2TP) [36].

Layer 2 Tunneling Protocol (L2TP) nastao je spajanjem najboljih značajki PPTP i L2F protokola. L2TP je mrežni protokol za tuneliranje PPP okvira preko mreže. L2TP enkapsulira PPP okvira za slanje preko IP, X25[28], *Frame Relay*, ATM. Struktura L2TP paketa vidljiva je sa slike 28. Podaci iz enkapsuliranih PPP okvira mogu biti šifrirani ili komprimirani. L2TP koristi UDP i nizove L2TP poruka za održavanje tunela preko IP mreža. L2TP se sastoji od dva elementa: L2TP pristupnog koncentratora (engl. L2TP *Access Concentrator*, LAC) i L2TP mrežnih poslužitelja (engl. L2TP *Network Servers*, LNS). LNS predstavlja logičku krajnju točku PPP sjednice koja se tunelira kroz neki sustav korištenjem pristupnog koncentratora (LAC). L2TP podržava obavezno definirane tunele kao i proizvoljne (engl. *voluntary*)[29]. Način rada obvezno definiranog tunela opisan je sljedećim nizom koraka:

L2TP definira dvije vrste poruka: kontrolne poruke i podatkovne poruke. Kontrolne poruke koriste se prilikom uspostave, održavanja i čišćenja tunela. Podatkovne poruke se koriste za enkapsulaciju PPP okvira koji se prenose kroz tunel. Kontrolne poruke definiraju pouzdani kontrolni kanal unutar L2TP koji garantira dostavu. Podatkovne poruke se šalju ponovno ukoliko dođe do gubljenja paketa. PPP okviri se preko nepouzdanog podatkovnog kanala šalju enkapsulirani sa L2TP zaglavljima, a zatim i sa prijenosnim zaglavljima kao što su UDP, *Frame Relay*[30], ATM itd. Kontrolne poruke šalju se preko pouzdanog L2TP kontrolnog kanala. Slijedni brojevi su nužni u svim kontrolnim porukama koje služe da bi osigurale pouzdanu dostavu kroz kontrolni kanal. Podatkovne poruke mogu imati slijedne brojeve za utvrđivanje ispravnog redoslijeda i detekciju paketa koji nedostaju [36].

L2TP koristi NCP (engl. *Network Control Protocol*) za dodjelu IP adresa i autentikacijske sheme PPP (PAP[31] i CHAP) za autentikaciju korisnika i kontrolu pristupa mrežnim resursima. L2TP pažnju obraća samo na povjerljivost, integritet i autentičnost L2TP paketa između krajnjih točaka tunela, odnosno LAC i LNS. Kada radi preko IP-a, sigurnost daje IPSec korištenjem ESP i/ili AH [36].

---

[27] Kod ovakvog načina tuneliranja udaljeni VPN poslužitelj konfigurira i kreira tunel. VPN poslužitelj je u tom slučaju krajnja točka.

[28] X.25 protokol omogućuje računalima na različitim javnim mrežama da komuniciraju putem posredničkog računala na razini mrežnog sloja.

[29] Računalo korisnika je krajnja točka tunela i djeluje kao klijent tunela. Ovdje klijent ili korisnik izdaje zahtjev za konfiguriranje i izradu dobrovoljnog tunela.

[30] *Frame relay* je pojednostavljena forma komutacije paketa u kome se sinkroni okviri podataka usmjeravaju k različitim odredištima zavisno od informacija sadržanih u zaglavlju okvira.

[31] PAP (engl. *Password Authentication Protocol*) rabi se za provjeru korisničnih računa bez korištenja šifre u komunikacijskom protokolu PPP.

5. Pregled i testiranje načina udaljenog pristupa

U sljedećim podpoglavljima svakom od načina udaljenog pristupa u računalnim mrežama pridružit će se odgovarajuća programska rješenja kojima privatni i poslovni korisnici mogu jednostavno ostvariti zahtijevanu funkcionalnost. Programski alati koji se testiraju odabrani su prema određenim kriterijima poput popularnosti, relevantnosti, kompatibilnosti, performansama korištenja i slično. Namjera autora je bila evaluacijom velikog broja rješenja koja su dostupna na tržištu za svaki od u radu obrađenih tehnologija, doći do najpraktičnijih rješenja koja će biti detaljno analizirana u sljedećim stranicama.

5.1 RealVNC *Enterprise*

Za svrhu udaljenog pristupa i kontrole računala danas postoje brojna rješenja. Najpoznatija su RealVNC, TightVNC, UltraVNC, EchoVNC, Remote Desktop Connection, TeamViewer. RealVNC je kompanija koja stoji iza rješenja koje je odabrano za testiranje. U 2002. godini američka telekomunikacijska kompanija AT&T zatvara svoj istraživački laboratorij Olivetti Research Lab u Cambridgeu. Tim ljudi prethodno zaposlenih u tom laboratoriju tada osniva RealVNC Limited. Osnivači RealVNC-a izumitelji su protokola RFB koji je prihvaćen kao Internet standard (RFC 6143). Od 2002. njihov programski alat je preuzet 250 milijuna puta od korisnika iz 160 zemalja svijeta, također dobitnici su brojnih nagrada [39].

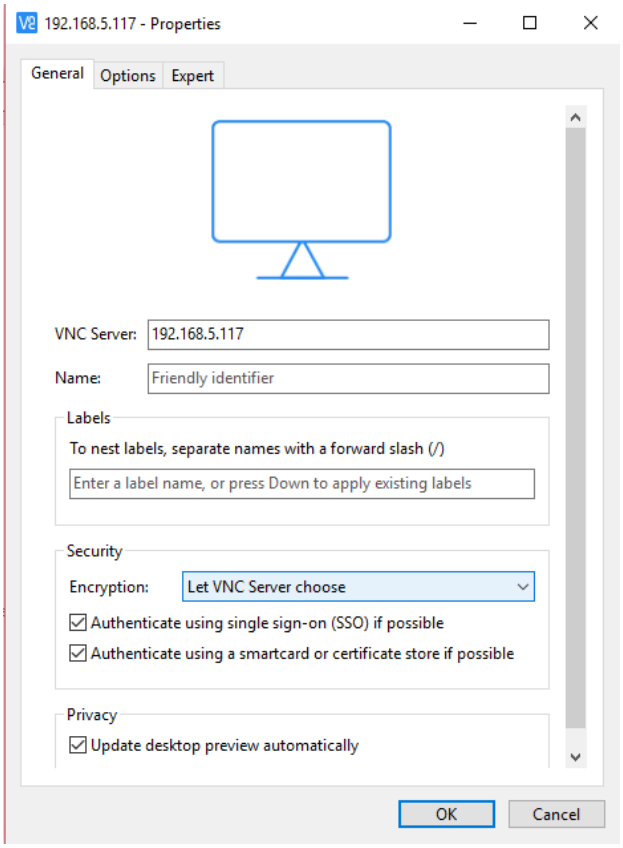
Programski alat je dostupan za *download* na sljedećem *linku* (korištena verzija je 6.2.0):

https://www.realvnc.com/en/download/vnc/

Postoje tri vrste pretplate*: Home, Professional i Enterprise*. *Home* verzija je besplatna za nekomercijalnu uporabu i pruža osnovne funkcionalnosti poput *cloud*[32] povezivosti, *online* upravljanje računom, podrška za sve platforme, autorizacija lozinkom, 128-bitnu AES enkripciju VNC sesije. Za potrebe testiranja koristit će se *Enterprise* verzija, plaća se godišnje, namijenjena je korištenju u korporacijama. *Enterprise* ima bolji sigurnosni aspekt, koristi 256 bitnu AES enkripciju [39].

U poglavlju VNC bilo je govora o nekim sigurnosnim ranjivostima RFB protokola. Svi slučajevi kada su iste detektirane i ispravljene transparentno su prikazani na RealVNC *web* stranici te sama kompanija tvrdi da u ovom trenutku nema opasnijih prijetnji i da konstantno ulažu napore u poboljšanje sigurnosti [39].

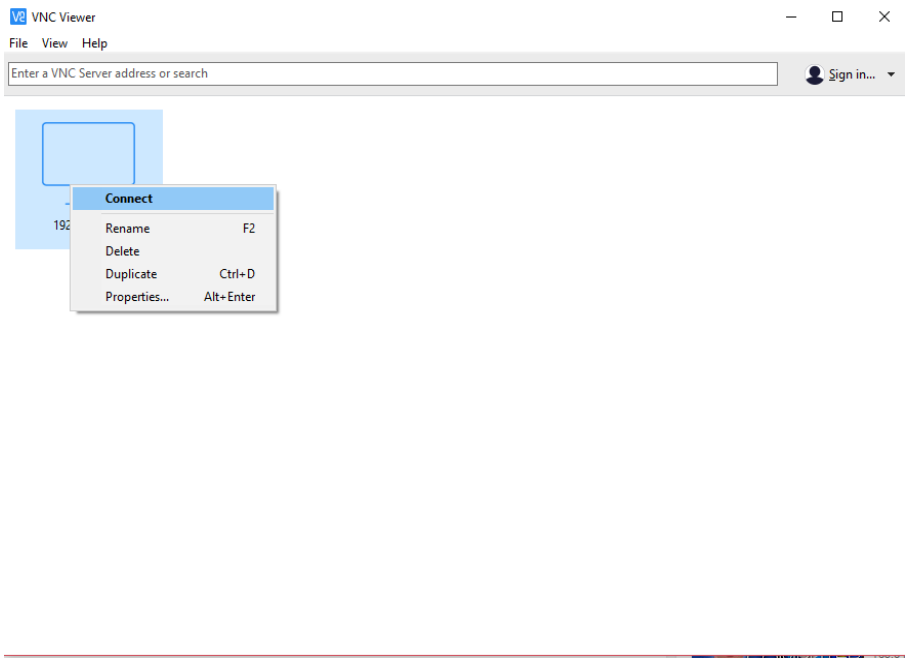
[32] Koncept računalstva u oblaku (engl. *Cloud*) se oslanja na dijeljenje resursa preko mreže, najčešće Interneta. Krajni korisnici pristupaju aplikacijama u oblaku preko *browsera* ili *desktop* aplikacije na mobilnom telefonu, dok se softver i korisnički podaci nalaze na serverima na udaljenoj lokaciji.



Slika 13: *Connect* funkcija VNC *Viewera*

Izvor: Autor

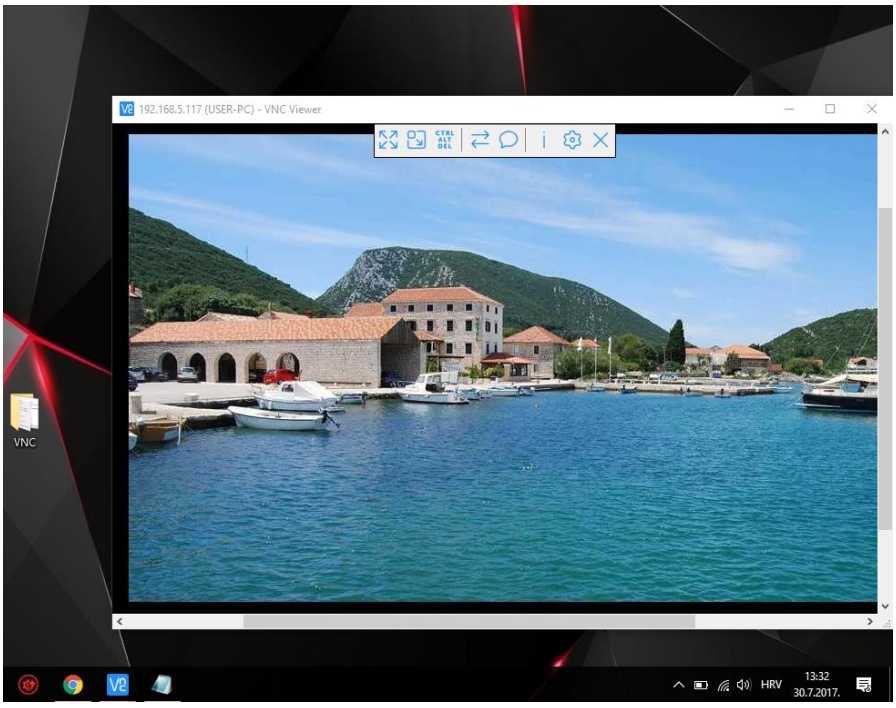
Na računalu s kojeg će se vršiti manipulacija udaljenim računalom instalira se VNC Viewer (VNC klijent). Na računalu kojeg se želi kontrolirati instalira se VNC Connect (VNC *server*). Kada instalacija završi vatrozid je odmah konfiguriran da propusti VNC sesiju. Računalo sa VNC Connect aplikacijom (*server*) ima automatski prikazanu IP adresu. Nju mora upisati korisnik na klijentskoj strani da bi se identificiralo *serversko* računalo. To se postiže tako da se u traci VNC Viewera odabere *Connect* (vidljivo sa slike 13). Moguće se dogovoriti sa *serverskom* stranom o vrsti enkripcije. Ovdje je odabrano da *serverska* strana odluči o tome. Potom je potrebno kliknuti *OK*.



Slika 14: Povezivanje sa udaljenim računalom u sučelju VNC Viewera

Izvor: Autor

Sada je samo potrebno desno kliknuti na novo pojavljenu ikonu *servera* u sučelju Viewera i odabrati *Connect* (vidljivo sa slike 14). Nakon toga slijedi proces autentifikacije u kojem klijent unosi ime i lozinku koju udaljeno računalo koristi prilikom prijave u operativni sustav.



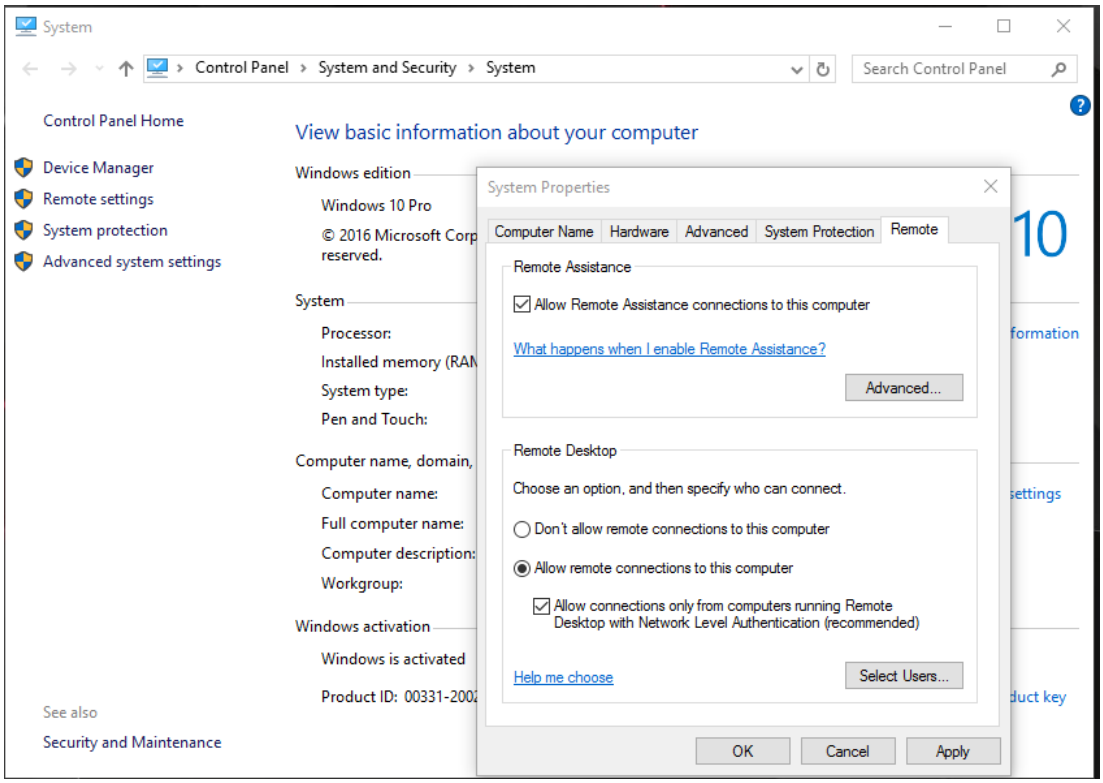
Slika 15: VNC sesija

Izvor: Autor

Sada je uspostavljena VNC sesija sa *serverom* (vidljivo sa slike 15). U VNC Vieweru vidljiv je *display servera*. Svaka akcija koju poduzme klijent, npr. pomicanje kursora, izvršit će se na *serverskoj* strani, a *display* Viewera ažurirat će se i prikazati promjene u stvarnom vremenu. U traci Viewera sada je moguće pokrenuti *chat* prozor sa korisnikom na serverskoj strani, ući u *Task Manager*, izvršiti transfer podataka između klijenta i *servera*, pregledati informacije o sesiji. Klijentsko računalo u ovom testiranju koristi Windows 10 OS dok serversko računalo koristi Windows 7 OS i nema nikakvih problema sa interoperabilnosti.

