Sigurnost u web komunikaciji

Furdić, Matija

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:195:727132

Sveučilište u Rijeci – Odjel za informatiku

Poslovna informatika

Matija Furdić

Sigurnost u web komunikaciji

Diplomski rad

Mentor: prof. dr. sc. Mario Radovan

Rijeka, rujan 2018.

Sadržaj

Zadatak diplomskog rada

Sažetak i ključne riječi

Tema ovog diplomskog rada „Sigurnost u web komunikaciji“ opisana je kroz uvod, razradu i zaključak. U uvodu govorim općenito o sigurnosti u web komunikaciji. Razradu sam započeo sa problematikom te sam opisao problemska područja. Zatim slijedi kriptografija. Nakon općenitog opisa kriptografije, opisao sam šifriranje sa simetričnim ključem gdje sam naveo algoritme za šifriranje sa simetričnim ključem i šifriranje po blokovima. Nakon toga sam opisao šifriranje sa javnim ključem te RSA algoritam. Nakon toga govorim o integritetu poruka i autentičnosti komunikatora te digitalnom potpisu. U posljednjem dijelu razrade opisao sam protokole SSL/TLS, HTTPS i DTLS. Na kraju rada nalazi se zaključak rada te literatura koju sam koristio prilikom izrade rada.

Ključne riječi: Sigurnost, web, šifriranje, dešifriranje. šifrat, ključ, RSA, integritet, autentičnost, digitalni potpis, protokol, SSL/TLS, HTTPS, DTLS, rukovanje, certifikat, klijent, server.

1. Uvod

Sigurnost u web komunikaciji je iznimno opsežna tema koja sadrži definirane probleme, metode, sredstva i ciljeve. Pomoću računalne mreže odvija se komunikacija. Mrežu je potrebno štititi kao tehnološki sustav. U procesu mrežne komunikacije postoje tri osnovna elementa zaštite i sigurnosti. Prvo je provjera tajnosti sadržaja, drugo je zaštita integriteta, izvornosti sadržaja i treće je utvrđivanje autentičnosti komunikatora. Prvi element odnosi se na sadržaj koji se prenosi mrežom. On mora biti tajan i ne smije doći do nikoga osim osobe za koju je namijenjen. Zaštita integriteta sadržaja jako je bitna jer može doći do namjernog iskrivljenja u prijenosu te imati znatne posljedice na ispravnost sadržaja. Zadnji element, utvrđivanje autentičnosti komunikatora odnosi se na zaštitu od osoba koje se lažno predstavljaju.

U web komunikaciji ima jako puno različitih komunikatora. To mogu biti privatne osobe, usmjerivači, klijent i server. Privatne osobe razmjenjuju osobne poruke na mreži. Usmjerivači međusobno razmjenjuju pakete podataka o stanju puteva i veza. Temeljem tih podataka oni izrađuju tablice usmjeravanja i prosljeđivanja. Rad računalne mreže mogu poremetiti lažni podaci ili iskrivljene poruke koje ti usmjerivači izmjenjuju. Za uspostavu pouzdane i sigurne TCP veze zaduženi su klijent i server. To je paradigma u kojoj jedna aplikacija, server pasivno čeka drugu aplikaciju, klijent koja inicira komunikaciju. Informacije mogu teći u oba smjera klijent je taj koji inicira komunikaciju.

Kada govorimo o pouzdanosti mislimo na ispravnost prijenosa sadržaja na mreži. Sustav mora otkrivati te ispravljati nehotične tehničke greške tako da krajnji korisnik dobije ispravan sadržaj. Kada govorimo o namjernim ometanjima i iskrivljenima sadržaja tada mislimo na sigurnost web komunikacije. Takvo ometanje zove se napad na sustav. Radi takvih napadača potrebno je dobro zaštititi mrežnu komunikaciju.

2. Problemska područja

Sigurnost i zaštita računalne mreže ima više problemskih područja. Neke od njih su pitanje povjerljivosti, autentičnosti i integriteta.

Zaštita tajnosti odnosno privatnosti sadržaja u računalnoj mreži naziva se povjerljivost. To je osnovno pitanje zaštite u računalnoj komunikaciji. Treba se omogućiti zaštita povjerljivosti kod osobne i poslovne komunikacije, financijskih transakcija te svih ostalih kao što je sigurnost zajednica i institucija te sličnih vrsta komunikacija koji se prenose mrežom.

Sadržaji koje se prenose mrežom moraju biti razumljivi i dostupni jedino pošiljatelju te onome kome je taj sadržaj namijenjen. Napadač najčešće prisluškuje mrežnu komunikaciju te kopira sadržaj iste. Ako napadač uspije kopirati sadržaj sustav zaštite mora imati mogućnost da ga ne razumije odnosno da ga ne može pročitati. Šifriranje ili enkripcija je zapisivanje sadržaja tako da ga mogu čitati samo oni kojima je on namijenjen. Povjerljivost se realizira tako da pošiljatelj šifrira (kriptira) sadržaj prije prijenosa mrežom i samo ga primatelj može dešifrirati.

Slika 1. Lozinka

(preuzeto sa: https://www.pcworld.com/article/2026979/tweet-of-day-how-to-create-a-strong-password-to-guard-against-hackers.html)

Kod integriteta sadržaja, računalna mreža mora osigurati prijenos sadržaja u izvornom obliku odnosno ne smije dozvoliti da dođe do namjernog iskrivljenja. Sustav za zaštitu ne zaustavlja namjerna ili nenamjerna iskrivljena nego omogući krajnjem primatelju poruke otkrivanje tih iskrivljenja. Tako on izbjegava neugodne posljedice koje netočne odnosno

iskrivljene poruke mogu izazvati. Jedan primjer je problem kod financijskih transakcija. Ukoliko napadač „presretne“ transakciju te izmjeni sadržaj, recimo umjesto broja računa primatelja preusmjeri novac na svoj račun dolazi do problema te novac neće biti isplaćen onome kome je namijenjen. Bitno je da to mrežna usluga prijenosa otkrije te upozori banku da se transakcija ne izvrši. Nehotična iskrivljenja se isprave tako da se ponovo pošalju samo iskrivljeni podaci.

Svaka strana u komunikaciji mora biti sigurna da komunicira sa osobom za koju se ona predstavlja da jest. To mogućnost je utvrđivanje autentičnosti komunikatora. Ta mogućnost važna je kod osobne komunikacije preko računalne mreže ali i kod poslovnih komunikacija i financijske transakcije. Računalne mreže moraju imati metode i sredstva koja omogućuju sigurno utvrđivanje pravog identiteta komunikatora. Lažna predstavljanja onemogućila bi poslovno komuniciranje.

Kod operativne sigurnosti sustava računalna mreža mora sadržavati način na koji se štiti od različitih vrsta napada koji bi ugrozili njezin rad. Funkcioniranje mreže može se ugroziti na puno načina. Najčešće je to putem različitih virusa, crva te ostalih programa koji ometaju mreži te poruke odnosno sadržaj na njoj. Napad je uglavnom usmjeren na pojedino web sjedište. Cilj napada je uništiti, iskriviti, ukrasti sadržaj ili samo ometati normalan rad. Postoje mnogo različitih vrsta napada na web sjedište. Neke od njih odvijaju se na način da napadač instalira razne nametnike preko drugih računala koji istodobno šalju zahtjev za uspostavu veze istom sjedištu. Tako web server pokušava odgovoriti na velik broj zahtjeva, više nego što može, te se time blokira njegov rad. Ukoliko ne dođe do blokiranja servera, onemogućava se uspostava veze sa stvarnim klijentima i izvršenje njihovih zahtjeva. Takav napad zove se odbijanje pružanja usluge ili uskraćivanje usluge (Denial of Servise - DoS) jer server odbija pružiti usluge korisniku. DoS je također napad preplavljivanjem frekventnog pojasa zato što velik broj lažnih zahtjeva sa raznih računala dovodi server i kapacitete veze do iscrpljenosti te klijenti ne mogu uspostaviti veze sa njim. DoS napadi se najčešće izvode putem tzv. boneta. To su računalne mreže koje su zaražene crvom ili trojanskim konjem. Na taj način jako je teško otkriti počinitelja. Moguće je da se on nalazi i u stranoj zemlji. Veliki problem kod DoS napada je što je ga je lakše pokrenuti nego se obraniti od njega. Napad uglavnom izgleda na način da netko kada pretražuje određene IP adrese koje su zaražene trojanskim konjem ili crvom. Zatim uz pomoć nekih alata koji su gotovi (npr. Stacheldraht) napadne određeno web sjedište. U Hrvatskoj napadi na ovaj način nisu rijetkost te mogu prouzročiti veliku štetu. Da bi se mogao saznati napadač,

stručnjaci surađuju sa ponuđačima internetskih usluga sa kojih je IP adresa napad prouzročen te napraviti detaljniju analizu datoteka na web sjedištu.

Da bi se olakšala obrana potebni su sposobni administratori te hardver i softver koji omogućuju obranu. Ako je napad dobro konstruiran teže se možemo obraniti. Brojni napadi uglavnom su jednostavni no grupa sposobnih ljudi, hakera, može prouzročiti velike probleme. Imamo dosta primjera napada na Facebook, Twitter, Blogger itd.

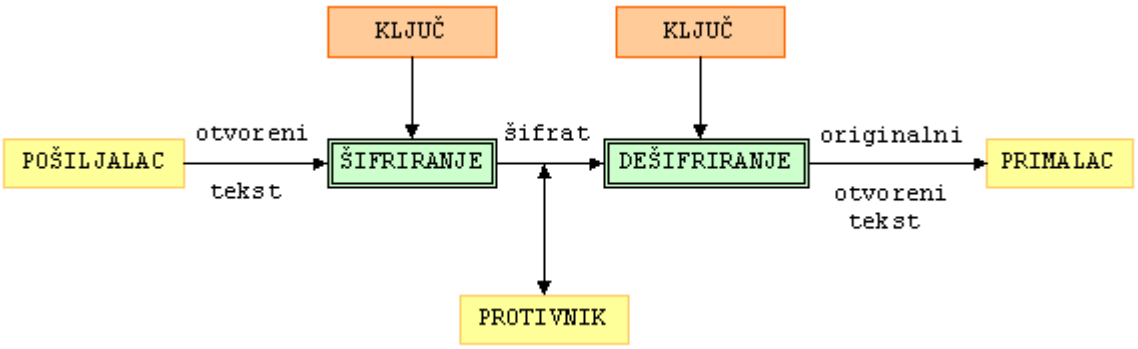
Napad prisluškivanjem uglavnom se događa na usmjerivačima kroz koji sadržaj prolazi. Napadač taj sadržaj kopira, preuzima ili skreće s puta, mijenja sadržaj te prosljeđuje izmijenjene poruke na adresu primatelja. Također napadač može i obrisati poruke koje presretne na putu do krajnjeg primatelja. Napadač može napraviti štetu lažno predstavljajući.

3. Kriptografija

Kriptografija je znanstvena disciplina koja proučava metode za slanje poruka tako da ih može pročitati samo onaj kome su namijenjene. Takav način zapisivanja zove se šifriranje sadržaja. To je riječ grčkog podrijetla a znači tajnopis.

Kriptografija se pojavljuje još u petom stoljeću prije Krista kod starih Grka. U posljednje vrijeme kriptografija je jako napredovala kako bi sigurnost web komunikacije bila što je moguće veća.

Osnovna zadaća kriptografije da se omogući pošiljatelju i primaocu komunikacija tako da treća osoba ne može nadzirati komunikacijski kanal niti razumjeti sadržaj poruke. Poruku koju prenosi nazvat ću otvoreni tekst. Pošiljatelj postupkom šifriranja izradi unaprijed dogovoreni ključ. Rezultat dobiven šifriranjem naziva se šifrat ili kriptogram. Zatim pošiljatelj šalje šifrat preko nekog komunikacijskog kanala. Treća osoba može doznati sadržaj šifrata ali ne može odrediti o kojem se sadržaju radi. S druge strane, primaoc zna o kojem se sadržaju radi te tako može dešifrirati otvoreni tekst.



Slika 2. Šifriranje

(Preuzeto sa: https://web.math.pmf.unizg.hr/~duje/kript/osnovni.html)

Šifriranje se izvodi uz pomoć određenog algoritma za šifriranje. Algoritmi mogu biti hardverski ili softverski realizirani. Struktura i veličina šifriranja definirana je zajedno sa algoritmom. Ključ se sastoji od niza bitova zadane dužine te se tako određuje kako će algoritam šifrirati sadržaj. Algoritam za šifriranje je obično javan dok ključ za šifriranje i dešifriranje mogu

biti javni i tajni. Šifriranje i dešifriranje mogu se izvesti sa istim ali i različitim ključevima. Jedan od ključeva svakako mora biti tajan jer bi onda svatko mogao šifrirati odnosno dešifrirati poruku.

Na slici je prikazano kako pošiljatelj daje algoritmu za šifriranje otvoren tekst te svoj ključ za šifriranje. Algoritam to prihvaća te na temelju toga stvara šifrat. Primatelj daje šifrat algoritmu za dešifriranje te na temelju toga algoritam proizvodi otvoreni tekst.

Postoji mnogo algoritama za šifriranje. Na slici 2. je prikazan postupak koji sadrži otvoreni tekst, ključeve od pošiljatelja i primatelja, ali nema naziva algoritma kako se izvodi šifriranje i dešifriranje.

Kriptosustav je uređena petorka (p, ℓ, Ќ, Ɛ, Ɗ) za koju vrijedi:

1. p je konačan skup svih mogućih osnovnih elemenata otvorenog teksta
2. ℓ je konačan skup svih osnovnih elemenata šifrata
3. Ќ je konačan skup svih mogućih ključeva
4. Za svaki K ∈ Ќ postoji funkcija šifriranja $e_K \in Ɛ$ i odgovarajuća funkcija dešifriranja $d_K \in Ɗ$. Pritom su e_K: p → ℓ i d_K: ℓ → p funkcije sa svojstvom da je $d_K(e_K(x)) = x$ za svaki otvoreni tekst x ∈ p.

Najbitnije je svojstvo u definiciji je $d_K(e_K(x)) = x$. Nakon toga funkcije e_K su injekcije.

Kriptosustave možemo klasificirati na tri kriterija. Tip operacije koje se koriste kod šifriranja, način kako se obradi otvoreni tekst te tajnost i javnost ključeva.

1. Tip operacije koje se koriste kod šifriranja

Postoji podjela na supstitucijske šifre. U njima svaki se element otvorenog teksta mijenja sa nekim drugim elementom te transpozicijske šifre gdje se elementi teksta permutiraju. Otvoreni tekst može biti bit, slovo, grupa bitova ili slova. Npr. ako riječ BAJKA šifriramo u YIWOX, napravili smo supstituciju. No ukoliko je šifriramo u JABAK, napravili smo transpoziciju. Napredniji sustavi vrše kombinaciju ove dvije metode.

2. Način kako se obradi otvoreni tekst

Kod ovog načina možemo razlikovati blokove šifra. Ta metoda vrši obradu jednog po jednog bloka elementa otvorenog teksta tako da koristi isti ključ K i protočne šifre. Kod protočne šifre

elementi otvorenog teksta obrade se jedan po jedan na način da se koristi niz ključeva koji se pritom generira paralelno.

3. Tajnost i javnost ključeva

Kod tajnog i javnog ključa postoje dvije podjele. To su simetrični kriptosustavi i kriptosustavi sa javnim ključem.

3.1. Šifriranje sa simetričnim ključem

Kod postupka šifriranja zamijeni se izvorni zapis sadržaja sa drugim zapisom tog sadržaja. Primatelj zapisa mora znati koji se algoritam koristi tj, na koji način je zapis šifriran te sa kojim ključem ga treba dešifrirati. Na taj način on dolazi do izvornog zapisa koji je čitljiv i razumljiv.