5.2 Remote Desktop Connection

Remote Desktop Connection (RDC) Microsoftovo je rješenje za udaljeni pristup integrirano u Windows operativne sustave. U sljedećim koracima objašnjen je način na koji se pristupa kućnom računalu sa udaljenog poslovnog računala. Korištena verzija alata je 10.0.14393.



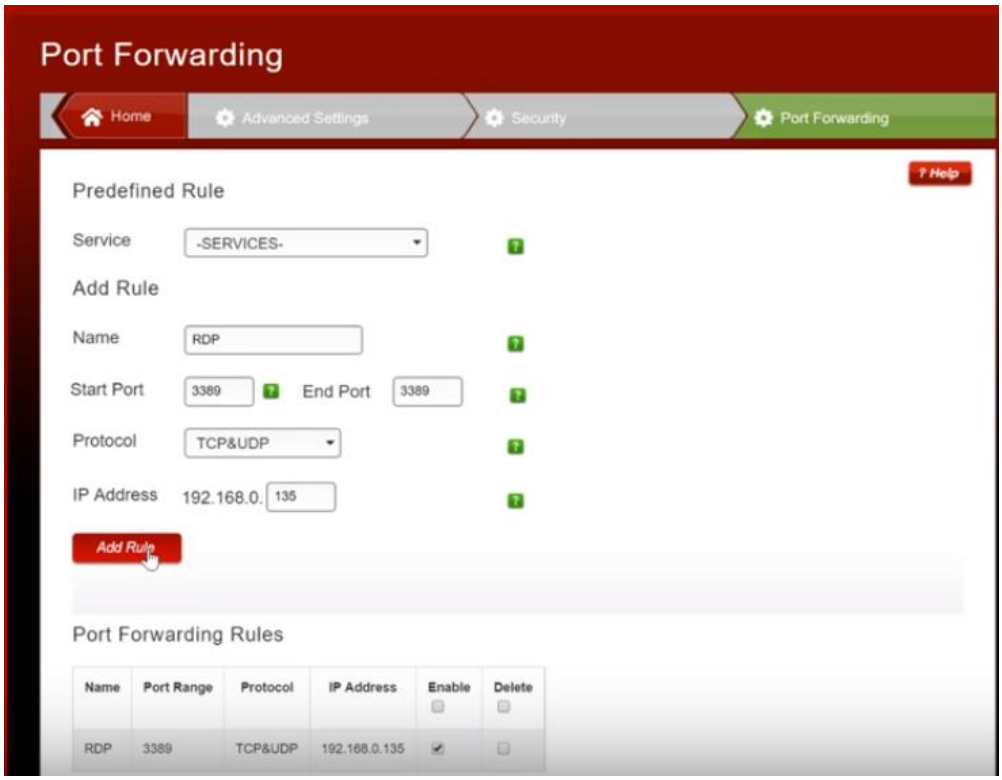
Slika 16: Omogućavanje udaljenog pristupa

Izvor: Autor

Prvo je potrebno konfigurirati kućno računalu da dozvoli *remote desktop* konekciju. Potrebno je ući u *Control Panel>System and Security>System* i odabrati

Remote Settings. U novom prozoru dalje je potrebno odabrati *Allow remote connections to this computer* i klikom na *OK* potvrditi izbor (vidljivo sa slike 16).

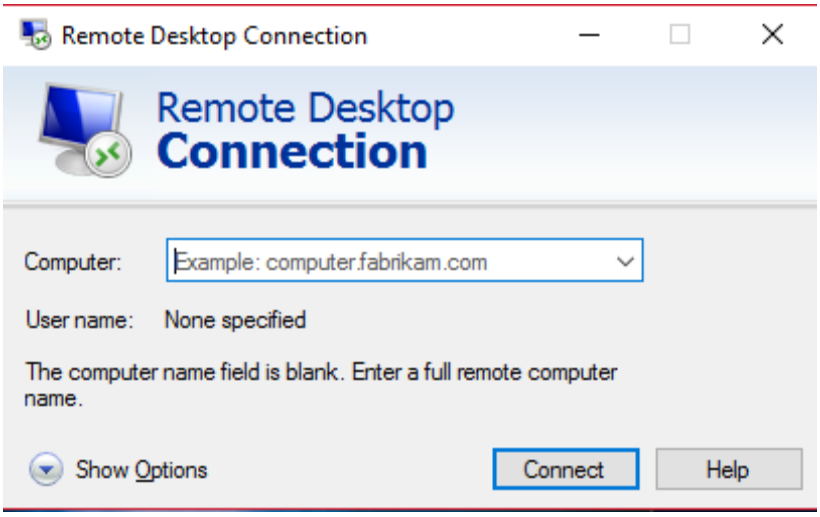
Zatim je potrebno konfigurirati vatrozid na kućnom računalu. Potrebno je ući u *Control Panel>System and Security>Windows Firewall>Allowed Apps*. Tamo je potrebno odabrati *Change Setting* i označiti Remote Desktop kao aplikaciju kojoj je dopuštena komunikacija kroz vatrozid.



Slika 17: Konfiguracija *routera*

Izvor: [40]

Sljedeći korak je konfiguracija kućnog *routera* (vidljivo sa slike 17). U tražilicu *browsera* unosi se IP adresu *routera* nakon čega slijedi autentifikacija. IP adresu *routera* moguće je saznati otvaranjem Windows komandne linije na kućnom računalu i unošenjem naredbe „ipconfig“. IP adresa *routera* tamo je prikazana kao „*default gateway*“ IP adresa. Nakon uspješne autentifikacije potrebno je ući u napredne postavke i odabrati *Sigurnost* te potom *Port forwarding*. Sada je potrebno dodati novo pravilo koje *routeru* govori da dozvoli vezu kojoj je početni i završni *port* 3389, protokol je TCP&UDP, a za IP adresu unosi se adresa kućnog računala [40].



Slika 18: Remote Desktop sučelje

Izvor: Autor

Kada je gotova konfiguracija, na kućnom računalu je potrebno otići na sljedeću *web* stranicu:

http://www.whatsmyip.org/

Na toj *web* stranici može se saznati eksterna IP adresa kućnog računala. Tu adresu potrebno je unijeti u Remote Desktop Connection program (vidljiv sa slike 18) na poslovnom računalu i kliknuti na *Connect*. Zatim slijedi autentifikacija podacima za prijavu koji se koriste na kućnom računalu. Nakon toga je uspješno uspostavljena RDP sesija između poslovnog računala (klijent) i kućnog računala (*server*) [40].

5.3 TeamViewer

Tvrtka TeamViewer osnovana je 2005. te je usmjerena na tehnologije u oblaku te omogućavanje mrežne podrške i suradnje u stvarnom vremenu diljem svijeta. Njihov programski alat za udaljeni pristup dostupan je za besplatan *download* za nekomercijalne svrhe na sljedećem *linku* (korištena verzija je v12.0.81460):

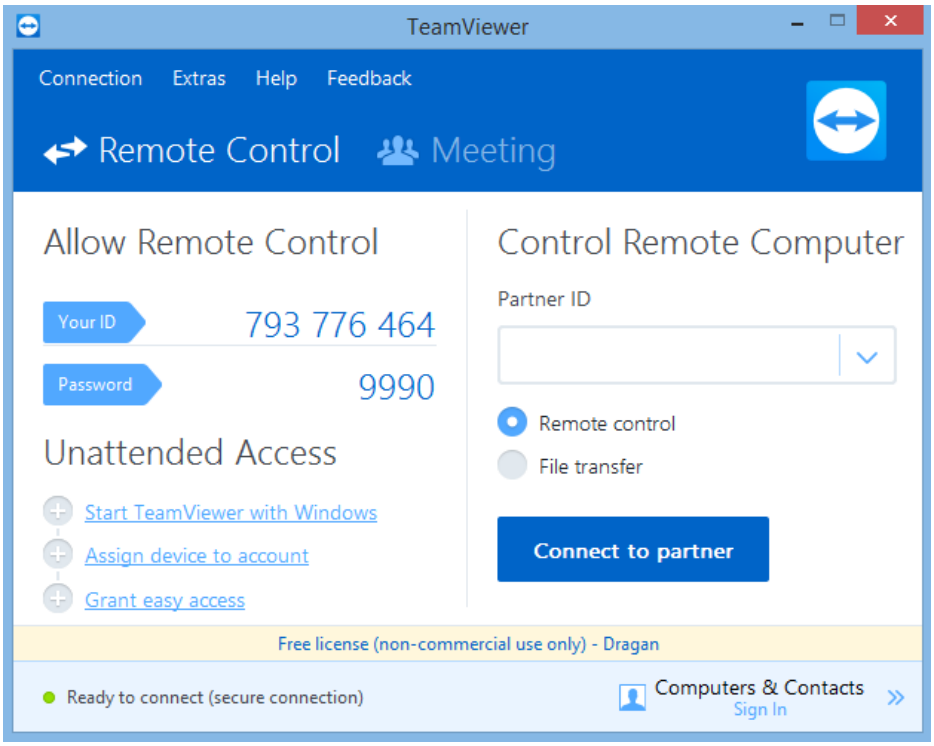
https://www.teamviewer.com/hr/download/windows/

Program je interoperabilan, omogužuje povezivanje s osobnog računala na osobno računalo, s mobilnog uređaja na osobno računalo, s osobnog računala na mobilni uređaj i povezivanje s mobilnog uređaja na mobilni uređaj. Podržava sustave Windows, Mac OS, Linux, Chrome OS, iOS, Android, Windows Universal Platform i BlackBerry [41].

TeamViewer program za udaljeni pristup, instaliran je na preko milijardu uređaja (za svaki se uređaj generira jedinstveni ID), stvara 750.000 novih ID-ova svakoga dana te ima preko 20 milijuna uređaja na mreži u svakom trenutku [41].

Protokol na kojem radi TeamViewer je njegov vlastiti (engl. *proprietary*) protokol koji je kombinacija VNC (RFB) i RDP protokola. *Port* koji se primarno koristi je TCP/UDP 5983, ukoliko to nije moguće koristit će TCP *port* 443 ili 80, ali će veza tada biti sporija i nepouzdanija [44].

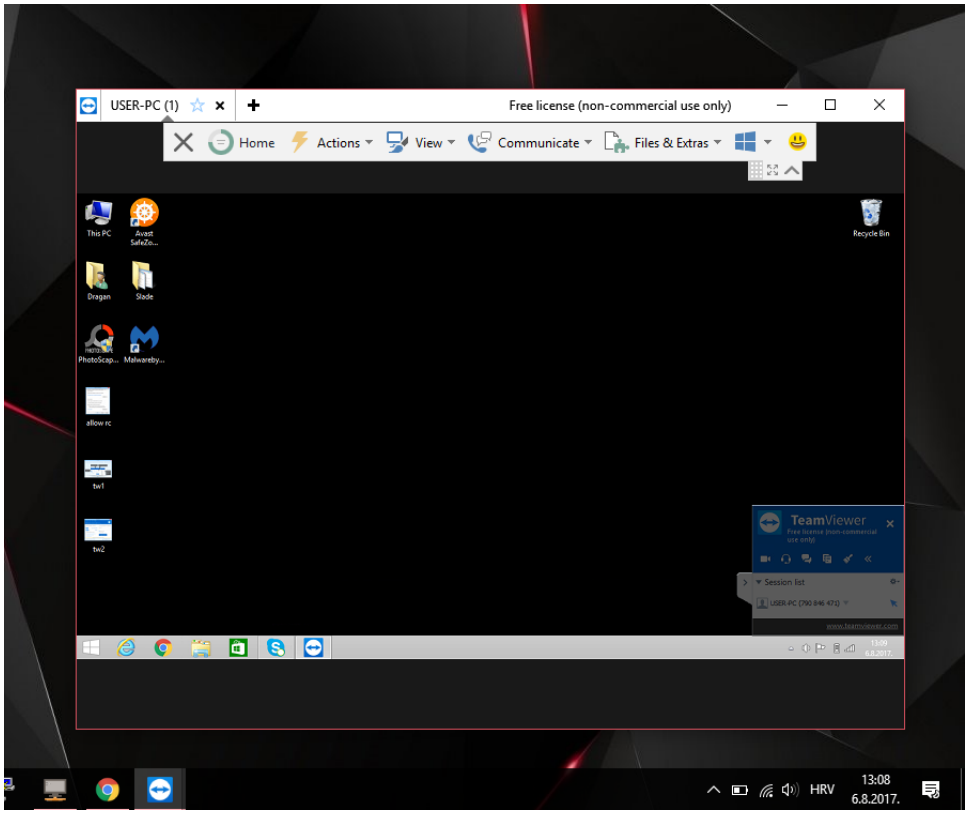
Program nije potrebno konfigurirati, jednostavan je za korištenje. Učinkovito koristi dostupnu pojasnu širinu kako bi pružio optimalne performanse. Upotrebljava RSA 2048 izmjenu javnih i privatnih kodova te 256-bitno AES kodiranje sesije od kraja do kraja. Osim ID-a partnera, TeamViewer generira lozinku za sesiju koja se mijenja prilikom svakog pokretanja softvera kako bi osigurao dodatnu zaštitu od neovlaštenog pristupa udaljenom sustavu. Funkcije u kojima je sigurnost važna, poput funkcije prijenosa datoteka, potrebno je dodatno, ručno potvrditi s udaljenog partnera.



Slika 19: Sučelje TeamViewera

Izvor: Autor

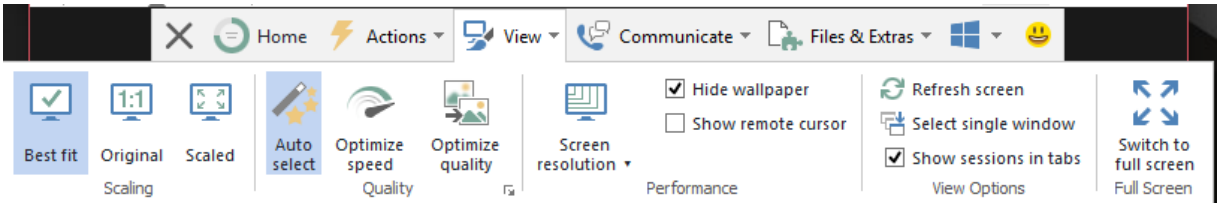
Sa slike 19 vidljivo je sučelje TeamViewera. Osim udaljenog upravljanja računalom (engl. *Remote Control*) postoji i *Meeting* opcija za potrebe videokonferencije. Svakom se računalu sa upaljenim TeamViewer klijentom dodjeljuje jedinstveni ID i jednokratna lozinka. Lozinka se mijenja svakim novim pokretanjem programa. Ovdje prikazani ID i lozinka odgovaraju udaljenom računalu kojem se želi pristupiti. Zato je potrebno ove podatke unijeti u lokalno računalo. Prvo se unosi ID udaljenog računala u *Partner ID* polje lokalnog računala, a potom i lozinka koja se zatražuje naknadno.