Neki algoritam šifriranja vrši zamjenu slova iz izvornog zapisa sa slovima abecede koja se nalaze onoliko niže u abecedi koliko je to zadano sa vrijednošću ključa. U hrvatskoj abecedi slova „lj“, „nj“, „dž“ pišu se kao dva znaka te ih algoritam ne prepoznaje kao jedan znak. Zato ovaj primjer uzimam sa standardnom abecedom koja ima 26 slova. Kod algoritama sa pomakom slova uzima se da nakon zadnjeg slova „z“, slijedi opet prvo slovo „a“.

Tekst poruke koju treba šifrirati neka bude „Sutra je praznik“. Vrijednost ključa je 4. Algoritam izvodi proces šifriranja te za ovaj tekst proizvodi šifrirani zapis : „Jezto ma ktolsuy“. Zapis izgleda drugačije od izvornog sadržaja, no ako se napadač uspije dokopati šifriranog zapisa, vrlo lako može otkriti stvarni sadržaj. Algoritam radi na način da svako slovo abecede S_i svaki put zamjenjuje isto slovo S_j te tako olakšava čitanje izvornog sadržaja.

Istom metodom mogu se napraviti i puno bolji i složeniji algoritmi za šifriranje. Još jedan primjer šifriranja sa abecednim pomakom, izvodi se prema ključu koji sadrži

četiri brojke. Prve dvije brojke pokazuju veličinu pomaka za neparna slova dok druge dvije pokazuju veličinu pomaka za parna slova. Neka ključ bude 1103, tada je šifrirani zapis izvornog teksta „Sutra je praznik“ „Hkyku lo jpuznxi“.

Sadržaj koji je šifriran na taj način puno je teže otkriti nego u prošlom primjeru. U ovom primjeru različita slova mogu se preslikati u isto slovo. Recimo u riječi „Sutra“ drugo slovo „u“ i četvrto slovo „r“ preslikavaju se u slovo „k“. Ukoliko napadač uspije doći do šifriranog zapisa, vrlo teško će otkriti izvorni tekst radi toga što se različita slova preslikavaju u isto slovo. Upravo to je svrha šifriranja, napadač ne može otkriti izvorni tekst već će tekst pročitati samo onaj kome je namijenjen. Primatelj prikazuje izvorni zapis tako da na šifrirani zapis učini obrnuti proces, po istom ključu. Taj se proces svodi na pomak po abecedi samo što se to radi obrnutim smjerom nego kod šifriranja.

Morfoalfabetske metode šifriranja su one metode kod kojih se jedno slovo zamjenjuje samo sa jednim slovom, tj, S_i se zamjenjuje sa S_j. Poliaflabetske metode šifriranja su one metode kod kojih se isto slovo može zamijeniti sa različitim slovima. Isto slovo S_i mijenja se sa slovom S_j samo ukoliko se nalaze na neparnom mjestu u izvornom slovu a ako se nalaze na parnom mjestu u izvornom slovo onda se zamjenjuju sa S_k. Kod polialfabetske metode postoji jako mnogo mogućnosti te je zato dosta teže napadaču otkriti o kojoj se šifri radi. Naravno, u praksi ti primjeri su mnogo složeniji nego u ovim jednostavnim primjerima. Oni koriste „zamješavanje“ izvornog teksta na razini bitova dok su ovi primjeri razini slova.

Još jedan primjer je tablica preslikavanja po kojoj se zamjena slova definira eksplicitno za pojedino slovo. To ne radi uz pomoć skupa parova. Prvi znak iz para preslika se jednoznačno u drugi znak. Primjer tablice preslikavanja može se vidjeti na slici.

Izvorni zapis	a	b	c	d	e	f	g	.....
Šifrirani zapis	;	3	f	t	c	9	a	.....

Tablica 1. Tablica preslikavanja

U tablici se mogu šifrirati samo slova no one mogu također sadržavati brojke, interpunkcije i druge znakove. Takve tablice imaju ulogu ključa. Ovaj proces je vrlo jednostavan, šifrirani tekst izrađuje se na način da za svako slovo pogledamo gornji redak, odnosno izvorni zapis. Na ovaj način riječ „abeceda“ u šifriranom zapisu izgledala bi ovako: „:3cfct:“.

Prednost ove metode je u tome što ima jako puno mogućnosti. Za X znakova postoji X! (X faktorijela) mogućih preslikavanja. To je broj veličine 10^x. Što je X (broj znakova) veći to je broj preslikavanja bitno veći te je time i sustav šifriranja sigurniji. Pošto ima jako velik broj mogućnosti tablica preslikavanja znakova, jako teško je dešifrirati zapis primjenjujući grubo računanje. Ako napadač uspije dobiti kopiju šifriranog zapisa ali nema tablicu po kojoj se izvršavalo šifriranje, on nema nikakve šanse da otkrije izvorni zapis metodom probavanja s velikim brojem tablica. Pošto ima jako mnogo tablica, vrlo je mala vjerojatnost da se, bez obzira na velike brzine današnjih računala, otkrije točna tablica. Trebalo bi puno godina da se metodom pokušavanja sa različitim tablicama dešifrira izvorni sadržaj.

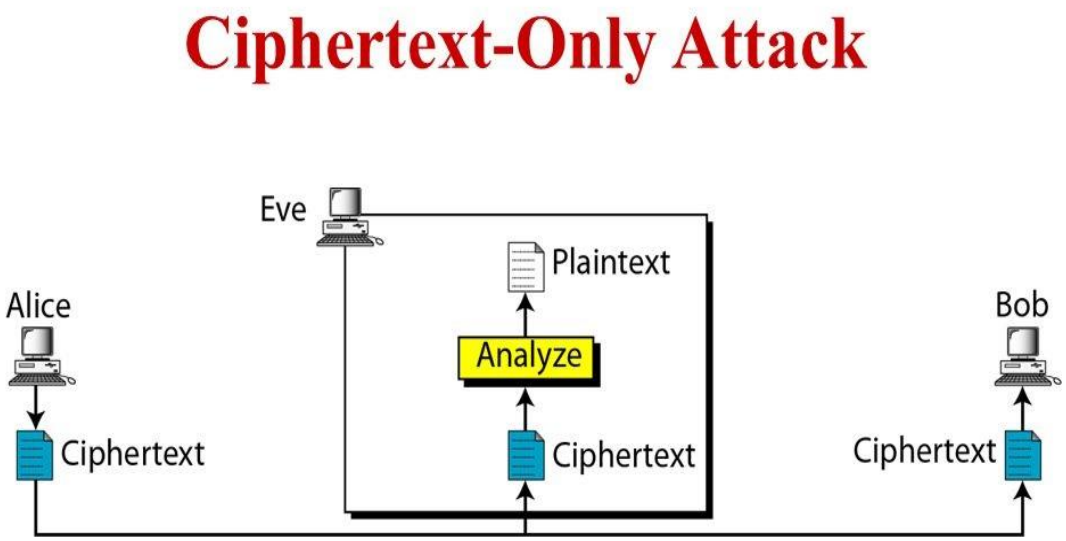
Uz današnja brza računala za otkrivanje sadržaja šifriranih zapisa mogu se koristiti i druga znanja. Jedno od toga su prirodni jezici. Rečenice imaju svoja svojstva i smisao. Ukoliko napadač poznaje svojstva jezika to mu može olakšati dešifriranje sadržaja. U engleskim tekstovima i govoru najčešće se koriste slovo e 13% teksta i slovo t koje je zastupljeno u 9% teksta. Iako je gotovo nemoguće otkriti tablicu preslikavanja metodom pokušavanja, vrlo se lako može zaključiti koji znak iz šifriranog zapisa je slovo e a koji je slovo t pošto se oni najviše pojavljuju.

U svakom jeziku postoje tipične riječi od dva ili tri slova. Kod engleskog jezika to su „on“, „in“, „it“, „and“, „the“. Također tu su i tipični završeci riječi, „ing“, „ion“ te na temelju toga može otkriti parove iz tablice preslikavanja. Ukoliko napadač zna o kojoj je temi riječ to mu isto pomaže. Kada se govori o nekoj temi uvijek postoje riječi koje karakteriziraju određenu temu te se vrlo često pojavljuju. Primjer je na temu „nogomet“ vrlo često se pojavljuje riječ „lopta“.