Slika 20: TeamViewer sesija

Izvor: Autor

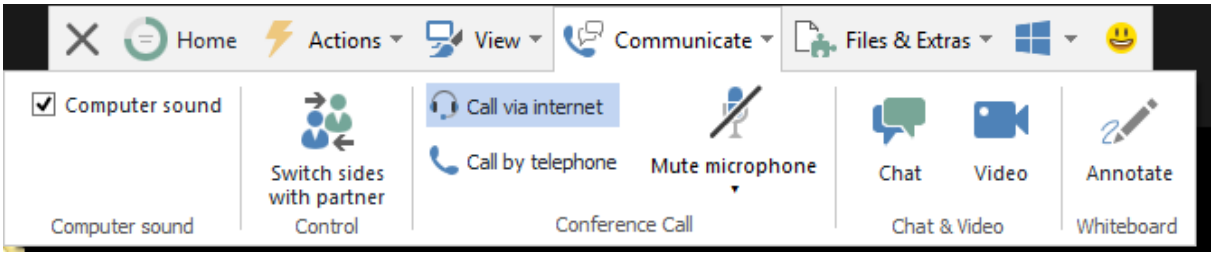
Nakon obavljenog postupka autentifikacije, kreira se sesija sa udaljenim računalom (vidljivo sa slike 20) najbolje usporediva sa RealVNC VNC sesijom. Ovdje je također na raspolaganju alatna traka pri vrhu prozora programa. Međutim, TeamViewer nudi multifunkcionalnost kakvu rijetko koji programi za udaljeni pristup imaju, a to ostvaruje kroz integraciju različitih usluga i mogućnosti. Najznačajnije opcije u alatnoj traci zasigurno su *View, Communicate* i *Files & Extras*.



Slika 21: *View* opcija u alatnoj traci TeamViewera

Izvor: Autor

View opcija (vidljivo sa slikem 21) omogućuje među ostalim upravljanje rezolucijom, ručno osvježavanje slike, optimizaciju brzine i kvalitete slike.



Slika 22: *Communicate* opcija u alatnoj traci TeamViewera

Izvor: Autor

Communicate opcija (vidljivo sa slike 22) omogućuje VOIP poziv, telefonski poziv, videokonferenciju, *chat* i interakciju crtanjem (engl. *Annotate*). Osim toga, komunikacijski partneri mogu zamijeniti strane tako da klijent prijeđe u ulogu *servera* i *server* u ulogu klijenta.

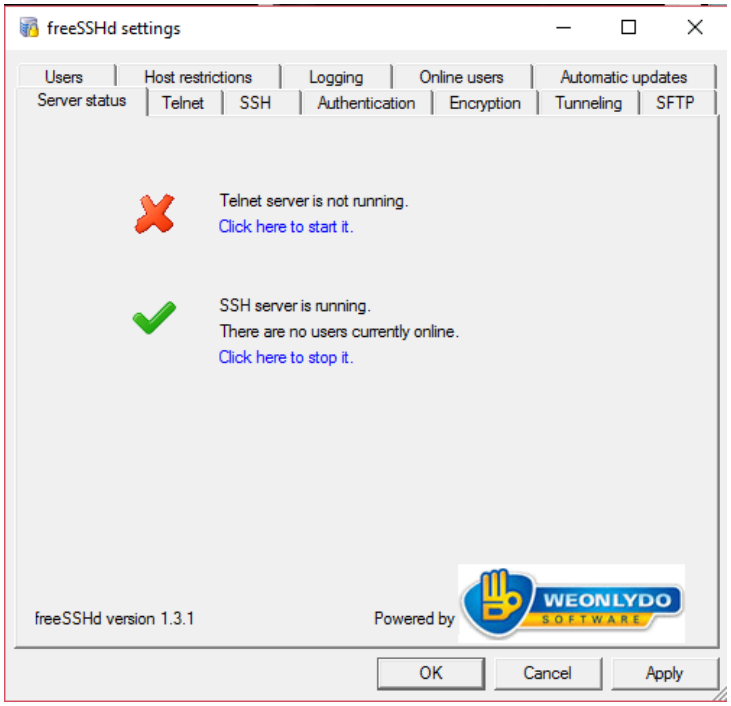
Files & Extras opcija omogućuje transfer datoteka između klijenta i *servera*.

5.4 FreeSSHd

Kako bi se nakon tri *remote desktop* rješenja krenulo sa predstavljanjem i testiranjem SSH klijenata (PuTTy, Tera Term i TTY Emulator), prvo je potrebno uspostaviti SSH *server* na računalu na koje se želi udaljeno pristupiti. FreeSSHd je FTP (engl. File Transfer Protocol) / SFTP (engl. Secure FTP) *server* koji omogućuje korisniku udaljeni pristup datotekama putem TCP/IP mreže kao što je Internet. Za razliku od FTP, FTPS i SFTP protokola, omogućuje sigurnu i snažnu enkripciju. Program je dostupan za *download* na sljedećem *linku* (korištena verzija je 1.3.1):

http://www.freesshd.com/?ctt=download

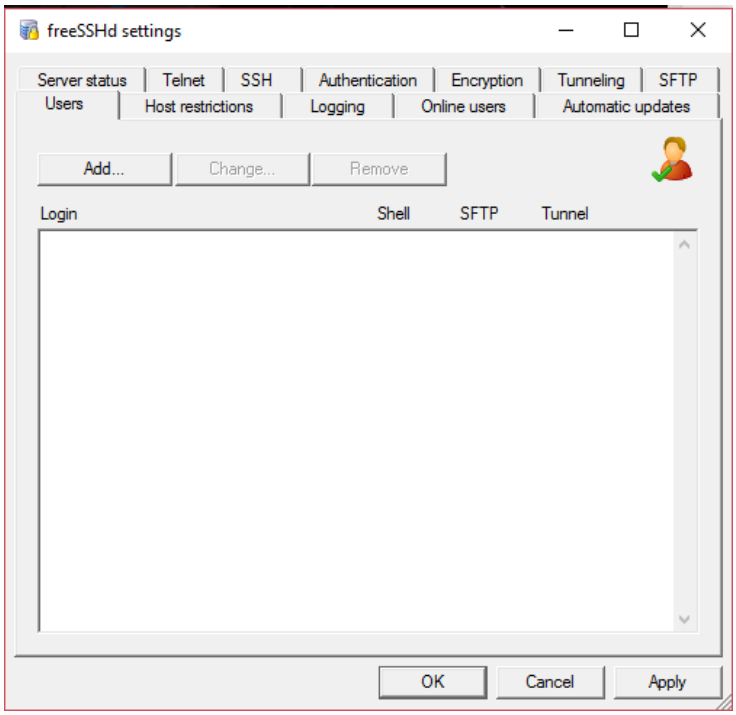
Radi na Windows operativnim sustavima.



Slika 23: Sučelje FreeSSHd *servera*

Izvor: Autor

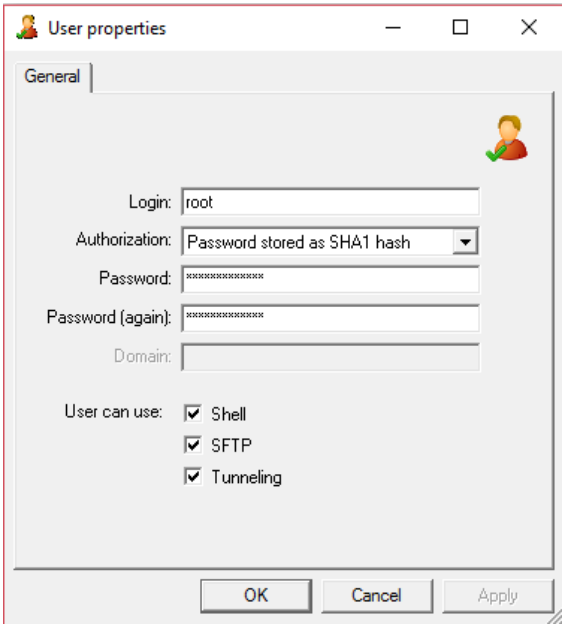
Sa slike 23 vidljivo je sučelje programa. Pri pokretanju programa korisnika se pozicionira na karticu *Server Status*. Vidljivo je da je SSH *server* aktivan.



Slika 24: *User* kartica SSH *servera*

Izvor: Autor

Sada je potrebno unijeti podatke jednog ili više korisnika koji mogu pristupiti ovom *serveru*. To se postiže pozicioniranjem na karticu *Users* i klikom na *Add* (vidljivo sa slike 24).



Slika 25: Dodavanje novog korisnika

Izvor: Autor

Sada je potrebno upisati proizvoljno ime i šifru ako je odabrana metoda autorizacije kao na slici (još je moguće prijaviti se koristeći Windows podatke za prijavu u OS i pomoću javnog ključa). Nakon toga potrebno je označiti *Shell*, *SFTP* i *Tunneling* kao na slici i kliknuti *OK*. Sada je dodan korisnik „root“ i sa svojim podacima ima ovlaštenje za pristup računalu na kojem je instaliran FreeSSHd server (vidljivo sa slike 25).

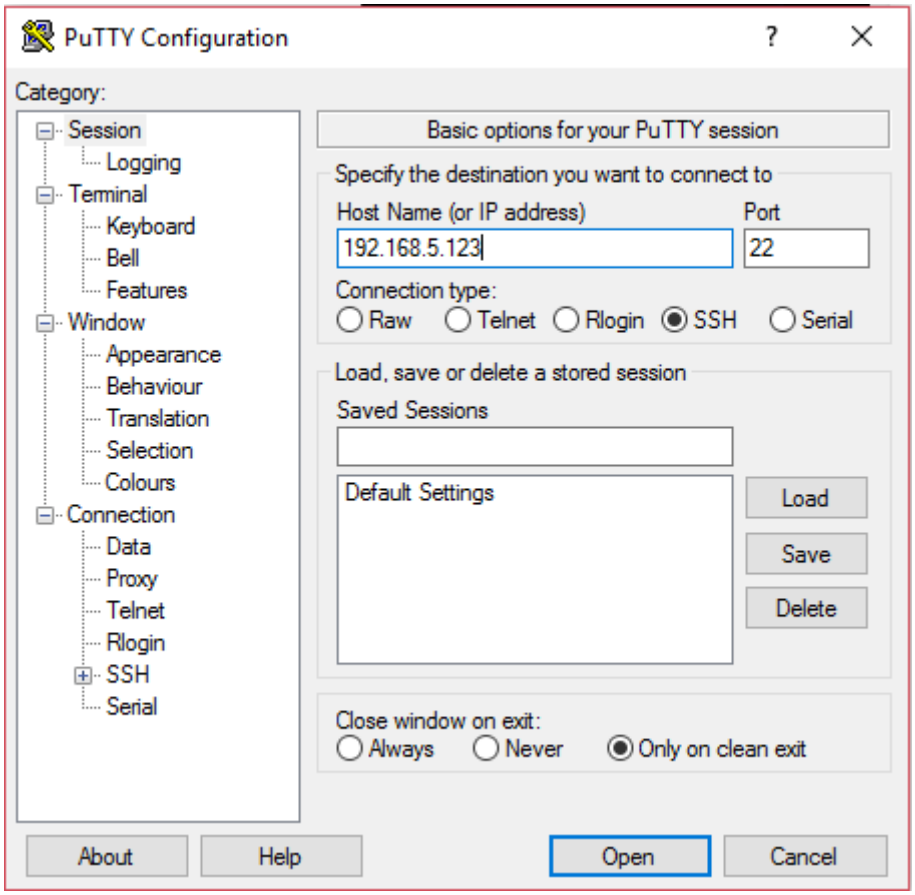
Sljedeća značajka koju treba omogućiti je *forwardiranje portova* klikom na karticu *Tunneling* te potom na *Allow port forwarding*. *Forwardiranje* je zanimljiva funkcionalnost SSH koja će biti testirana u sljedećem podpoglavlju. Osim toga nije potrebno dalje konfigurirati SSH *server*. Mogu se samo nabrojati ostale mogućnosti poput blokiranja pristupa definiranim računalima (IP adresama) u *Host Restrictions*, odabira enkripcijskog algoritma *u Encryption* itd.

5.5 PuTTy

PuTTY je najpopularniji *open source* SSH i telnet[33] klijent. Razvio ga je Simon Tatham za Windows platformu. Dostupan je za *download* na sljedećem *linku* (korištena verzija je 0.70) :

[33] Telnet je zastarjeli protokol koji je korisniku jednog računala osiguravao sesiju za korištenje komandne linije na drugom računalu.

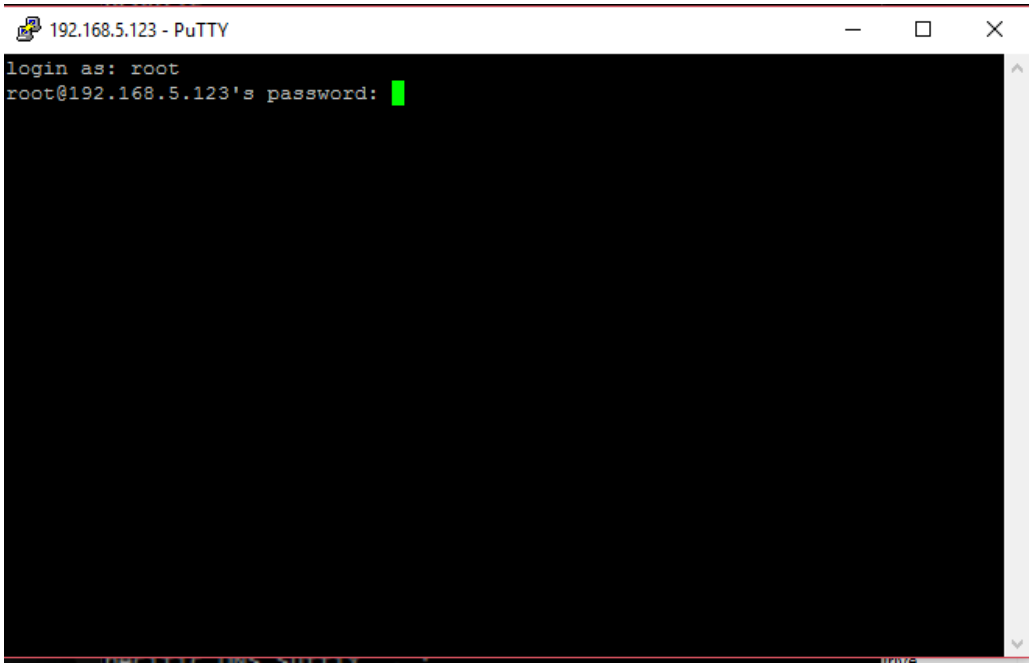
https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html



Slika 26: PuTTy sučelje

Izvor: Autor

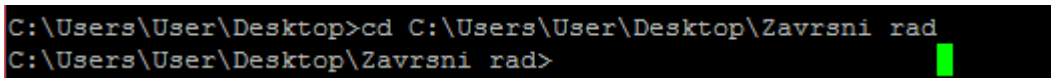
Sa slike 26 vidljivo je sučelje PuTTy SSH klijenta. U polje *Host Name* unosi se IP adresa računala na kojem je instaliran SSH *server* kako bi se inicirala SSH sesija. *Default port* za SSH je 22. Mogu se spremiti podaci o sesiji klikom na *Save* te potom klikom na *Open* otvoriti konzolu.



Slika 27: Konzola PuTTy programa

Izvor: Autor

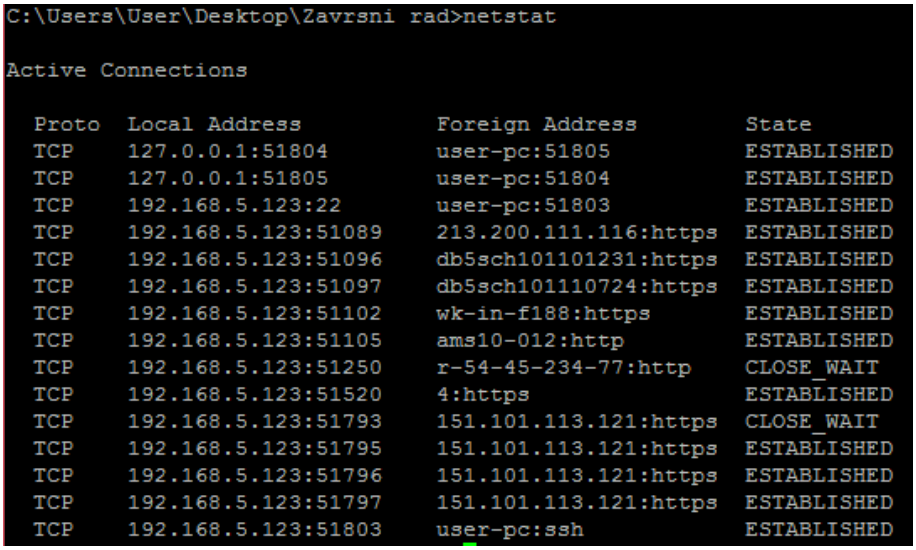
Nakon klika na *Open* otvorila se konzola programa (vidljivo sa slike 27). Sada je potrebno upisati podatke za autorizaciju. Nakon uspješne prijave otvorena je SSH sigurna veza sa *server* računalom te se sada mogu unositi tzv. komande za upravljanje istim računalom.



Slika 28: Ispis komande „cd“

Izvor: Autor

Ukoliko korisnik želi pristupiti određenoj datoteci na *desktopu* ili drugoj lokaciji koristit će komandu „cd“ i unijeti definiranu lokaciju iza nje (vidljivo sa slike 28).



Slika 29: Ispis komande „netstat“

Izvor: Autor

Unosom komande „netstat“ ispisat će se sve aktivne konekcije računala (vidljivo sa slike 29).

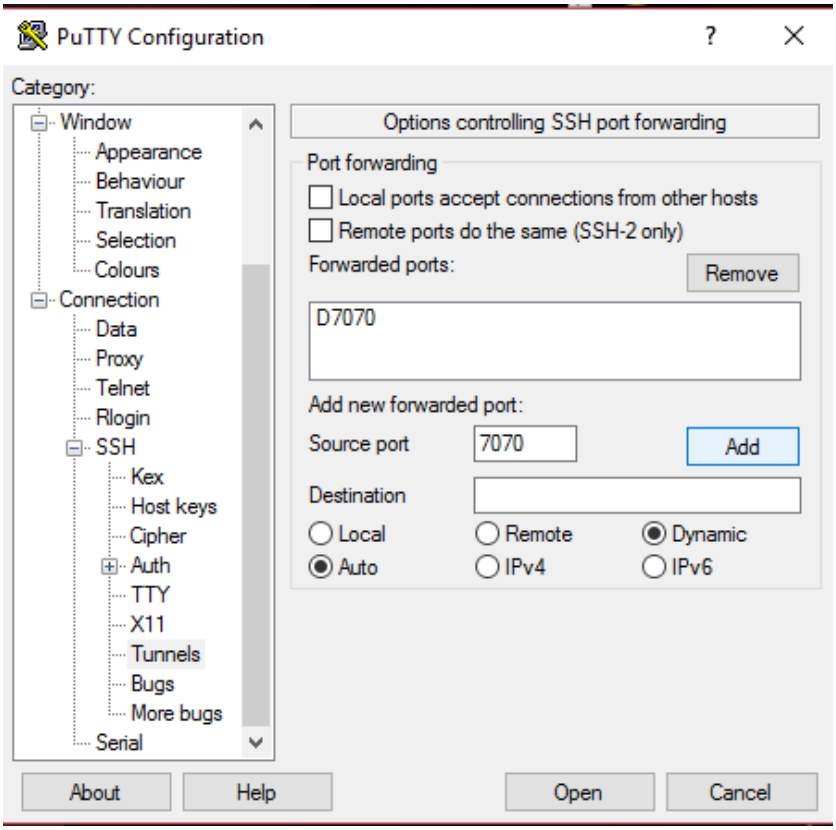
Sve naredbe dostupne su na izvoru [42]. U primjerima se koriste Windows naredbe, ali ukoliko bi *server* bio uspotavljen na Linux OS-u koristile bi se Linux naredbe poput „ls“ (*list*) naredbe koja ne bi funkcionirala unosom u konzolu u ovom konkretnom slučaju.

SSH tuneliranje/forwardiranje

U ovom dijelu ukratko je objašnjeno dinamičko *forwardiranje* i kako se isto izvodi uz pomoć SSH klijenta (PuTTy, Tera Term, TTY Emulator...) i *web browsera* npr. Firefoxa.

Forwardiranje portova je korisna značajka SSH. Omogućuje kriptiranje mrežnog prometa nesigurnih protokola poput telneta i SMTP-a koji koriste TCP transportni protokol kao i SSH. *Forwardiranje portova* se još naziva i tuneliranje jer se ovim procesom postiže uspostavljanje sigurnog tunela kroz koji mogu putovati druge TCP/IP konekcije [43].

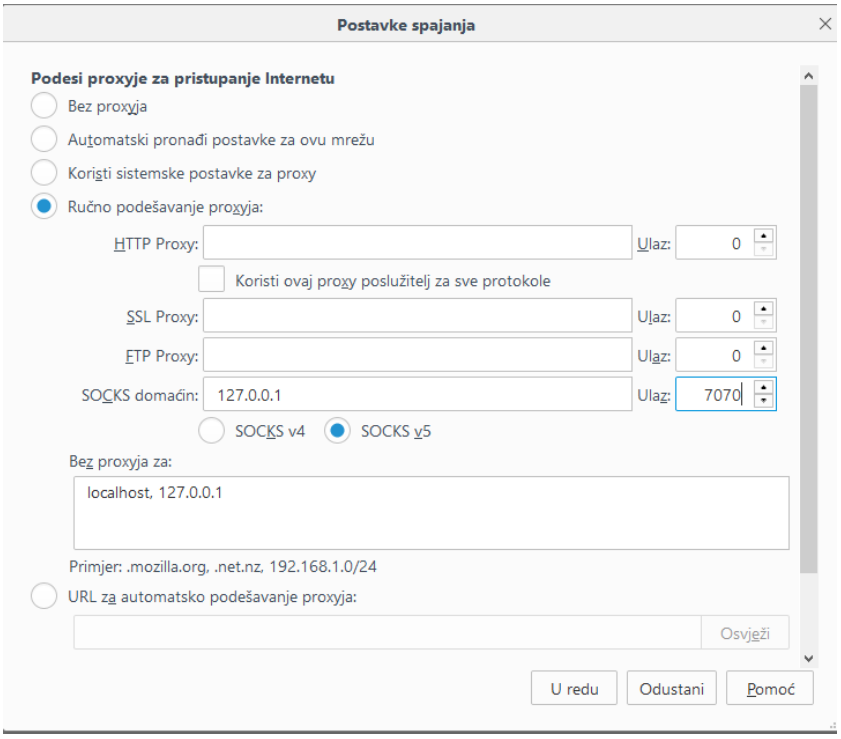
SSH klijenti pružaju mogućnost dinamičkog *forwardiranja* čime se postiže pristup različitim TCP *serverima* zaobilaženjem vatrozida koji je konfiguriran da blokira direktan pristup istima [46].



Slika 30: Dinamičko forwardiranje

Izvor: Autor

Za tuneliranje koristit će se programski alat PuTTy. Potrebno je u lijevom izborniku odabrati *SSH* te potom *Tunnels*. Za *Source port* potrebno je upisati proizvoljno velik broj poput 7070 ili 8080 i slično. Uz to je potrebno označiti *Dynamic* i kliknuti *Add* te povezati se na *server* klikom na *Open* (vidljivo sa slike 30).



Slika 31: Postavke spajanja u Firefoxu

Izvor: Autor

Nakon autentifikacije na *server*, potrebno je pokrenuti Mozilla Firefox *web browser*. Nakon toga u *browseru* se odabiru *Postavke>Napredno>Postavke spajanja*. Potom je potrebno ući u *Ručno podešavanje* te izvršiti kratku konfiguraciju. U *SOCKS domaćin* polje upisuje se „127.0.0.1“ tj. adresa *localhosta* (vidljivo sa slike 31).

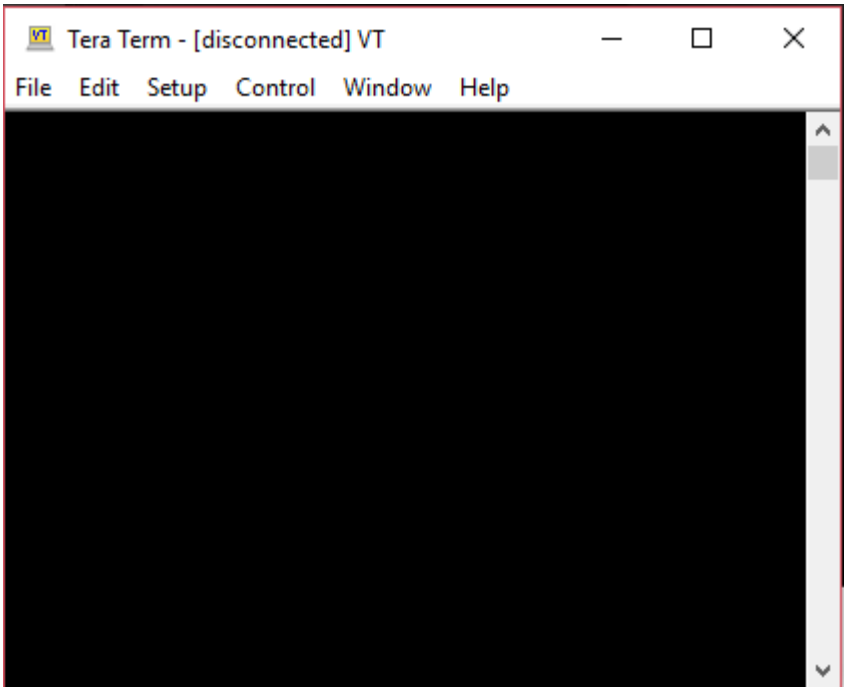
SOCKS (engl. *Secure Sockets*) je protokol koji preusmjerava promet prema *proxy serveru*. *Proxy server* se ponaša kao posrednik u komunikaciji između klijentskog računala i udaljenog *web servera*. Postiže se cilj maskiranja komunikacije i osiguranja anonimnosti na *webu*.

U polje *Ulaz* upisuje se adresa porta 7070. Klikom na *U redu* završen je postupak konfiguracije te je IP adresa računala izmjenjena. Na ovaj način mogu se izbjeći restrikcije nametnute od vatrozida.

5.6 Tera Term

Tera Term je besplatni *open source* programski alat koji služi kao simulator terminala. Podržava SSH1, SSH2, telnet i serijske veze. Po značajkama najsličniji je PuTTy programu, moguće je tuneliranje, sigurno kopiranje (SCP), više vrsta autentifikacije, itd. Dostupan je za *download* na sljedećem *linku* (korištena verzija je 4.95):

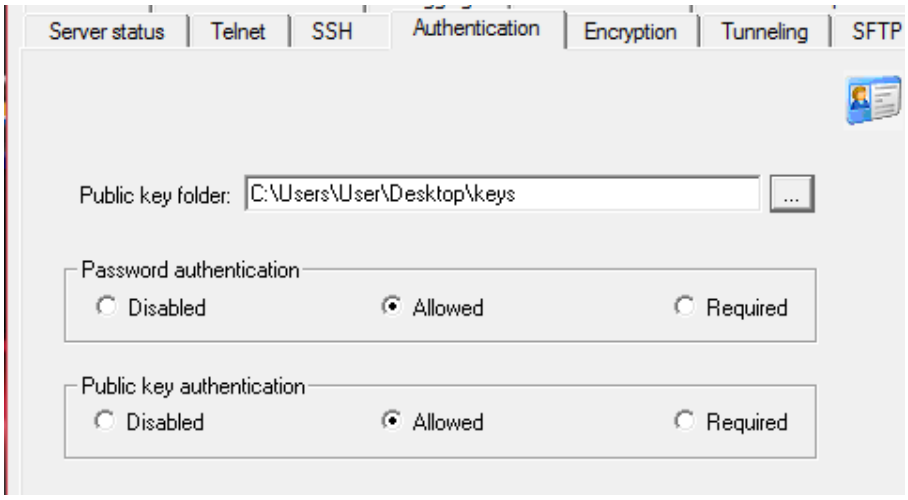
https://osdn.net/projects/ttssh2/releases/



Slika 32: Sučelje Tera Term programa

Izvor: Autor

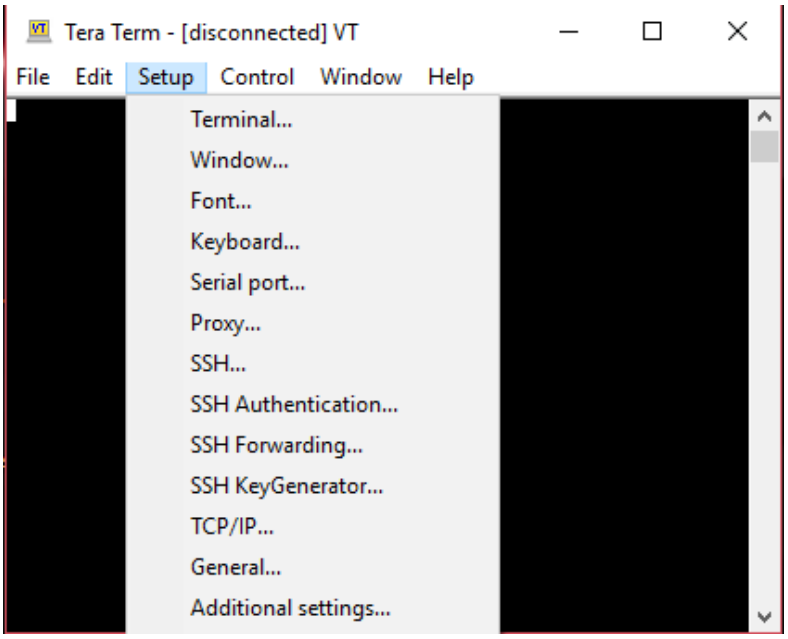
Na slici 32 vidljivo je sučelje programa Tera Term. Klikom na *Setup* dostupne su različite funkcionalnosti. Zbog velike sličnosti sa PuTTy programom neće se ponovno ulaziti u tuneliranje nego će se pokušati udaljeno povezivanje na SSH *server* koristeći privatni i javni ključ. Ta vrsta autentifikacije dostupna je i na PuTTy-u. Postupak je također sličan.



Slika 33: FreeSSHd *Authentication* kartica

Izvor: Autor

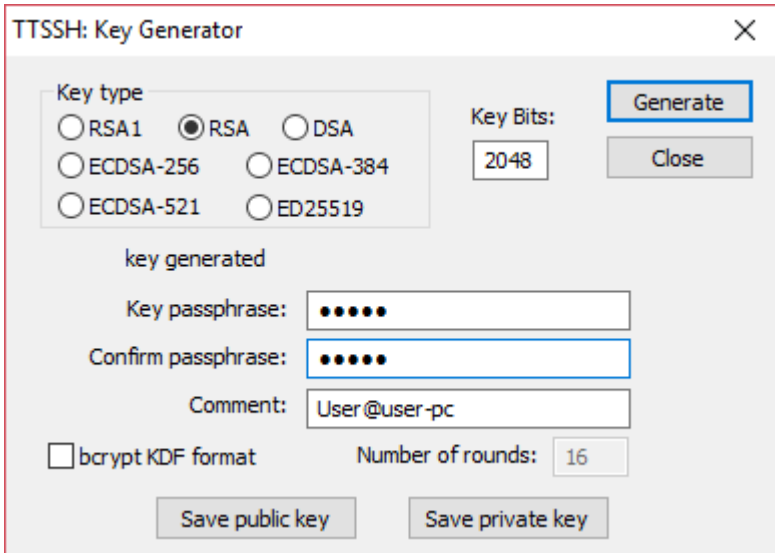
Prvo je potrebno učiniti preinake u *Authentication* kartici FreeSSHd servera. Treba se pobrinuti da je dopuštena autentifikacija javnim ključem te uz to definirati datoteku koja će poslužiti za čuvanje javnog i privatnog ključa. Ovdje je napravljena nova datoteka naziva „keys“ na radnoj površini za tu potrebu. Uz to u kartici *Users* dodaje se novog korisnika naziva „root1“ te se za autentifikacijsku metodu navodi „*Public key*“.