Probijanje ili razbijanje koda je kada otkrivamo izvorni sadržaj iz šifriranog zapisa sadržaja. Taj šifrirani zapis izvornog sadržaja zove se kod ili kodni zapis. Napad odnosno probijanje koda koje vrši napadač može se izvesti u različitim okolnostima. Tri

osnovne situacije su kad napadač posjeduje samo šifrirani zapis sadržaja, napadač posjeduje izvorne zapise nekih sadržaja i šifrirane zapise istih sadržaja te ukoliko napadač dobije šifrirani zapis onog sadržaja koji je sam odabrao.

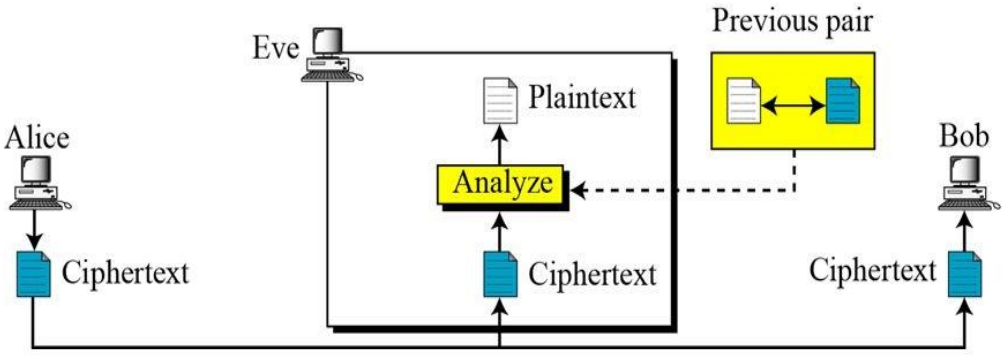
Prvi slučaj u kojem je napadač došao do šifriranog zapisa sadržaja no on ne zna o kojoj se vrti sadržaja radi ni o čemu on govori. To se zove napad samo s šifriranim sadržajem. Primjer se može vidjeti na slici. Ovakva situacija nije obećavajuća za napadača jer nema nikakva dodatna saznanja na temelju čega može probiti šifrirani zapis.



Slika 3. Napadač ima šifru zapisa

(Preuzeto sa: https://slideplayer.com/slide/7911430/)

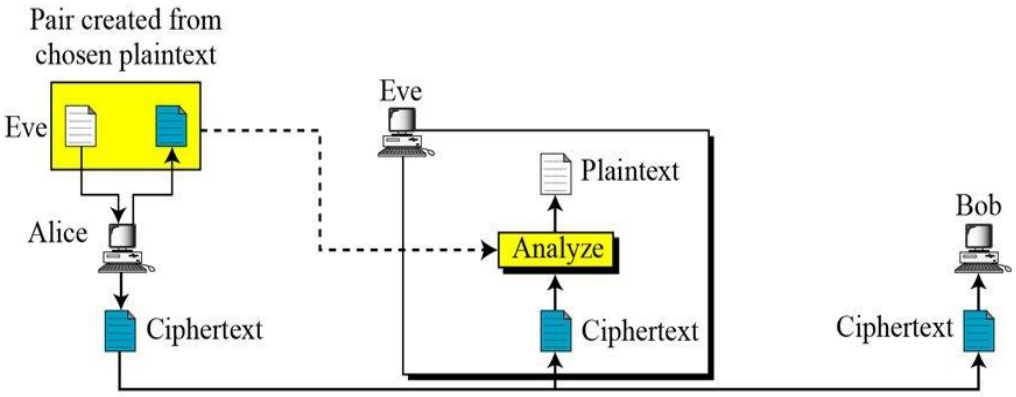
Druga situacija je kada napadač ima izvorne zapise nekih sadržaja te šifrirane zapise istih sadržaja koji su šifrirani prema nekom algoritmu i ključu. To je napad sa poznatim izvornim sadržajem. Slika pokazuje napad sa poznatim izvornim ključem. Kod te situacije napada se algoritam i ključ šifriranja. Napadač ima izvorni zapis sadržaja te šifrirani zapis istog te preko toga pokušava otkriti algoritam i ključ šifriranja. Ukoliko napadač uspije u svojem naumu moći se sve druge poruke dešifrirati koje su šifrirane tim algoritmom i ključem. Ukoliko je sustav zaštite dobar napadač ne bi smio otkriti algoritam i ključ.



Slika 4. Napadač ima izvorne zapise

(Preuzeto sa: https://slideplayer.com/slide/7911430/)

Treći slučaj je onaj u kojem napadač dobije šifrirani zapis točno onog sadržaja koji je odabrao. Napadač može navesti pošiljatelja da šifrira i šalje sadržaj koji napadač želi. Napadač će birati sadržaj za koji smatra da će mu trebati da otkrije algoritam i ključ. Napadač treba biti u situaciji da može kopirati šifriran sadržaj poruka za koje je on naveo pošiljatelja da ih šifrira te pošalje na krajnji cilj. Takva vrsta napada zove se napad sa odabranim izvornim sadržajem. Na slici možemo vidjeti prikaz napada. Ovaj treći slučaj je najpovoljnija situacija za napadača ali će vrlo teško otkriti algoritam i ključ šifriranja ako je sustav napravljen kako treba.



Slika 5. Napadač bira tekst

(Preuzeto sa: https://slideplayer.com/slide/7911430/)

3.1.1. Algoritmi za šifriranje sa simetričnim ključem

DES prethodnici i sljedbenici:	Višenamjenski:
Lucifer	Panama
DES	Sapphire
DESX	
3-DES	Ostali:
Blowfish	CAST
DEAL	CMEA
FEAL	E2
ICE	GOST

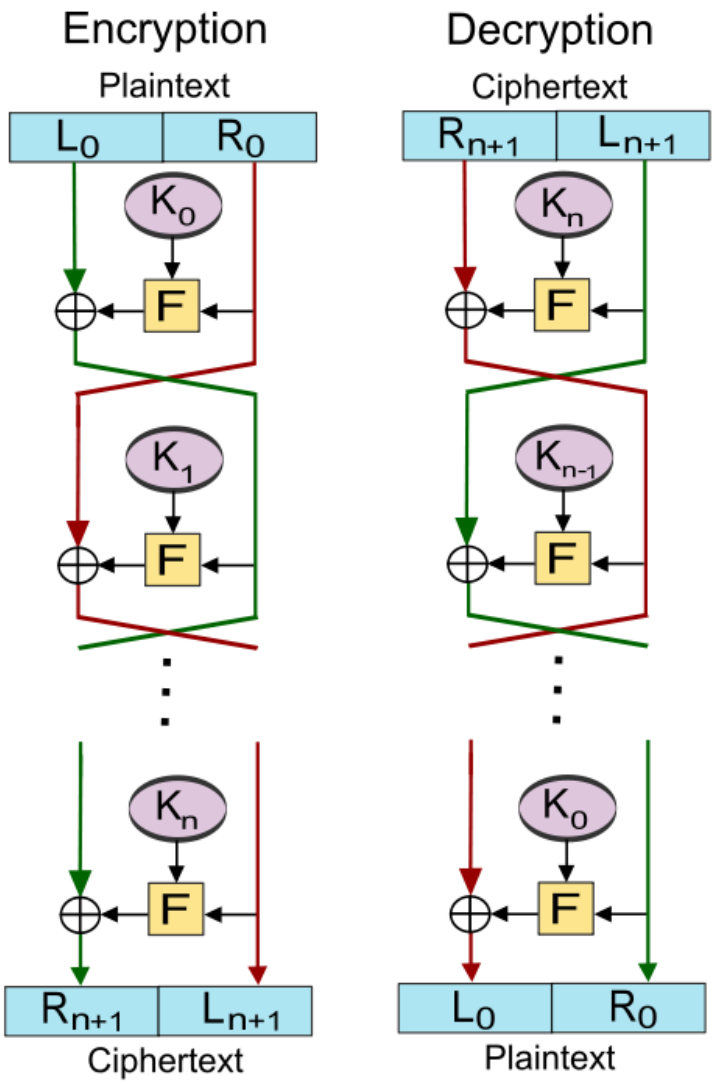
IDEA	Hasty Pudding
Khufu	Misty
MacGuffin	SAFER++
NewDES	SEA
RC2	Skipjack
RC5	Square
Akelarre	Turtle
SHARK	ARC4
	BBC
AES kandidati:	CRAB
AES (Rijndeal)	Crypt
FROG	Crypton
LOKI97-sim	Damond2
LOKI91-sim	DFC
MARS	Khafre
RC6	LOKI89
Magenta	LOKI91
Serpent	MDC
Twofish	MMB
	MPJ
Jednostavni:	NSEA
3-Way	ORYX
ENIGMA	Q128
Solitaire	Quadibloc
TEA	Rainbow
	REDOC
Kriptiranje toka podataka:	S1
A-5	Scop
Helix	Yarro
Pike	
RC4	
SEAL	
SOBER	

WAKE	

Tablica 2. Popis simetričnih algoritama

Prvi simetrični algoritam s blok šifriranjem je Lucifer. Razvio ga je IBM u ranim sedamdesetim godinama. On je prethodnih DES algoritma te je puno jednostavniji od DES-a. Lucifer enkriptira blok veličine 128 bita. Koristi 16 podključeva koji su dužine 72 bita. Kod njega dešifriranje se izvršava inverznom enkripcijom.

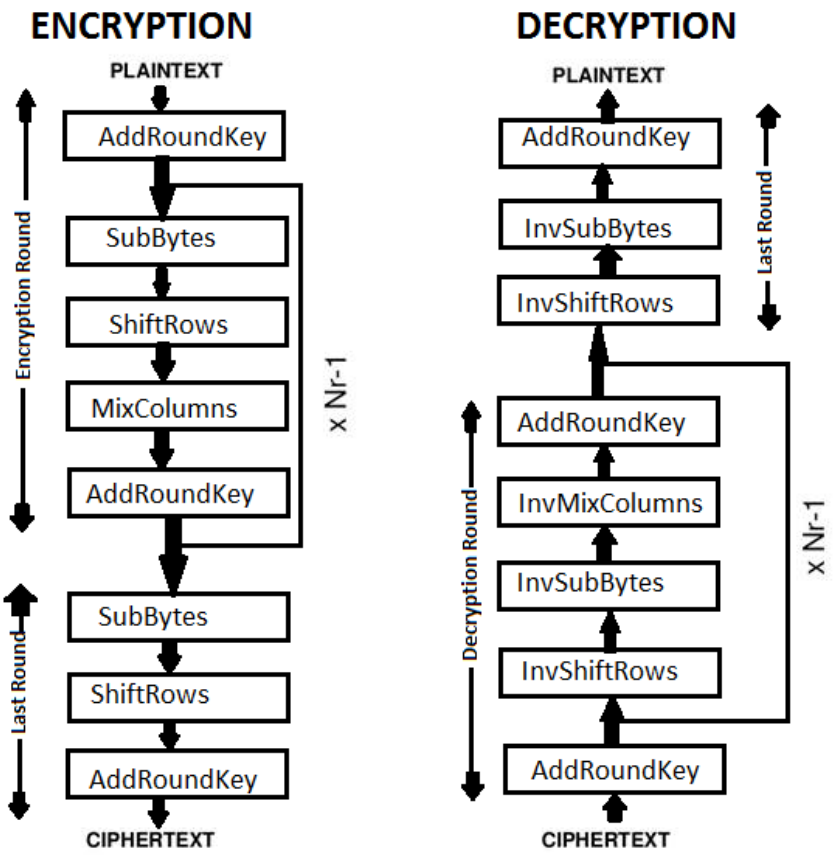
Algoritam za šifriranje sa simetričnim ključem dijelimo u dvije grupe: stream šifriranje i blok šifriranje. Stream šifriranje radi na način da se enkripcija poruke odvija bit po bit dok se kod šifriranja po blokovima, kao što i sam naziv kaže, izvršava po blokovima podataka. Algoritam uzima blok od više bitova (64, 128, 256, ...) te se on enkriptira kao cjelina. Za dešifriranje se izvršava inverzno enkriptiranje. Algoritam je isti, no podključevi enkripcije izvode se obrnutim redom. Slabost Lucifera je u korištenju ključa. Slab je na napad diferencijalne kriptoanalize. U današnje vrijeme on je nesiguran algoritam, ali radi dužine ključa i brzog enkriptiranja moguće ga je koristiti kod enkriptiranja ali samo kada se kombinira sa nekim dobrim simetričnim algoritmom. Primjer dobrog Simetričnog algoritma je DES. On je bio standardni algoritam za enkripciju. DES je izrađen od Lucifera. Enkriptira blok veličine 64 bita. Koristi 16 podključeva dužine 48 bita. Prema statistikama DES je najkorišteniji simetrični algoritam.



Slika 6. DES algoritam

(Preuzeto sa: https://en.wikipedia.org/wiki/Data_Encryption_Standard)

U budućnosti će ga zamijeniti AES koji je puno napredniji i sigurniji algoritam te će on postati standardni algoritam enkripcije. Razlog tome je što su ostali algoritmi nastali prije više od 10 godina a tehnologija je rapidno rasla te su oni zastarjeli. Algoritme kao što je DES u današnje vrijeme moguće je kompromitirati.



Slika 7. AES algoritam

(Preuzeto sa: http://nevonprojects.com/image-encryption-using-aes-algorithm/)

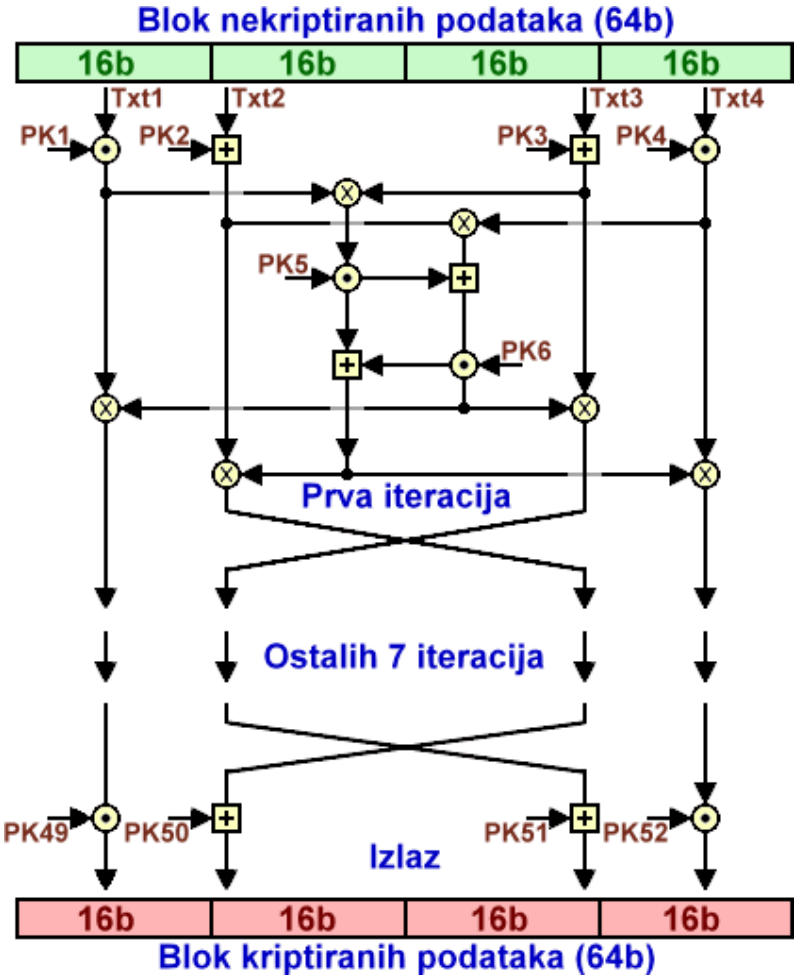
3.1.2. Šifriranje po blokovima

Po računalnoj mreži prenosi se sadržaj koji je zapisan pomoću niza bitova. Jedan takav niz podataka je IP paket. Blok je jedan dio niza (podniz). Takvo šifriranje izvodi se po blokovima. Jedan blok može biti razne veličine, uzet ćemo da je blok 64 bita.