Slika 34: Tera term *Setup* traka

Izvor: Autor

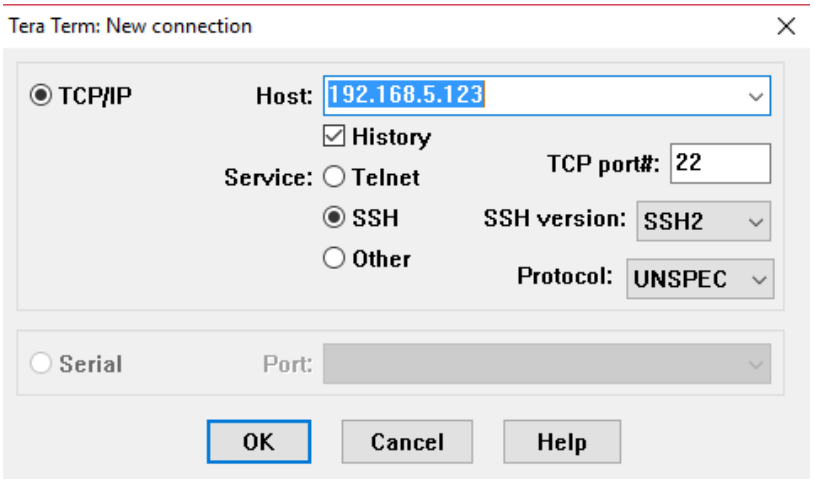
Zatim se u Tera Term *Setup* kartici, vidljivoj sa slike 34, odabire *SSH KeyGenerator*.



Slika 35: Tera Term *KeyGenerator*

Izvor: Autor

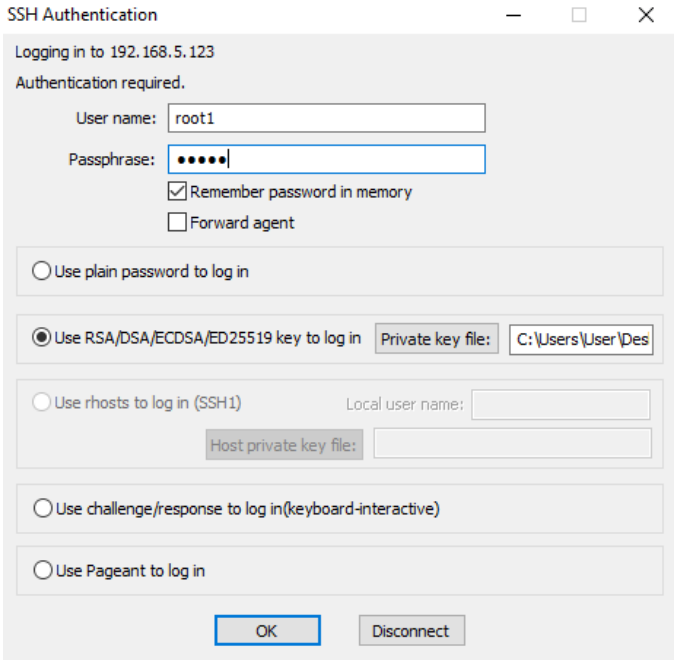
Otvorit će se *Tera Term Key Generator* za stvaranje javnog i privatnog ključa, vidljiv sa slike 35. Sada se odabire vrsta enkripcije. Ovdje je odabrana RSA. Može se odabrati bilo koja enkripcija osim RSA1 jer je ona vezana za SSH1 protokol, dok su ostale vezane za SSH2. Nakon odabira enkripcije potrebno je kliknuti na *Generate*. Zatim se unosi *passphrase*. To je šifra koja će se koristiti uz priloženi ključ prilikom autentifikacije. Nakon toga sprema se javni ključ u datoteku „keys“ i naziva ga se „root1“ te privatni ključ također u istu datoteku te se njega naziva „root1Private.ppk“.



Slika 36: Tera Term nova konekcija

Izvor: Autor

Sada se u Tera Term sučelju odabere *File>New Connection*. Upiše se IP adresa SSH *servera* i klikne *OK* (vidljivo sa slike 36):



Slika 37: Tera Term autentifikacijski prozor

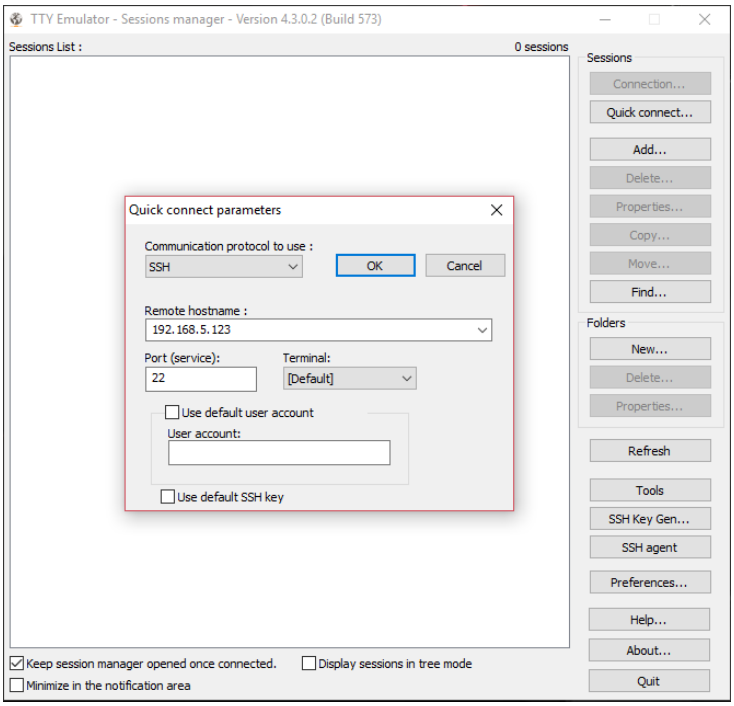
Izvor: Autor

Sada je u prozoru za autentifikaciju, vidljivom sa slike 37, potrebno upisati ime i *passphrase* te odabrati *Use RSA key to log in*. Označi se datoteku privatnog ključa i stisne *OK*. Proces je tada završen i korisnik „root1“ je prijavljen na *server* te može izvršavati komande.

5.7 TTY emulator

TTY emulator je besplatna terminal emulator aplikacija koja se može ponašati kao klijent za SSH, telnet, rlogin i sl. Program je razvijen za Microsoft Windows. Dostupan je za *download* na sljedećem *linku* (trenutna verzija 4.3.0.2):

http://www.ttyemulator.com/index.php?id=7



Slika 38: Sučelje TTY emulator

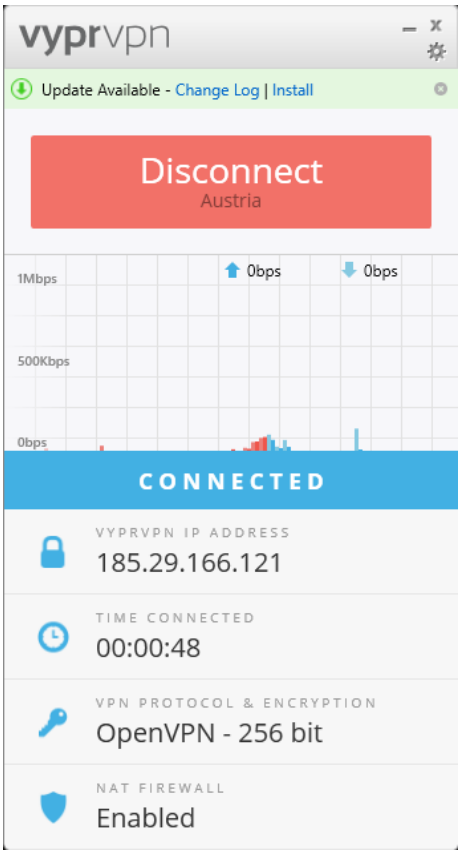
Izvor: Autor

Neke od značajki ovog programa su podrška za SSH1 i SSH2, integrirani generator ključeva i agent, šifriranje (DES, 3DES, Blowfish), omogućeno tuneliranje. Veza sa SSH *serverom* postiže se klikom na *Quick Connect* i upisivanjem adrese SSH *servera*. Sam program dosta je sličan prethodno obrađenim SSH klijentima (vidljivo sa slike 38).

5.8 Vypr VPN

Sljedeći na redu za testiranje su VPN klijenti. Vypr VPN proizvod je tvrtke Golden Frog. Prema tvrdnjama osnivača, tvrtka je pokrenuta kao odgovor na zloglasnu sobu 614a u San Franciscu, gdje je NSA obavljala nadzor AT&T-ove telekomunikacijske mreže. Kada su dostavili dokaze federalnoj komunikacijskoj komisiji (FCC) o ovom slučaju nelegitimnog prisluškivanja i kršenja privatnosti korisnika bili su ignorirani. To im je poslužilo kao motiv da među ostalim kreiraju i ovo po mnogima ponajbolje VPN rješenje. Program je dostupan za *download* na sljedećem *linku* (trenutna verzija je 2.9.6.7227):

https://www.goldenfrog.com/vyprvpn/buy-vpn

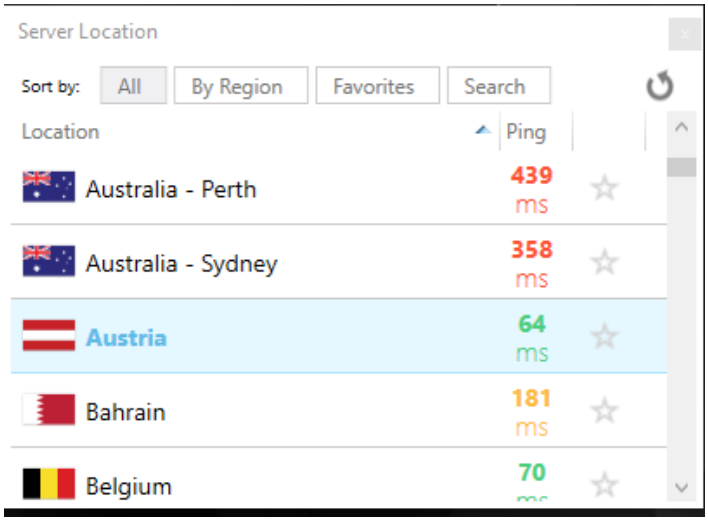


Slika 39: Sučelje Vypr VPN

Izvor: Autor

Sa slike 39 vidljivo je sučelje programa. Sučelje je jednostavno za korištenje, dostupne su informacije o eksternoj IP adresi računala kada ono nije spojeno na VPN *server*. Kada se korisnik spoji na neki od dostupnih VPN *servera* postaje anoniman na Internetu jer se sav promet prosljeđuje preko tog posredničkog *servera*. Tada adresa VPN *servera* postaje eksterna IP adresa računala jer je VPN njegov posrednik u komunikaciji na Internetu. Također, tu su i informacije o protokolu i enkripciji, vremenu koje je računalo provelo na vezi sa *serverom*, količini prenesenih

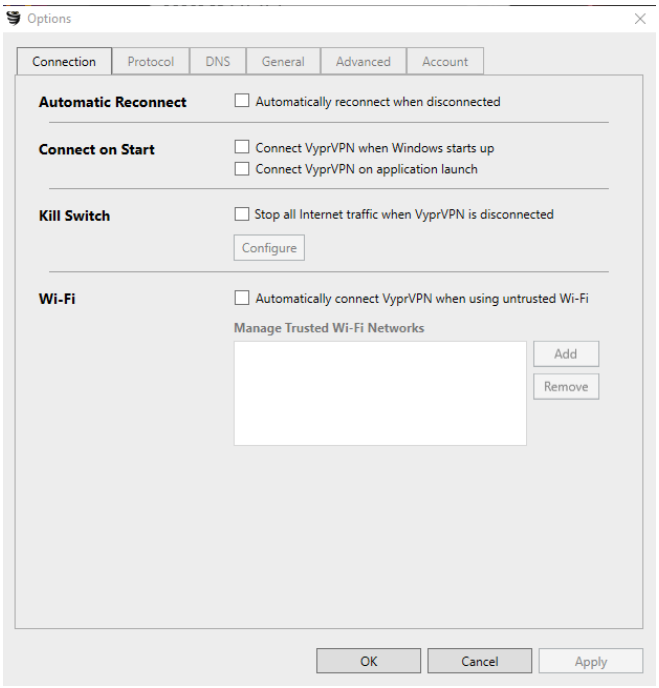
podataka u *uplinku* i *downlinku* te informacija o uspostavljenom NAT vatrozidu između VPN *servera* i Interneta za filtraciju eventualnih malicioznih paketa.



Slika 40: Odabir lokacije VPN *servera*

Izvor: Autor

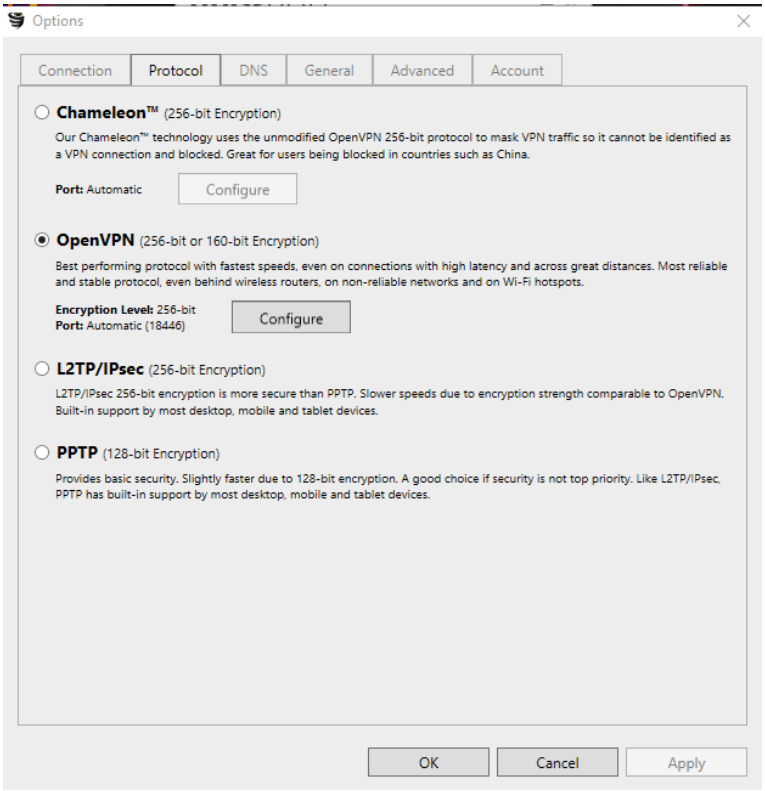
Vypr VPN pruža mogućnost odabira između više od 700 VPN *servera* u preko 50 država. Tvrtka barata sa ukupno više od 200 000 različitih IP adresa. Kada se odabere određena država dostupna je informacija o kašnjenju paketa od klijenta do *servera* u milisekundama. U ovom primjeru klijent je spojen na VPN *server* u Austriju zbog povoljno malog kašnjenja (vidljivo sa slike 40).



Slika 41: *Connection* opcije Vypr VPN-a

Izvor: Autor

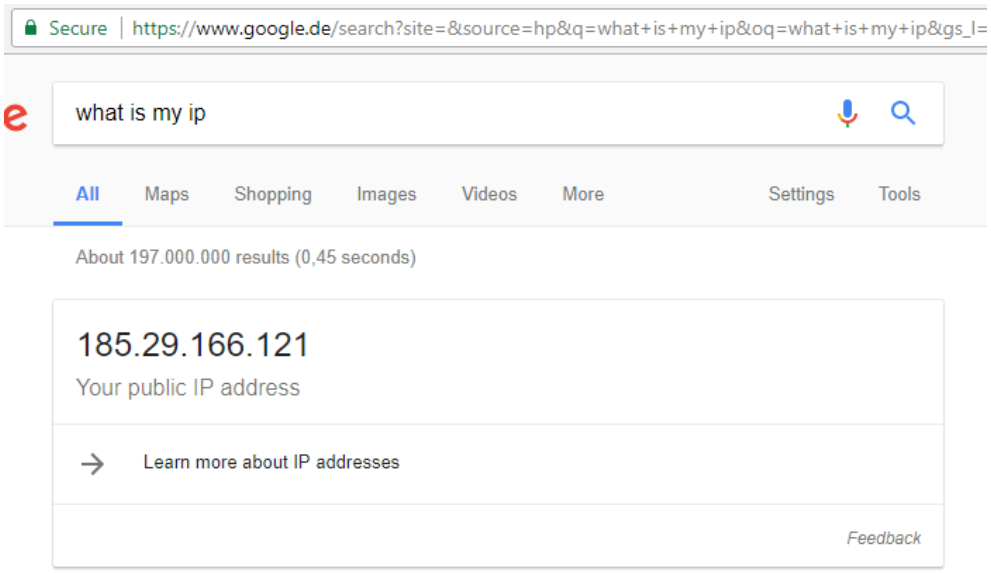
U opcijama veze Vypr VPN-a vidljivima sa slike 41, pružaju se standardne mogućnosti koje mora imati gotovo svaki VPN na tržištu a krucijalne su za održavanje sigurne veze. To je npr. automatsko ponovno povezivanje nakon ispada s veze, povezivanje na VPN odmah nakon podizanja operativnog sustava, funkcija osigurača (zaustavlja se Internet promet ako je Vypr VPN isključen) i automatsko povezivanje na VPN kada se koristi javna Wi Fi mreža.



Slika 42: Izbor protokola u Vypr VPN

Izvor: Autor

Samo određeni VPN klijenti na tržištu daju mogućnost izbora protokola i enkripcije sigurne veze. Vypr VPN ima upravo tu mogućnost izbora između PPTP, L2TP/IPsec, OpenVPN i Chameleon (vidljivo sa slike 42). Chameleon je protokol razvijen od Golden Froga. Nameće se kao najbolji izbor uz OpenVPN. Golden Frog tvrdi da se upotrebom Chameleon protokola efektivnije maskira identitet na Internetu. Teže je autoritetima koji nadziru Internet promet otkriti da se radi o VPN vezi. To je posebno korisna značajka za korisnike u npr. Kini koji se bore s restrikcijama određenih sadržaja koje je njihova država označila kao neprimjerene.



Slika 43: *Web* stranica za provjeru IP adrese

Izvor: Autor

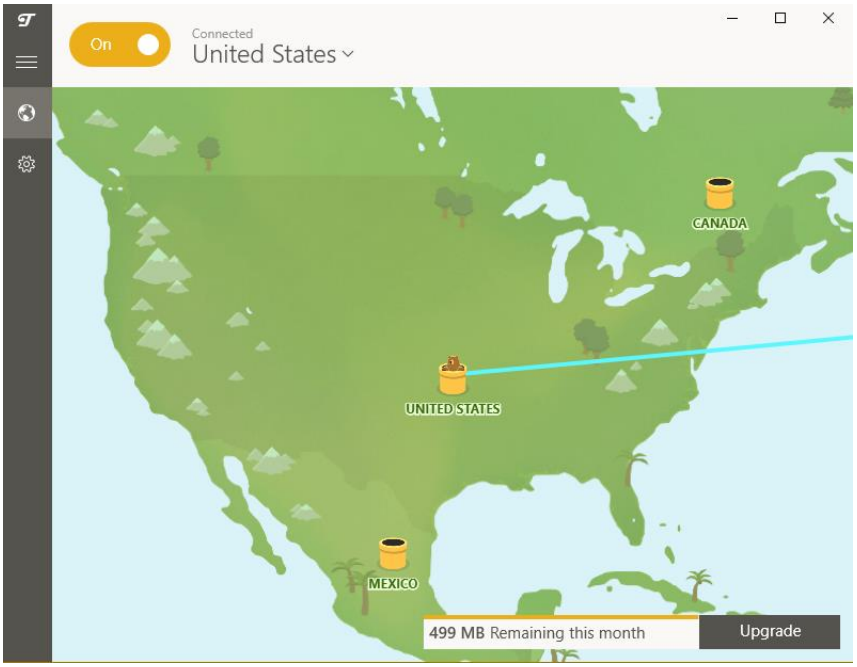
Za kraj testiranja Vypr VPN-a pokrenut je *web browser* te se provjerava IP adresa. Vidljivo je da je to upravo adresa u Austriji kako je i prikazao programski alat te je također vidljivo sa slike 43 da je Google automatski promjenio svoju tražilicu na njemačku verziju.

5.9 TunnelBear

TunnelBear je VPN proizvod tvrtke TunnelBear Inc. Dostupan je za platforme Android, Microsoft Windows, Mac OS, iOS. U besplatnoj verziji postavljeno je ograničenje VPN prometa od 500 MB. *Link* za *download* (trenutna verzija 3.0.36.9):

https://www.tunnelbear.com

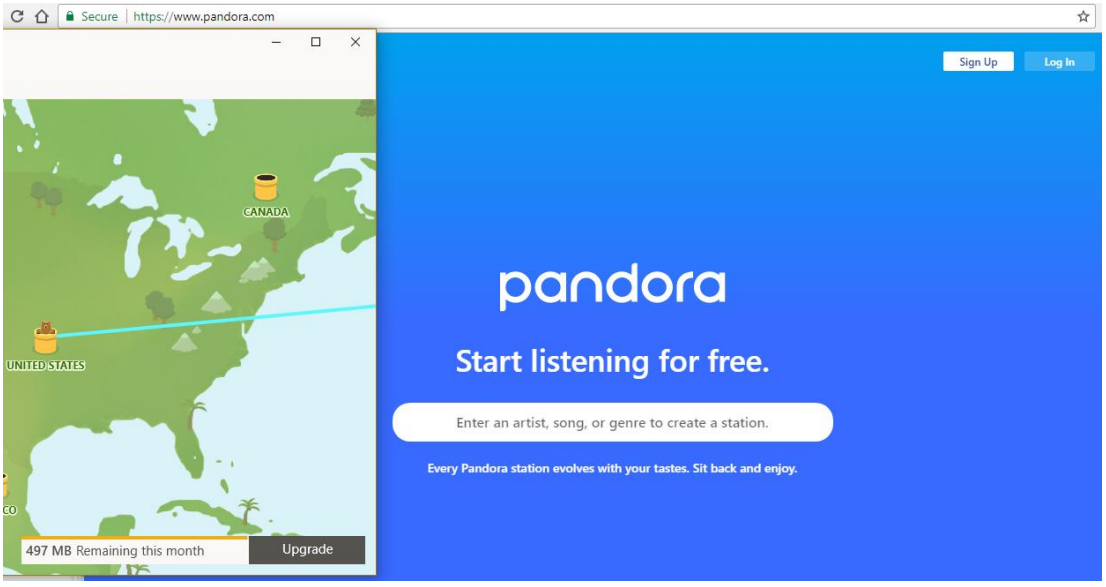
TunnelBear na izbor daje neku od 20 različitih država u kojima ima *servere*. Konfiguacijske mogućnosti nisu zavidne. Osim funkcije *killswitcha*, automatskog povezivanja i ostalih mogućnosti povezivanja ne nudi se mnogo više. Protokol za Windows verziju je OpenVPN i ne može se mijenjati. Koristi se AES 256-bitna enkripcija.



Slika 44: VPN veza sa *serverom* u SAD-u

Izvor: Autor

Na slici 44 vidljivo je sučelje TunnelBear VPN klijenta. Korištenje je maksimalno pojednostavljeno. Sve što je potrebno je odabrati državu i pritisnuti *Connect*. Nakon toga, dogode se promjene na grafičkom sučelju koje sugeriraju da je ostvarena VPN veza.



Slika 45: Omogućen pristup Pandora radiju

Izvor: Autor

Za kraj testiranja praktični primjer. Određene *web* stranice imaju postavljene georestrikcije, odnosno na temelju IP adrese, korisnicima iz stranih zemalja odbija se

pristup sadržaju. Pandora radio je *web* stranica za *streaming* audio sadržaja kojoj pristupiti mogu samo računala iz SAD-a. Korištenjem TunnelBearovog servera u SAD-u uspijeva se zaobići te restrikcije i pristupiti *web serveru* (vidljivo sa slike 45).

5.10 Hide.me

Za potrebe usporedne analize u sljedećem poglavlju testirat će se još jedan VPN klijent. Hide.me VPN je proizvod tvrtke eVenture Limited. Klijent je preuzet više od tri miljuna puta na platformama Windows, Mac OS, Android, iOS, Windows Phone. Koristi će se besplatna verzija dostupna za *download* na sljedećem *linku* (trenutna verzija 1.2.2):

https://hide.me/en/software

U sljedećem tekstu ukratko će se proći kroz značajke ovog programskog alata.

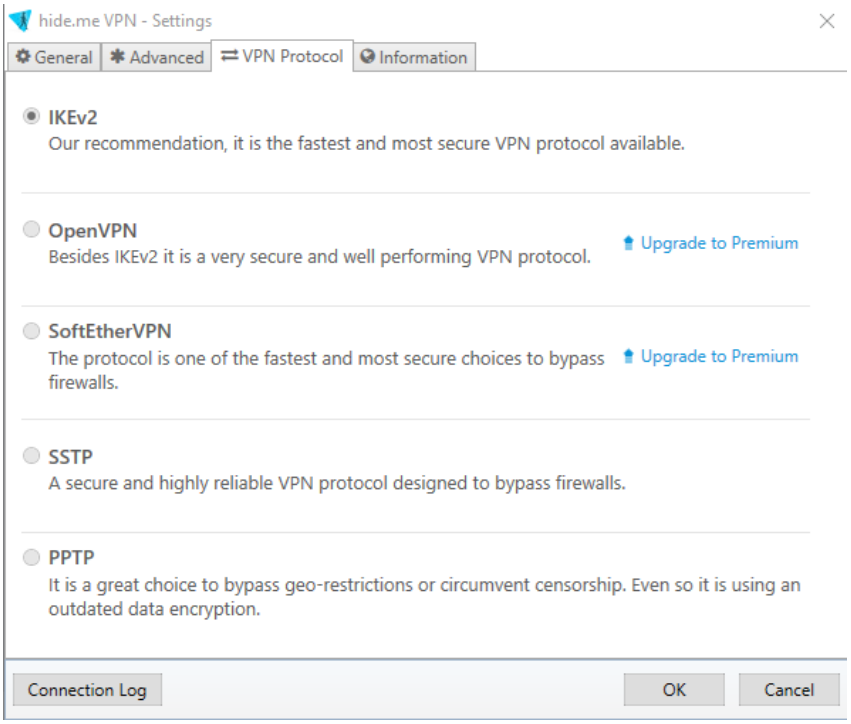
Slika 46: Sučelje Hide.me klijenta

Izvor: Autor

Sa slike 46 vidljivo je sučelje programskog alata. Povezivanje na VPN *server* jednostavan je postupak. Ovaj put odabrana je lokacija u Nizozemskoj. Vidljiva je nova IP adresa računala.

Moguće je povezati se na više od 100 *servera* u 24 različite države.

Hide.me VPN klijent ima slične napredne značajke kao i prethodno testirani VPN klijenti. Tako postoje opcije automatske veze na početku rada računala, automatski izbor najbržeg *servera*, onemogućavanje curenja IP adrese, *killswitch* za blokiranje mrežnog prometa u slučaju pada zaštite i slično.



Slika 47: Izbor protokola

Izvor: Autor

Hide.me VPN ima mogućnost izbora protokola (vidljivo sa slike 47). U ponudi su IKEv2, OpenVPN, SoftEtherVPN, SSTP i PPTP. Kao *default* protokol namješten je IKEv2. Još je moguće odabrati SSTP i PPTP dok je za korištenje OpenVPN i SoftEtherVPN potrebno nadograditi program na komercijalnu verziju.

6. Usporedna analiza načina udaljenog pristupa računalnim mrežama

Nakon pregleda i testiranja praktičnih programskih rješenja za svaki od načina udaljenog pristupa, u ovom će se poglavlju usporediti i analizirati značajke VNC/RDP, SSH i VPN. S tablice 1 vidljivi su parametri koje je autor rada uspostavio u svrhu usporedne analize.

Tablica 1: Usporedba načina udaljenog pristupa u računalnim mrežama

-	Sučelje	Arhitektura	Enkripcija	Tuneliranje	Upravljanje udaljenim računalom	Protokoli
VNC/RDP	GUI	K-S	DA	NE	DA	RFB, RDP
SSH	GUI, konzola	K-S	DA	DA	DA	SSH1, 2
VPN	GUI	K-S	DA	DA	NE	PPTP, L2F, L2TP i dr.

Izvor: Autor

Prvi je parametar „Sučelje“. Kod VNC/RDP programskih alata primjetno je da se radi o GUI (engl. *Graphical User Interface*) grafičkom sučelju. S primjera testiranja RealVNC *Eneterprisea*, Microsoft Remote Desktopa i TeamViewera vidljivo je da se radi upravo o tome. Osnovni princip funkcioniranja je unos identifikatora i lozinke udaljenog računala kako bi se grafičko sučelje transformiralo u sesijski prozor.

Kod SSH programa poput PuTTya, Tera Terma i TTY emulatora osim GUI sučelja postoji i konzolno sučelje. GUI sučelje služi, slično kao i kod VNC-a, za unos identifikatora SSH *servera* koji je pokrenut na udaljenom računalu. Osim toga postoje dodatne napredne opcije poput tuneliranja, odabira protokola, generiranja ključeva za autentifikaciju, itd. Prilikom povezivanja na IP adresu SSH *servera*, automatski se lansira drugo, konzolno sučelje. U konzolu se prvo unose autentifikacijski podaci. Potom, kada je gotov proces prijave sa određenim identitetom te se dobiju pripadajuća ovlaštenja, konzola postaje platforma za unos komandi. Konzolno sučelje unikatna je značajka SSH klijenata u odnosu na ostala programska rješenja za udaljeni pristup.

Kod VPN-a također je prisutno GUI sučelje. Izvedeno je tako da se maksimalno olakša povezivanje na željeni *server*. Često je potrebno samo kliknuti odgovarajući gumb.

Sljedeći parametar za usporedbu je arhitektura programa. Načini udaljenog pristupa koriste popularnu klijentsko-serversku arhitekturu (K-S). Npr. VNC Viewer tj. klijent instaliran je na lokalnom računalu dok je VNC Connect tj. *server* instaliran na udaljenom računalu. RDC je specifičan jer su klijentska i *serverska* strana integrirane u Microsoftov OS. Korisnik može pokrenuti samo klijentski program. *Serverska* strana ne pokreće nikakve programe, ali mora modificirati svoj vatrozid da bi se ostvarila sesija. S druge strane, TeamViewer je izveden na način da obje strane pokreću istu

aplikaciju te će klijentska strana biti ona koja unese identifikacijske podatke druge strane. Sam TeamViewer programski alat ima implementiran jednostavan tranzicijski postupak u slučaju da klijent i *server* žele zamijeniti strane tijekom sesije.

Kod VPN-a korisnik preuzima klijentski program neke od mnoštva mogućih kompanija koje pružaju VPN uslugu. Potom korisnik ima mogućnost u bilo kojem trenutku se spojiti na birani *server* te kompanije.

Sada na red dolazi vrlo bitan sigurnosni aspekt u vidu dva sljedeća parametra: enkripcije i tuneliranja. Prvi protokoli za udaljenu komunikaciju u računalnim mrežama poput SNMP-a i telneta poznati su kao vrlo nesigurni protokoli. Štoviše, cijeli je sigurnosni aspekt tu potpuno zanemaren, poruke se preko mreže šalju u obliku otvorenog teksta. Osobe kojima te poruke nisu namijenjene, a imaju malo znanja o funkcioniranju računalnih mreža, mogu jednostavno prisluškivati komunikaciju. Zbog toga su uloženi napori u stvaranje sigurnih protokola koji koriste enkripcijske postupke za šifriranje sadržaja poruke [44].