Za šifriranje općenito možemo reći da se preslikava jedan blok u drugi, odnosno jedan ulazni niz bitova preslikava se u jedan izlazni niz bitova iste dužine. Ne može se ulazni niz dužine 128 bita preslikati u izlazni niz dužine 64 bita. Preslikati smatra se zamijeniti jedan niz bitova sa drugim nizom bitova. Dakle kod procesa koji se zove šifriranje jedan se blok treba zamijeniti sa drugim blokom, a on je šifrirani zapis polaznog sadržaja. Proces šifriranja izvodi se

na način da se izvodi za svaki blok posebno. Šifrirani blok slaže se jedan za drugim te iz toga nastaje šifrirani zapis izvornog sadržaja.

Ukoliko šifriramo blokove koji su dužine 64 bita, to nije dobro raditi pomoću parova koji nam pokazuju koji se ulazni blok preslikava na izlazni blok. Skup parova koji izgleda $<B_{ulaz}, B_{Izlaz}>$ ne bi bilo dobro napraviti. U tome B_{ulaz} je 64-bitni niz koji treba šifrirati dok je B_{Izlaz} također 64-bitni niz koji zamjenjuje ulaz i označava ga u šifriranom zapisu. Taj niz koji je dužine 64 bita može dobiti 2^{64} kombinacija bitova odnosno 2^{64} parova koji izgledaju kao $<B_{ulaz}, B_{Izlaz}>$. To je jako puno parova blokova te ih se teško može eksplicitno definirati radi ne prikladnosti za uporabu prilikom provedbe procesa šifriranja. Zbog navedenih razloga algoritam za šifriranje je postavljen za proces gdje se paralelno preslikava 8 dijelova jednog bloka. Svaki pojedini dio dugačak je 8 bitova. Taj niz od 8 bitova poprima 2^8 raznih kombinacija vrijednosti bitova ("0" i "1"). To je sveukupno 256 raznih vrijednosti niza. Prema tome, kod definiranja preslikavanja za sve moguće kombinacije vrijednosti nizova od 8 bita potrebno je 256 parova koji su ovlika <ulazni-8-bitni-niz, izlazni-8-bitni-niz>. Kod svakog od 8 nizova potrebno je definirati jednu tablicu preslikavanja, što nije problem napraviti.



Slika 8. Šifriranje po blokovima

(Preuzeto sa: http://sigurnost.zemris.fer.hr/en/2004_kudelic/index.html)

Kod procesa šifriranja ulazi 64-bitni blok. On se dijeli na 8 jednakih dijelova po 8 bitova. Svaki pojedini dio preslika se prema izrađenoj tablici preslikavanja koja sadrži 8-bitne parove. Prvi elementi su svu mogući ulazni nizovi. Drugi elementi su svi nizovi koji ih zamjenjuju te reprezentiraju ulazne nizove šifriranog zapisa. Kada se prođe kroz proces ovakvog preslikavanja, dobiveni 8 nizova koje se sastoje od 8 bitova čine 64-bitni niz gdje su preslikani bitovi unutar svake osmorke. Osmorke nastaju istim redom koji su bile na početku. Nakon što to završi izvodi se permutacija bitova na razini 64- bitnoga bloka. Na toj razini se permutiraju svi bitovi cijeloga bloka. Dobiveni rezultat izvršene permutacije je krajnji rezultat trećega prolaska kroz algoritam šifriranja jednoga bloka. Zatim se rezultat koji smo dobili pošalje na ulaz algoritma te se proces

šifriranja koji sam opisao, izvodi ponovo. To je ciklički proces šifriranja jednoga bloka a on može sadržavati više desetaka iteracija. Kada se iteriranje završi, rezultat koji smo dobili prilikom posljednjeg prolaska je šifrirani zapis 64 bitnoga bloka. Nakon toga šifrira se idući 64-bitni blok izvornog sadržaja i tako se odvija sve do kraja izvornog sadržaja.

3.2. Šifriranje s javnim ključem

Simetričan kriptosustav temelji se na ideji da pošiljatelj i primatelj biraju tajni ključ k_i te na temelju njega iz kriptosustava dobiju funkcije za kriptiranje i dekriptiranje. Dekriptiranje d_k je ista funkcija kao i kriptiranje. Dešifriranje se vrši obrnutim redoslijedom od šifriranja. Veliki nedostatak je sigurnost jer ona ovisi i tajnosti ključa. Prije nego se izvrši šifriranje, pošiljatelj i primatelj trebaju izmijeniti ključeve putem nekog drugom, sigurnog komunikacijskog kanala. Šifriranje velikog broja poruka dosta smanjuje sigurnost pa zato pošiljatelj i primatelj trebaju češće mijenjati ključeve. Šifriranje s javnim ključem ima e_k (šifriranje) te d_k(dešifriranje). Za njih vrijedi:

- d_k je inverzno od e_k za svaki K
- d_k je poznat samo osobi K dok je e_k javan
- e_k je jednosmjerna funkcija

Ukoliko pošiljatelj A želi poslati određenu poruku z primatelju B, tada osoba B prvo šalje osobi A svoj javni ključ e_B. Nakon toga šifrira poruku uz pomoć e_B te šalje primatelju šifrat y = $e_B(z)$. Na kraju osoba B dešifrira šifrat tako što koristi svoj javni ključ d_B te dobije

$$d_B(\mathrm{y}) = d_B(e_B(z)) = \mathrm{z}$$

Jedan primjer komuniciranja osobe A i B:

1. Osoba B pošalje osobi A svoj javni ključ $k_{B^{javni}}$
2. Osoba A pošalje osobi B svoj javni ključ $k_{A^{javni}}$
3. Osoba B pošalje osobi A poruku koja je šifrirana A sa javnim ključem: C_{B-A} = E$k_{A^{javni}}$ (p_{B-A})

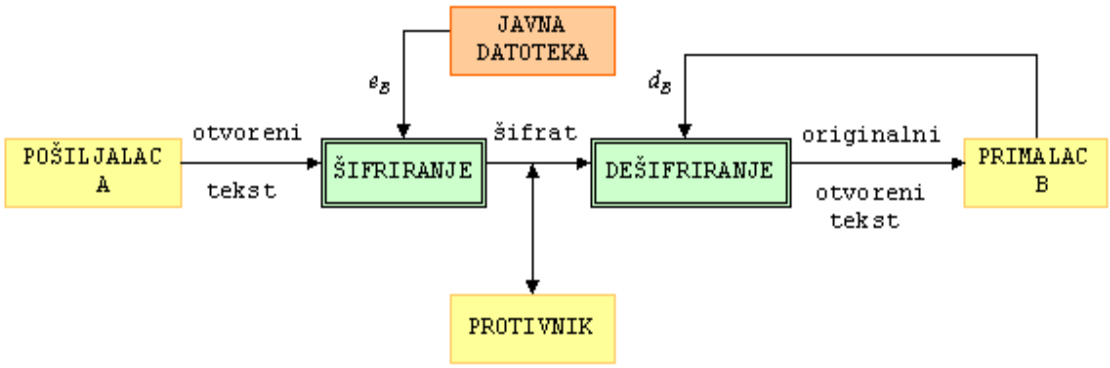
4. Osoba A treba dešifrirati poruku sa privatnim ključem:
   p_{B-A} = D$k_{A^{privatni}}$ (C_{B-A})
5. Osoba A pošalje osobi B poruku koja je šifrirana sa njezinim javnim ključem: C_{A-B} = E$k_{B^{javni}}$ (p_{A-B})
6. Osoba B dešfirira poruku sa privatnim ključem p_{A-B} = D$k_{B^{privatni}}$ (C_{A-B})

Bilo tko može imati pristup funkciji e_B te je moguće da se lažno predstavlja kao osoba A. Taj problem može se riješiti tako da:

- Osoba A pridoda u svojoj poruci neki broj a od nekoliko znamenki, npr. 10
- Osoba B generira svoj slučajan broj od 10 znamenki i šalje osobi A poruku $e_A(a+b)$
- Osoba A izračuna b uz pomoć formule $b = d_A(e_A(a + b)) - a$. Nakon toga ponovo šalje svoju prvu poruku uz to da joj pridoda b, te identično napravi sa idućom porukom koju šalje osobi B