Današnji načini udaljenog pristupa u svojim trenutno dostupnim verzijama, smatraju se dosta sigurnima. Sve su veze od klijenta do *servera* enkriptirane, poruke mogu pročitati samo osobe kojima su iste namijenjene. Koriste se javni i privatni ključevi, *hash*, digitalno potpisivanje i druge metode kako bi se osigurala tajnost i vjerodostojnost komunikacije.

Tuneliranje je značajka koju posjeduju SSH i VPN. Međutim, postoje bitne razlike. VPN klijenti omogućuju spajanje na privatnu mrežu preko javne mreže kao što je Internet. Veza između klijenta i *servera* je kriptirana. Računala kojima se pristupa pri korištenju VPN-a, percipiraju VPN *server* kao izvor mrežnog prometa. Na taj način mogu se izbjeći georestrikcije[34] i zaštititi privatnost i osjetljive komunikacije na Internetu.

Kod SSH tuneliranja, SSH klijent postavlja se da se ponaša kao SOCKS *proxy*. Tada je moguće konfigurirati aplikacije na klijentskom računalu poput *web browsera* da koriste SOCKS *proxy*. Mrežni promet koji ulazi u SOCKS *proxy* prosljeđuje se SSH kriptiranom konekcijom te se to onda naziva *tunneling.* Princip je sličan pretraživanju *weba* preko VPN-a, *web serveru* se tada čini da promet dolazi od SSH *servera*. Implementacija VPN i SSH tunela uspješno je testirana u prethodnom poglavlju. Razlika je u tome što SSH tuneliranje ne pruža sve benefite VPN tuneliranja. VPN osigurava da promet svih aplikacija ide preko VPN *servera* dok je kod SSH potrebno konfigurirati svaku od aplikacija posebno.

Sljedeći parametar je „Upravljanje računalom“. SSH se koristi za konfiguriranje i administraciju SSH *servera* preko konzolnog sučelja u kojeg se upisuju odgovarajuće komande. Dok SSH programski alati imaju i neke druge svrhe poput tuneliranja, VNC je apsolutno orijentiran na upravljanje udaljenim računalom. VNC se često koristi za

[34] Oblik tehnološke mjere zaštite gdje je pristup internetskim sadržajima ograničen na temelju zemljopisnog položaja korisnika.

pružanje tehničke podrške. U prozoru VNC sesije vidljiv je *display* računala kojeg se kontrolira. Kada se inicira određena akcija na njemu, biti će vidljive i promjene u stvarnom vremenu. SSH nije toliko vizualan. U konzolu se upisuju komande i kao povratnu informaciju dobiva se određeni ispis sukladno komandi.

Protokoli koji se koriste kod *remote desktop* načina udaljenog pristupa su RFB, RDP i drugi *proprietary* protokoli. SSH koristi SSH verziju 1 i 2. VPN protokoli su najbrojniji, neki od njih su PPTP, L2F, L2TP, OpenVPN, itd.

6.1 Usporedna analiza *remote desktop* programskih alata

S tablice 2 vidljiva je usporedba *remote desktop* programskih alata koji su testirani u prethodnom poglavlju. Prvi parametar predstavlja operativne sustave koji su podržani. RealVNC podržava Windows, Mac OS, Linux platforme dok za Javu, Android i iOS pruža samo klijentski dio programa (ne i *server*). Remote Desktop Connection integriran je u Windows platformu te je to jedina platforma koju podržava. TeamViewer je najšire zastupljen po platformama, podržava Windows, Mac OS, Linux, Android te samo klijentski dio za iOS, Chrome, Javu i BlackBerry.

Tablica 2: Usporedba *remote desktop* programskih alata

-	Operativni sustav	Protokol	Ugrađena enkripcija	*File Transfer*	Audio podrška	Video podrška	*Chat*	NAT *pass.*
RealVNC *Enterprise*	Windows, Mac OS, Linux, Java, Android, iOS	VNC	AES-256	DA	NE	NE	DA	DA
Remote Desktop Connection	Windows	RDP	RC4-128	DA	NE	NE	NE	NE
TeamViewer	Windows, Mac OS, Linux, Java, Android, iOS, Chrome, BlackBerry	*Proprietary*	AES-256	DA	DA	DA	DA	DA

Izvor: [45]

Protokol koji koristi RealVNC je VNC protokol, Remote Desktop Connection koristi Microsoftov RDP protokol dok TeamViewer koristi svoj vlastiti (engl. *proprietary*) protokol koji je kombinacija VNC i RDP protokola.

Sva tri rješenja sadrže ugrađenu enkripciju. Kod RealVNC *Enterprisea* i TeamViewera to je jaka AES-256 enkripcija dok RDC koristi RC4-128.

Mogućnost prijenosa podataka dostupna je kod sva tri rješenja. RDC nema nikakvih interakcijskih mogućnosti sa *serverskom* stranom dok RealVNC ima ugrađen *chat* alat. TeamViewer je najbogatiji komunikacijskim mogućnostima, sadrži audio, video i *chat* alate.

Posljednji promatrani parametar je NAT *passthrough.* Odnosi se na mogućnost uspostavljanja sesije bez potrebe za ručnom konfiguracijom *routera*. RealVNC i TeamViewer imaju tu mogućnost dok je za RDC nužno konfigurirati *router*.

6.2 Usporedna analiza SSH programskih alata

S tablice 3 vidljiva je usporedba SSH klijenata koji su testirani u prethodnom poglavlju. Prvi promatrani parametar je operativni sustav. PuTTy klijent zastupljen je na najvećem broju platformi (Windows, djelomična potpora za Mac OS, BSD (engl. *Berkeley Software Distribution*), Linux, Solaris) dok su ostala dva klijenta dostupna samo na Windows platformi.

Tablica 3: Usporedba SSH programskih alata

-	Operativni sustav	Sučelje	SSH1, SSH2	telnet	rlogin	*Port forwarding*	SFTP/SCP	Autent. javnim ključem	*Proxy*
PuTTy	Windows, Mac OS, BSD, Linux, Solaris	Komandna linija i GUI	DA	DA	DA	DA	DA	DA	SOCKS 4,5; HTTP, Telnet, Local
Tera Term	Windows	Komandna linija i GUI	DA	DA	NE	NE	SCP	DA	SOCKS 4,5; HTTP; Telnet
TTY emulator	Windows	Komandna linija i GUI	DA	DA	DA	DA	NE	DA	SOCKS, 4, 4a, 5; HTTP, Local

Izvor: [46]

Sučelje se kod sva tri rješenja sastoji od GUI i komandne linije kako je prikazano i objašnjeno u prethodnim poglavljima.

Svi klijenti podržavavaju SSH1 i SSH2 verzije protokola. SSH1 bila je prva iteracija te je imala određena ograničenja kao npr. nemogućnost *forwardiranja portova*. Nakon toga dolazi druga iteracija u vidu SSH verzije 2 koja ima prednosti u odnosu na prvu iteraciju u vidu: poboljšane sigurnosti, veće fleksibilnosti SFTP-a

(engl. *Secure File Transfer Protocola*), interoperabilnosti sa više različitih algoritama javnog ključa poput Diffie Hellman i nove arhitekture koja zahtjeva manje korištenje programerskih vještina [47.].

Osim SSH veze, ova tri programska rješenja podržavaju i udaljeni pristup računalu korištenjem starih i nesigurni protokola. Prvi je telnet kojeg podržavaju svi klijenti dok rlogin funkcionalnost omogućavaju PuTTy i TTY emulator.

Mogućnost dinamičkog *forwardiranja portova* odnosno SSH tuneliranja, koje je praktično prikazano u prošlom poglavlju, imaju PuTTy i TTY emulator dok Tera Term nema tu mogućnost.

Obje SSH ekstenzije, SFTP i SCP, za sigurni udaljeni prijenos podataka podržava samo PuTTy, Tera Term podržava samo SCP dok TTY emulator nema takvih mogućnosti.

Sljedeći parametar je *proxy* klijent, drugačija funkcionalnost od *forwardiranja portova*, a definira se kao mogućnost SSH klijenta da se spoji na *proxy*. Protokoli koji se koriste u tu svrhu su SOCKS, HTTP itd.

Posljednji parametar je mogućnost autentifikacije javnim ključem. U prethodnom poglavlju objašnjen je proces uspostavljanja takvog vida autentifikacije pomoću generatora ključeva. Svi testirani klijenti imaju tu mogućnost.

6.3 Usporedna analiza VPN programskih alata

S tablice 4 vidljiva je usporedba testiranih VPN klijenata. Sva tri klijenta zastupljena su na najvećim platformama: Windows, Mac OS, Android i iOS. S protokolarne strane najveći izbor pruža Hide.me VPN s čak pet dostupnih protokola u komercijalnoj verziji. Vypr VPN na izbor daje četiri protokola od kojih je Chameleon razvijen od strane njihovog tima modificiranjem OpenVPN protokola. Kod TunnelBear VPN klijenta nema mogućnosti izbora protokola već se koristi samo OpenVPN.

Tablica 4: Usporedba VPN programskih alata

-	OS	Protokol	IP adrese	Države	*Log*	Brzina
Vypr VPN	Windows, Mac OS, Android, iOS	Chameleon, OpenVPN, L2TP/IPsec, PPTP	200000	51	NE	Najbrži
TunnelBear	Windows, Mac OS , Android, iOS	OpenVPN	50000	20	NE	Brz
Hide.me	Windows, Mac OS, Android, iOS	IKEv2, OpenVPN, SoftEtherVPN, SSTP, PPTP	-	24	NE	Brz

Izvor: [48]

Prema podacima o IP adresama, Vypr VPN može dodjeliti 200000 različitih IP adresa, TunnelBear 50000 dok za Hide.me nema dostupnih podataka o tom parametru. Može se spekulirati o većem broju nego kod TunnelBeara zbog nešto većeg broja raspoloživih *servera*.

Vypr VPN *serveri* nalaze se u 54 različite države svijeta, Hide.me u 24 države, a TunnelBear u 20 država. Svi testirani VPN proizvodi vole se pohvaliti činjenicom da se registar (engl. *log*) informacija o uspostavljenim vezama korisnika ne čuva u arhivi iz razloga moguće kompromitacije privatnosti.

Brzina je također važan faktor pri izboru VPN klijenta. Poznato je da se enkapsuliranjem podataka i preusmjeravanjem prometa preko *servera* gubi na brzini Internet veze. Za ova tri VPN rješenja ipak se može reći da omogućuju nesmetan *streaming* video sadržaja te je to dobar pokazatelj u smislu brzine. Vypr VPN može se ocjeniti najbržim iz autorovog iskustva korištenja tijekom testiranja.

7. Zaključak

Za potrebe izvođenja funkcije udaljenog pristupa dostupna su brojna besplatna i komercijalna programska rješenja. Procesom testiranja utvrđena je praktičnost i efektivnost promatranih VNC, VPN i SSH programskih alata.

Programi su sastavljeni od klijentske i poslužiteljske komponente. Interakcija se odvija preko sučelja. Korištenje je maksimalno pojednostavljeno kod VNC-a i VPN-a dok SSH zahtjeva nešto veće znanje poglavito zbog konfiguracije SSH *servera* te uporabe komandi za izvođenje akcija tijekom sesije.

Svi programi funkcioniraju na aplikacijskom sloju OSI modela. Implementirane su sigurnosne mjere u vidu kriptografskih metoda za šifriranje komunikacije između klijenta i poslužitelja. Očite su sličnosti u nekim aspektima ovih načina udaljenog pristupa, ali postoje i bitne razlike koje se nastojalo istaknuti u predposljednjem poglavlju.

VNC/RDP i SSH omogućuju udaljeno upravljanje i konfiguraciju računala. VPN-om se ostvaruje sigurni mrežni promet informacija preko javne računalne mreže. Za istu svrhu može poslužiti i SSH prosljeđivanjem prometa preko *proxya*.

Potom se prešlo na odvojeno testiranje klijenata prema načinu udaljenog pristupa kojem pripadaju. *Remote desktop* alati uspoređeni su prema operativnom sustavu, protokolu, enkripciji, NAT *passthrough* i komunikacijsko-interakcijskim mogućnostima. Prema navedenim parametrima najbolje rješenje je TeamViewer. Dostupan je na najvećem broju platformi: Windows, Mac OS, Linux, Java, Android, iOS, Chrome, BlackBerry. Koristi snažnu AES-256 enkripciju, podržava audio, video i *chat* mogućnosti čime je znatno olakšan pronalazak potencijalnih problema kroz komunikaciju sa drugom stranom (ako se koristi za tehničku podršku). Uz to nije potrebno vršiti konfiguraciju vatrozida.

SSH klijenti uspoređeni su prema operativnom sustavu, protokolima, mogućnosti tuneliranja, SSH ekstenzijama, autentifikaciji javnim ključem, mogućnosti povezivanja na *proxy*. PuTTy klijent jedini ispunjava sve sljedeće kriterije: dostupan je na najviše platformi, omogućava SFTP/SCP, omogućava *forwardiranje portova*, podržava metodu autentifikacije javnim ključem, a uz sve to se i dalje mogu koristiti telnet i rlogin.

Za kraj, uspoređeni su VPN klijenti prema operativnom sustavu, protokolima, broju različitih IP adresa koje se mogu dodjeliti korisnicima, broju država svijeta u kojima se nalaze *serveri*, brzini surfanja prilikom korištenja VPN-a te prema brizi o privatnosti korisnika u vidu brisanja registara uspostavljenih veza. Analizom je utvrđeno da je od testiranih klijenata najbolji izbor Vypr VPN zbog najšire dostupnosti po platformama, najvećem broju *servera,* IP adresa i protokola za odabir te zbog najveće brzine surfanja Internetom prilikom VPN sesije.

Literatura

[1]Internetski izvor: https://sysportal.carnet.hr/node/342 (10.5.2017.)

[2]Internetski izvor:

http://study.com/academy/lesson/types-of-networks-lan-wan-wlan-man-san-pan-epn-vpn.html (10.5.2017.)

[3]Internetski izvor:

http://dbrzovic.blogspot.hr/2012/05/racunalne-mreze-lan-wan-pan-man.html (10.5.2017.)

[4]Cisco, Internetski izvor:

http://cisco2960.over-blog.com/2014/02/internet-intranet-and-extranet.html (10.5.2017.)

[5]Internetski izvor: http://techdifferences.com/difference-between-client-server-and-peer-to-peer-network.html (10.5.2017.)

[6]Internetski izvor:

http://www.webopedia.com/quick_ref/OSI_Layers.asp (10.5.2017.)

[7]Kavran, Z.; Grgurević, I.: Autorizirana predavanja iz kolega Računalne mreže, Fakultet prometnih znanosti, Sveučilište u Zagrebu, Zagreb, travanj 2016.

[8]Bonaventure, O.: Computer Networking: Principles, Protocols and Practice, 2011.

[9]Rouse M., Haughn M., Gibilisco S.: Confidentiality, integrity, and availability (CIA triad), Techtarget, 2014.

[10]Internetski izvor:

https://www.vircom.com/blog/human-factors-in-cyber-security-preventing-errors/ (20.6.2017.)

[11]Internetski izvor:

http://161.53.18.5/static/erg/2005/rebac/simetrCrypto.html (20.6.2017.)

[12]Internetski izvor:

http://161.53.18.5/static/erg/2005/rebac/asimetrCrypto.html (20.6.2017.)

[13]Rouse M.: Asymmetric cryptography (public key cryptography), Techtarget, 2016.

[14]IBM, Internetski izvor:

https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_7.1.0/com.ibm.mq.doc/sy10510_.htm (20.6.2017.)

[15]Tanenbaum, A.S., Wetherall, D.J.: Computer Networks (5th Edition), Pearson Education, Inc., USA, 2011.

[16]SSL, Internetski izvor:

https://www.ssl247.com/kb/ssl-certificates/generalinformation/what-is-sha1-sha2 (1.7.2017.)