Pojedini kriptosustavi omogućuju korisnicima digitalno pisanje poruke. Ta stvar je bitna jer poslije osoba A ne može poreći da je baš on poslao određenu poruku. Ukoliko pretpostavimo da je $P = C$ onda osoba A može potpisati poruku z na način da osobi B šalje šifrat $r = d_A(y) = d_A(e_B(x))$. Kad osoba B dobije poruku za koju smatra da ju je poslala osoba A, tada on upotrijebi javni ključ e_A a nakon toga svoj tajni ključ d_B:

$$d_B(e_A(\mathrm{r})) = d_B(e_A(d_A(e_B(\mathrm{x})))) = \mathrm{x}.$$



Slika 9. Šifriranje sa javnim ključem

(Preuzeto sa: https://web.math.pmf.unizg.hr/~duje/kript/idejajavni.html)

Kada uspoređujemo kriptosustave sa javnim ključem i kriptosustave sa simetričnim ključem možemo vidjeti da prvonavedeni imaju dosta prednosti. Neke od prednosti su:

- nije potrebno koristiti sigurni komunikacijski kanal za razmjenu ključeva
- kod komunikacije grupe od N osoba potrebno je 2N ključeva
- najveća prednost je mogućnost potpisa poruke

Kod moderne kriptografije koja se koristi u komercijalnom svijetu svakodnevni prizor je da osoba A hoće kupiti nešto od osobe B putem interneta. Tu se javljaju određeni problemi:

1. Povjerljivost: Poruku koju osoba A pošalje osobi B ne može pročitati nitko drugi
2. Vjerodostojnost: osoba B zna da je samo osoba A mogla poslati poruku koju je ona primila
3. Netaknutost: osoba B zna da je poruku koju je poslal osoba A nije promijenjena kod slanja
4. Nepobitnost: osoba A ne može poslije zanijekati da je upravo ona poslala poruku osobi B

3.2.1. RSA algoritam

Najpopularniji algoritam za kriptiranje sa javnim ključem je RSA. On je objavljen 1978. godine. Njega su stvorili Ronald Rivest, Adi Shamir i Leonard Adelman. Po prvim slovima njihovih prezimena sustav je dobio ime RSA. Sigurnost algoritma zasniva se na složenosti faktorizacije velikih brojeva. Javni i tajni ključ određuju se odabirom para velikih prostih brojeva. Sigurnost algoritma zasniva se na činjenici da:

- vrlo je lako odrediti je li veliki broj prost te pomnožiti dva velika prosta broja
- vrlo je zahtjevno faktorizirati veliki broj koji je umnožak dva velika prosta broja

Implementacija RSA algoritma vrši se na sljedeći način:

- prvo se odaberu dva velika prosta broja p i q te se oni pomnože
  n = p * q
- kad se postavi r = (p-1) * (q-1) dobiva se m^r = 1 mod n gdje je za svaki m uzajamno prosto s n
- zatim se odabere slučajni broj e koji je uzajamno prost sa r
- nakon toga se odredi d na način da vrijedi : e * d = 1 mod r

Operacija šifriranja i dešifriranja definira se ovako:

- $e_{(e,n)}$(m) = m^e mod n
- $d_{(d,n)}$(c) = c^d mod n

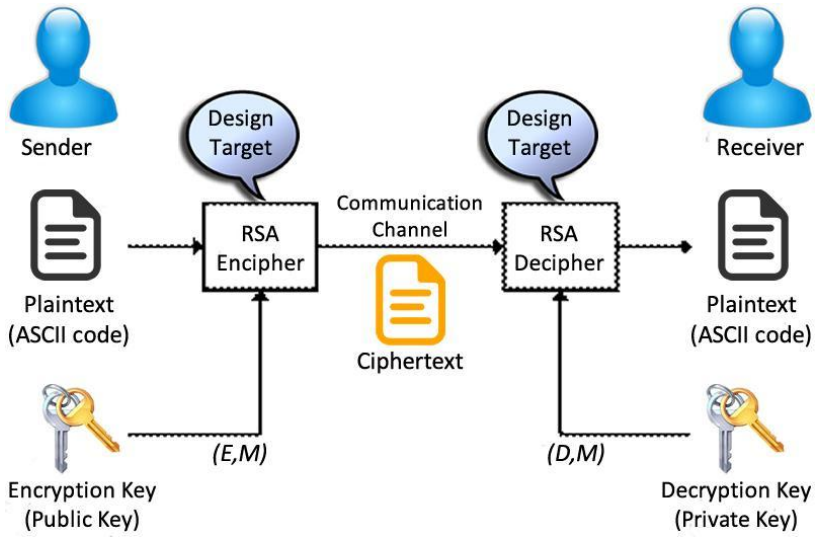
Primjer RSA algoritma

Ukoliko osoba A hoće poslati osobi B poruku m, ona preuzima od osobe B javni ključ (e, n) sa servera te izračuna c = m^e mod n te pošalje osobi B šifrat c. Osoba B primi šifrat c te ga dešifrira sa svojim privatnim ključem m = c^d mod n. Poruka m obavezno je manja od n. Radi toga pošiljatelj dijeli poruku na blokove. Njihova vrijednost je manja od n te ih parcijalno šifrira.

Uzmimo da je p = 3 i q = 11. Dobijemo da je n = p * q, tj. n = 3 * 11 = 33 i r = 20. Eksponent e treba biti relativno prost s 20. Uzeti ćemo da je e = 7. U tome slučaju d = 3. Naš javni ključ je (n, e) = (33, 7). Netko nam hoće poslati poruku x = 17. Zatim treba izračunati e_K(x)=17^7 mod 33:

17^7 = 17 * 17^2 * 17^4 = 17 * 25 *(-2) = -25 = 8 (mod 33).

Šifrat je y = e_K(x) = 8



Slika 10. RSA algoritam

(Preuzeto sa: https://www.researchgate.net/figure/RSA-algorithm-structure_fig2_298298027)

4. Integritet poruke i autentičnost komunikatora

Pod pojmom integritet poruka smatramo čuvanje izvornoga sadržaja poruka prilikom procesa prijenosa iste od strane pošiljatelja do primatelja. Smatramo da je integritet poruke sačuvan ukoliko je primatelj dobio identičnu poruku kakvu mu je pošiljatelj poslao. Ukoliko nije tako kažemo da je integritet poruke u procesu prijenosa narušen.

Autentičnost komunikatora odnosi se na pitanje je li stvaran pošiljatelj nekoga teksta upravo ta osoba ili sustav za koji se on predstavlja. Ukoliko osoba C pošalje poruku osobi A te se predstavi kao osoba B, onda osoba B nije autentični pošiljatelj. Osoba C se lažno predstavlja tvrdeći da je to osoba B. Smatramo da je autentičnost komunikatora narušena kada se osoba koja šalje poruku lažno predstavlja.

Autentičnost poruke nije narušena ukoliko poruka koju je osoba A poslala osobi B bude identična kada ju osoba B bude čitala, odnosno ako sadržaj poruke bude isti te ako je pošiljatelj autentičan, tj. onaj za kojeg se predstavlja. Smatramo da je autentična poruka ukoliko je očuvan njezin integritet sadržaja te ukoliko je pošiljatelj onaj za kojeg se predstavlja.

Vrlo je važno zaštititi sadržaj koji se prenosi mrežom. Posebno je bitno u poslovnom segmentu gdje se bazira posao na financijama. No provjera integriteta sadržaja i autentičnost komunikatora važnija je za uspješniji rad računalnih mreža. Ne mora nužno samo osoba biti komunikator. To je bilo koji entitet koji pošalje i primi neku poruku. Tih entiteta ima u svim razinama računalnih mreža. Npr. to mogu biti mrežne kartice te usmjerivači.