[17]SHA1 vs SHA256, Internetski izvor:

https://www.keycdn.com/support/sha1-vs-sha256/ (1.7.2017.)

[18]SHA kakulator, Internetski izvor:

http://www.xorbin.com/tools/sha256-hash-calculator (1.7.2017.)

[19]Entrust, Internetski izvor:

https://www.entrust.com/digital-signatures/ (1.7.2017.)

[20]Docusign, Internetski izvor:

https://www.docusign.com/how-it-works/electronic-signature/digital-signature/digital-signature-faq (1.7.2017.)

[21]Comodo, Internetski izvor:

https://www.comodo.com/resources/small-business/digital-certificates3.php (1.7.2017.)

[22]Živković, G.: IPsec (IP security), Seminarski rad, Fakultet elektrotehnike i računarstva, 2006.

[23]Wikipedia, Internetski izvor: https://en.wikipedia.org/wiki/IPsec (1.7.2017.)

[24]CARNet CERT, LS&S: IPsec, 2004.

[25]Internetski izvor: http://www.webopedia.com/TERM/F/firewall.html (1.7.2017.)

[26]Kurose, J.F., Ross, K.W. : Computer Networking: A Top-Down Approach (5th Edition), Pearson Education, Inc., USA, 2009

[27]Richardson T., Stafford-Fraser Q., Wood K., Hopper A.: Virtual Network Computing, IEE Internet Computing Vol.2, 1998.

[28]CARNet, LS&S: Virtual Network Computing, 2010.

[29]Wikipedia, Internetski izvor:

https://en.wikipedia.org/wiki/Virtual_Network_Computing (10.7.2017.)

[30]Microsoft, Internetski izvor:

https://msdn.microsoft.com/en-us/library/aa383015(v=vs.85).aspx (10.7.2017.)

[31]Rouse M.: Remote desktop protocol, Techtarget, 2017.

[32]Rouse M., Cobb M.: Secure Shell (SSH), Techtarget, 2016.

[33]Stalings W.: Protocol Basics: Secure Shell Protocol, The Internet Protocol Journal, Volume 12, No.4

[34]Wikipedia, Internetski izvor:

https://en.wikipedia.org/wiki/Secure_Shell#Architecture (20.7.2017.)

[35]Cisco, Internetski izvor:

http://www.cisco.com/c/en/us/support/docs/security-vpn/ipsec-negotiation-ike-protocols/14106-how-vpn-works.html (20.7.2017.)

[36]CARNet CERT, LS&S: Osnovni koncepti VPN tehnologije, 2003.

[37]Express VPN, Internetski izvor:

https://www.expressvpn.com/what-is-vpn/protocols/pptp (20.7.2017.)

[38]Internetski izvor:

http://telekomunikacije.ba/virtualne-privatne-mreze-vpn/ (20.7.2017.)

[39]RealVNC, Internetski izvor: https://www.realvnc.com/en/ (27.7.2017.)

[40]Youtube, Internetski izvor:

https://www.youtube.com/watch?v=gsP46ltENRY (27.7.2017.)

[41]TeamViewer, Internetski izvor: https://www.teamviewer.com/hr/ (5.8.2017.)

[42]Mueller J.P.: Windows Command Line Administration Instant Reference, Wiley, 2010.

[43]Barrett D., Silverman R., Byrnes R.: SSH, The Secure Shell: The Definitive Guide, O'Reilly Media, Inc, 2005.

[44]McNab C.: Network Security Assessment: Know Your Network, O'Reilly Media, Inc, 2016.

[45]Wikipedia, Internetski izvor:

https://en.wikipedia.org/wiki/Comparison_of_remote_desktop_software (10.8.2017.)

[46]Wikipedia, Internetski izvor:

https://en.wikipedia.org/wiki/Comparison_of_SSH_clients (10.8.2017.)

[47]Dwivedi H.: Implementing SSH: Strategies for Optimizing the Secure Shell, Wiley, 2003.

[48]Internetski izvor: http://whatismyipaddress.com/vpn-comparison (10.8.2017.)

Popis kratica i akronima

AES (engl. *Advance Encryption Standard*) napredni enkripcijski standard

AH (engl. *Authentification Header*) protokol koji je dio IPsec seta protokola, osigurava autentifikaciju i integritet *datagrama*

ASN.1 (engl. *Abstract Syntax Notation One)* jezik za opis sučelja često korišten u telekomunikacijama, računalnim mrežama i kriptografiji

AT&T (engl. *American Telephone and Telegraph Inc.*) američka telekomunikacijska kompanija

ATM (engl. *Asynchronous Transfer Mode*) tehnika prijenosa u telekomunikacijama

ATMP (engl. *Ascend Tunnel Management Protocol*) jedna od tehnologija za obavljanje postupka tuneliranja

BSD (engl. *Berkeley Software Distribution*) inačica Unix operativnog sustava

CA (engl. *Certification Authority*) entitet koji izdaje digitalne certifikate

CAN (engl. *Campus Area Network*) više međusobno spojenih lokalnih mreža na području kampusa

CD (engl. *Compact Disc)* medij za pohranu podataka

CIA (engl. *Confidentiality Integrity Availability*) model za vođenje politike informacijske sigurnosti

DDOS (engl. *Distributed Denial Of Service*) napad kojem je svrha onemogućavanje raspoloživosti preopterećivanjem računalne mreže

DES (engl. *Data Encryption Standard*) podatkovni enkripcijski standard

DNS (engl. *Domain Name System*) hijerarhijski decentralizirani sustav za imenovanje računala, usluga i drugih resursa spojenih na Internet ili privatnu mrežu

DVD (engl. *Digital Versatile Disc*) medij za pohranu podataka

ECC (engl. *Elliptic Curve Cryptography*) pristup kriptografiji javnog ključa temeljen na algebarskoj strukturi eliptičnih krivulja preko konačnih polja

EPN (engl. *Enterprise Private Network*) privatna mreža tvrtke

ESP (engl. *Encapsulated Security Payload*) dio IPsec seta protokola, pruža povjerljivost, autentifikaciju, integritet i *anti replay*

FTP (engl. *File Transfer Protocol*) protokol za prijenos datoteka

GRE (engl. *Generic Routing Encapsulation*) protokol za tuneliranje razvijen od strane Cisco Systems

GSSAPI (engl. *Generic Security Service Application Program Interface*) standard koji rješava problem više sličnih nekompatibilnih sigurnosnih servisa

GUI (engl. Graphical User Interface) grafičko sučelje programa

HAN (engl. *Home Area Network*) mala kućna računalna mreža

HTTP (engl. *HyperText Transfer Protocol*) najčešći protokol za prijenos informacija na *webu*

IBM (engl. *International Business Machines Corporation*) američka tehnološka kompanija

ICV (engl. *Integrity Check Value*) polje za provjeru autentičnosti i integriteta

IETF (engl. *Internet Engineering Task Force*) velika i otvorena međunarodna zajednica mrežnih dizajnera, operatora, proizvođača i istraživača

IKE (engl. *Initial Key Exchange*) standardni protokol za sigurno dogovaranje virtualne privatne mreže

IP (engl. *Internet Protocol*) standardni protokol kojim se *datagrami* šalju od jednog do drugog računala

IPsec (engl. *Internet Protocol Security*) set protokola za autentifikaciju i enkripciju paketa poslanih mrežom

ISO (engl. *Internation Organization for Standardization*) međunarodna organizacija za normiranje

ITU (engl. *International Telecommunication Union*) međunarodna telekomunikacijska unija

JPEG (engl. *Joint Photographic Experts Group*) komprimirani slikovni format s gubicima izveden iz bitmape

K (engl. *Key*) tajni ključ

KE (engl. *Key Encryption*) ključ za enkripciju

KD (engl. *Key Decryption*) ključ za dekripciju

L2F (engl. *Layer 2 Forwarding*) protokol za tuneliranje razvijen od strane Cisco Systems

L2TP (engl. *Layer 2 Tunneling Protocol*) protokol za tuneliranje, kombinacija L2F i PPTP protokola

LAC (engl. L2TP *Access Concentrator*) pristupni koncentrator, jedna od krajnjih točaka L2TP tunela

LAN (engl. *Local Area Network*) lokalna mreža

LNS (engl. L2TP *Network Server*) mrežni poslužitelj, jedna od krajnjih točaka L2TP tunela

MAC (engl. *Media Access Control*) donji podsloj podatkovnog sloja OSI modela

MAC (engl. *Message Authentication Code*) polje koje sadrži određenu vrijednost ako je prethodno dogovorena autentifikacija poruka. Služi za provjeru sigurnosne nepovredivosti paketa.

MAN (engl. *Metropolitan Area Network*) više međusobno povezanih mreža na području grada

MPEG (engl. *Moving Picture Experts Group*) radna grupa koja postavlja standarde za audio i video kompresiju

MS-CHAP (engl. *Microsoft Challenge Handshake Authentication Protocol*) Microsoft-ova verzija CHAP protokola za autentifikaciju

NAT (engl. *Network Address Translation*) translacija mrežnih adresa

NCP (engl. *Network Control Procotol*) protokol transportnog sloja korišten u ARPANET-u

NTLM (engl. NT *Lan Manager*) Microsoftov sigurnosni protokol za pružanje autentifikacije, integriteta i povjerljivosti korisnicima

OpenPGP (engl. *Open Pretty Good Privacy*) *open source* verzija PGP enkripcije

OS (engl. *Operating System*) operativni sustav

OSI (engl. *Open Systems Interconnection*) model za mrežnu komunikaciju donesen od ISO-a

PAN (engl. *Personal Area Network*) osobna računalna mreža na malom području

PAP (engl. *Password Authetication Protocol*) protokol koji se rabi za provjeru korisničnih računa bez korištenja šifre u komunikacijskom protokolu PPP

PB javni ključ

PICT format za grafičke datoteke predstavljen na Apple Macintosh računalu

PKI (engl. *Public Key Infrastructure*) protokol za kreiranje, upravljanje, distribuciju, korištenje, pohranu i ukidanje digitalnih certifikata

PPP (engl. *Point to Point Protocol*) protokol za komunikaciju od točke do točke

PPTP (engl. *Point to Point Tunneling Protocol*) zastarjeli protokol za implementaciju virtualnih privatnih mreža

RC4 (engl. Rivest *Cypher 4*) simetrična kriptografska protočna šifra

RDP (engl. *Remote Desktop Protocol*) Microsoftov protokol za omogućavanje udaljenog pristupa

RDC (engl. *Remote Desktop Connection*) Microsoftov klijent za udaljeni pristup

RFB (engl. *Remote Framebuffer*) protokol koji koristi VNC za omogućavanje udaljenog pristupa

RFC (engl. *Request For Comments*) formalni dokument sastavljen od IETF koji opisuje specifikacije za određenu tehnologiju

RS2321 (engl. *Recommended Serial 2321*) nekadašnji standardni ulaz na osobnom računalu

RSA (Rivest Shamir Adleman) jedan od prvih praktičnih kriptosustava sa javnim ključem

S/MIME (engl. *Secure Multipurpose Internet Mail Extension*) standard koji nudi dodatni sloj zaštite za e-poštu

SA (engl. *Security Associations*) uspostavljanje zajedničkih sigurnosnih atributa između dvaju mrežnih entiteta za podršku sigurne komunikacije

SB privatni ključ

SAN (engl. *Storage Area Network*) mreža za spajanje računala na spremišta podataka

SCP (engl. *Secure Copy*) ekstenzija SSH, omogućuje kopiranje datoteka između *hostova*

SFTP (engl. *Secure File Transfer Protocol*) mrežni protokol za sigurni transfer datoteka preko SSH

SHA (engl. *Secure Hash Algorithm*) obitelj kriptografskih *hash* funkcija

SMTP (engl. *Simple Mail Transfer Protocol*) protokol za prijenos e-pošte na Internetu

SSH (engl. *Secure Shell*) mrežni protokol koji korisnicima omogućuje uspostavu sigurnog komunikacijskog kanala između dva računala putem nesigurne računalne mreže

SSL (engl. *Secure Sockets Layer*) transportni protokol koji omogućuje sigurnu komunikaciju preko Interneta za različite aplikacije

SSTP (engl. *Secure Sockets Tunneling Protocol*) VPN tunel koji pruža mehanizme za prijenos PPP prometa preko SSL/TLS kanala

TCP (engl. *Transmission Control Protocol*) protokol transportnog sloja za spojnu pouzdanu vezu

TLS (engl. *Transport Layer Security*) kriptografski protokol koji omogućuje sigurnu komunikaciju preko Interneta

UDP (engl. *User Datagram Protocol*) protokol transportnog sloja za nespojnu nepouzdanu vezu

UTP (engl. *Unshielded Twisted Pair*) bakrena parica

VNC (engl. *Virtual Network Computing*) način udaljenog pristupa za upravljanje udaljenim računalom

VPDN (engl. *Virtual Private Dial-up Network*) mreža koja proširuje klijente *dialup* daljinskog pristupa na privatnu mrežu

VPN (engl. *Virtual Private Network*) virtualna privatna mreža

WAN (engl. *Wide Area Network*) računalna mreža na velikom geografskom području

WLAN (engl. *Wireless Local Area Network*) bežična lokalna mreža

Popis slika

Slika 1 Sigurna komunikacija preko nesigurnog kanala 10
Slika 2: Simetrična kriptografija 11
Slika 3: Asimetrična kriptografija 12
Slika 4: SHA-1 *hash* generator 14
Slika 5: SHA-2 (SHA-256) *hash* generator 14
Slika 6: Digitalno potpisivanje i slanje dokumenta 16
Slika 7: Standardni IP *datagram* 16
Slika 8: Zaglavlje AH 17
Slika 9: ESP datagram 18
Slika 10: VNC klijent, *server* i protokol 20
Slika 11: Protokolni složaj SSH 23
Slika 12: Razmjena paketa kod Transport Layer protokola 24
Slika 13: *Connect* funkcija VNC *Viewera* 30
Slika 14: Povezivanje sa udaljenim računalom u sučelju VNC Viewera 31
Slika 15: VNC sesija 31
Slika 16: Omogućavanje udaljenog pristupa 32
Slika 17: Konfiguracija *routera* 33
Slika 18: Remote Desktop sučelje 34
Slika 19: Sučelje TeamViewera 35
Slika 20: TeamViewer sesija 36
Slika 21: *View* opcija u alatnoj traci TeamViewera 36
Slika 22: *Communicate* opcija u alatnoj traci TeamViewera 37
Slika 23: Sučelje FreeSSHd *servera* 38
Slika 24: *User* kartica SSH *servera* 38
Slika 25: Dodavanje novog korisnika 39
Slika 26: PuTTy sučelje 40
Slika 27: Konzola PuTTy programa 41
Slika 28: Ispis komande „cd“ 41
Slika 29: Ispis komande „netstat“ 42
Slika 30: Dinamičko forwardiranje 43
Slika 31: Postavke spajanja u Firefoxu 44
Slika 32: Sučelje Tera Term programa 45
Slika 33: FreeSSHd *Authentication* kartica 45
Slika 34: Tera term *Setup* traka 46
Slika 35: Tera Term *KeyGenerator* 46
Slika 36: Tera Term nova konekcija 47
Slika 37: Tera Term autentifikacijski prozor 47
Slika 38: Sučelje TTY emulator 48
Slika 39: Sučelje Vypr VPN 49
Slika 40: Odabir lokacije VPN *servera* 50
Slika 41: *Connection* opcije Vypr VPN-a 50
Slika 42: Izbor protokola u Vypr VPN 51

Slika 43: *Web* stranica za provjeru IP adrese 52
Slika 44: VPN veza sa *serverom* u SAD-u 53
Slika 45: Omogućen pristup Pandora radiju 53
Slika 46: Sučelje Hide.me klijenta 54
Slika 47: Izbor protokola 55

Popis tablica

Tablica 1: Usporedba načina udaljenog pristupa u računalnim mrežama 56
Tablica 2: Usporedba *remote desktop* programskih alata 58
Tablica 3: Usporedba SSH programskih alata 59
Tablica 4: Usporedba VPN programskih alata 61