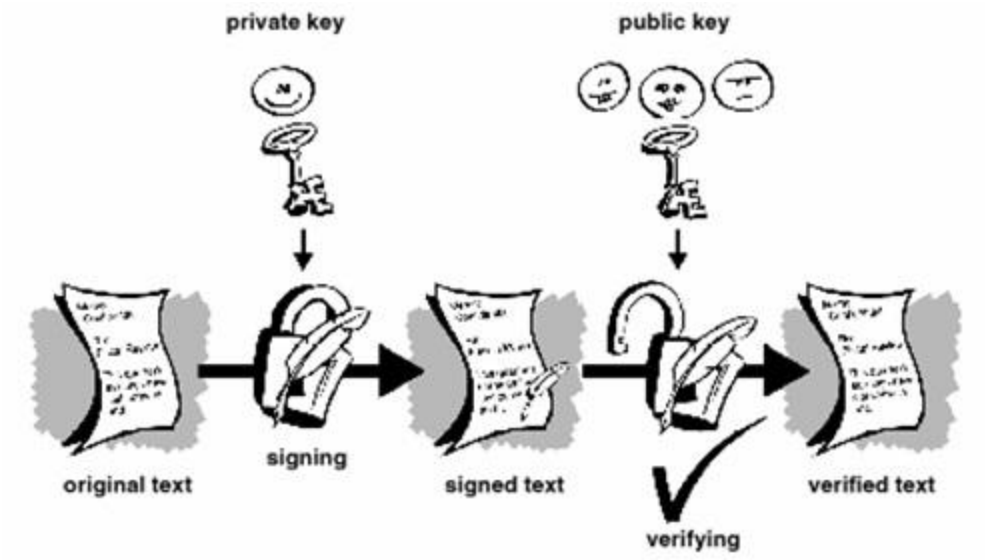
Usmjerivači cijelo vrijeme među sobom izmjenjuju pakete sa podacima o stanju veze te na temelju njih održavaju svoje tablice usmjeravanja i prosljeđivanja. Pomoću tablica prosljeđivanja oni proslijede svoje pakete prema cilju. Napadač može poremetiti rad računalne mreže tako što mijenja sadržaj paketa u prijenosu. Ukoliko se to dogodi, usmjerivači izrade krive tablice usmjeravanja te pošalju pakete na krivo odredište.

Ukoliko se iskrivi sadržaj paketa dolazi do problema pošto usmjerivači izrade krive tablice usmjeravanja te pošalju paket na krivu adresu. Ako se tako nešto dogodi, ugrožava se rad mreže.

Još jedan problem je lažno predstavljanje. Ukoliko se napadač lažno predstavi, ispostaviti će se da je on usmjerivač te će slati izmišljene podatke o stanju veze te ugroziti rad mreže. Radi toga su za dobar rad mreže vrlo bitni zaštita integriteta poruke i autentičnost komunikatora. To je dosta bitno kod računalne mreže kao prijenosnog sustava.

4.1. Digitalni potpis

Digitalni potpis je sustav kriptografskog potpisa koji omogućuje da osoba potpiše sadržaj zapisan u digitalnom obliku koji se prenosi računalnom mrežom. On je jedinstven u cijelom svijetu, mora biti takav ga nije moguće falsificirati te da je moguće provjeriti njegovu autentičnost.



Slika 11. Digitalni potpis

(Preuzeto sa: http://www.maturski.org/INFORMACIONI%20SISTEMI/DigitalniPotpis%20.html)

Sustav šifriranja koji koristi javni i privatni ključ, jedan par javnog i privatnoga ključa koji mu pripada dodjeljuje jednom korisniku. Radi toga svi takvi parovi ključeva mogu imati ulogu jednoznačnoga identifikatora. Kod sustava za šifriranje sa javim ključem K_A te privatnim K_B vrijedi:

$K_B\ (K_A(\mathrm{m})) = K_A(K_B(\mathrm{m})) = \mathrm{m}$

Lijeva strana zapisa govori slijedeće. Šifrirani zapis poruke m koja ima šifru sa javnim ključem K_A osobe A, mora se dešifrirati sa odgovarajućim privatnim ključem K_B od osobe A. Tako se dobije izvorni zapis sa sadržajem poruke m.

Desna strana zapisa govori slijedeće. Šifrirani zapis poruke m koja ima šifru sa privatnim ključem K_B osobe B, mora se dešifrirati sa odgovarajućim javnim ključem K_A od osobe A. Tako se dobije izvorni zapis sa sadržajem poruke m.

Ona poruka koja je šifrirana sa javnim ključem treba se dešifrirati sa privatnim ključem. Isto vrijedi i obrnuto, ona poruka koja je šifrirana sa privatnim ključem treba se dešifrirati sa javnim ključem. Onaj tko ima par ključeva (javni, privatni), može šifrirati svoje poruke sa privatnim ključem kao digitalni potpis.

Osoba A treba šifrirati sadržaj poruke m sa privatnim ključem K_B te poslati šifrirani zapis K_B(m) osobi B koja prima poruku. Zatim osoba B dešifrira taj zapis putem svojeg javnoga ključa K_A te tako dobije izvorni zapis sadržaja poruke m.

Kada se šifrira sadržaj sa privatnim ključem K_B osoba A koja je pošiljatelj ujedno digitalno potpisuje sadržaj. Šifrirane poruke od osobe A može dešifrirati tko god hoće zato što je javni ključ K_A javno poznat. Ono što je šifrirano sa privatnim ključem K_B od strane osobe A moguće je jedino dešifrirati sa njegovim javnim ključem K_A.

5. Protikoli

5.1. SSL/TLS

Transport Layer Security(TLS) i njegov prethodnik Secure Socket Layer (SSL) koji je zastario su kriptografski protokoli koji pružaju sigurnost komunikacije preko računalne mreže. Nekoliko verzija tih protokola našlo je široku uporabu u aplikacijama kao što su web preglednik, e-pošta, instant poruke te VoIP. Web stranice su u mogućnosti koristiti TLS kako bi osigurali komunikaciju između servera i web preglednika. TLS protokol se sastoji od dva sloja. To su TLS protokol zapisa i TLS protokol rukovanja.

TLS protokol prvenstveno ima za cilj osigurati tajnost i integritet podataka u komunikaciji dvije ili više računalne aplikacije. Pod sigurnosti TLS-a, komunikacija između klijenta (web preglednik) i servera ima jednu ili više sljedećih svojstava:

- veza je privatna jer se koristi simetrična kriptografija za šifriranje podataka. Ključevi za simetrično šifriranje generiraju se jedinstveno za svaku vezu i temelje se na tajnosti dogovorenoj na početku. Server i klijent detaljno dogovaraju koji algoritam i kriptografski ključ će se koristiti prije nego krene prijenos podataka. Dogovori su tajni, sigurni i pouzdani, napadač ne može izmijeniti podatke tijekom komunikacije bez da bude otkriven.
- identitet komunikatora može se provjeriti pomoću javnog ključa. Autentičnost se ne mora izvršiti ali je općenito potrebna samo za jednu stranu. Obično je to server.
- veza je pouzdana jer svaka poruka koja se prenosi uključuje provjeru integriteta poruke koristeći sigurnosni kod kako bi se spriječio gubitak ili promjena podataka prilikom prijenosa.

Uz navedena svojstva, pažljivo konfiguriranje TLS protokola može pružiti dodatna svojstva kao što su napredna tajnost, osiguranje da svako buduće objavljivanje ključeva za šifriranje se ne može koristiti za dešifriranje TLS komunikacijskih zapisa u prošlosti. TLS podržava mnogo različitih metoda za razmjenu ključeva, šifriranje podataka i provjeru autentičnosti i integriteta poruke. Kao rezultat toga, sigurnost konfiguracije TLS-a uključuje mnoge parametre.

TLS protokol je unaprijeđen nekoliko puta kako bi se riješili opasnosti od sigurnosnih prijetnji. Programeri su također unaprijedili sigurnost web preglednika tako što su otkrili slabosti te ih otklonili.

SSL protokol je imao tri verzije (jedna nije bila izdana), a TLS ima četiri verzije. Najnovija verzija TLS 1.3 izdana je u kolovozu 2018. godine, prije manje od mjesec dana. Prethodna verzija bila je na snazi deset godina.

Protokol	Godina izdanja
SSL 1.0	Nije objavljen
SSL 2.0	1995.
SSL 3.0	1996.
TLS 1.0	1999.
TLS 1.1	2006.
TLS 1.2	2008.
TLS 1.3	2018.

Tablica 3. Povijest SSL/TLS protokola

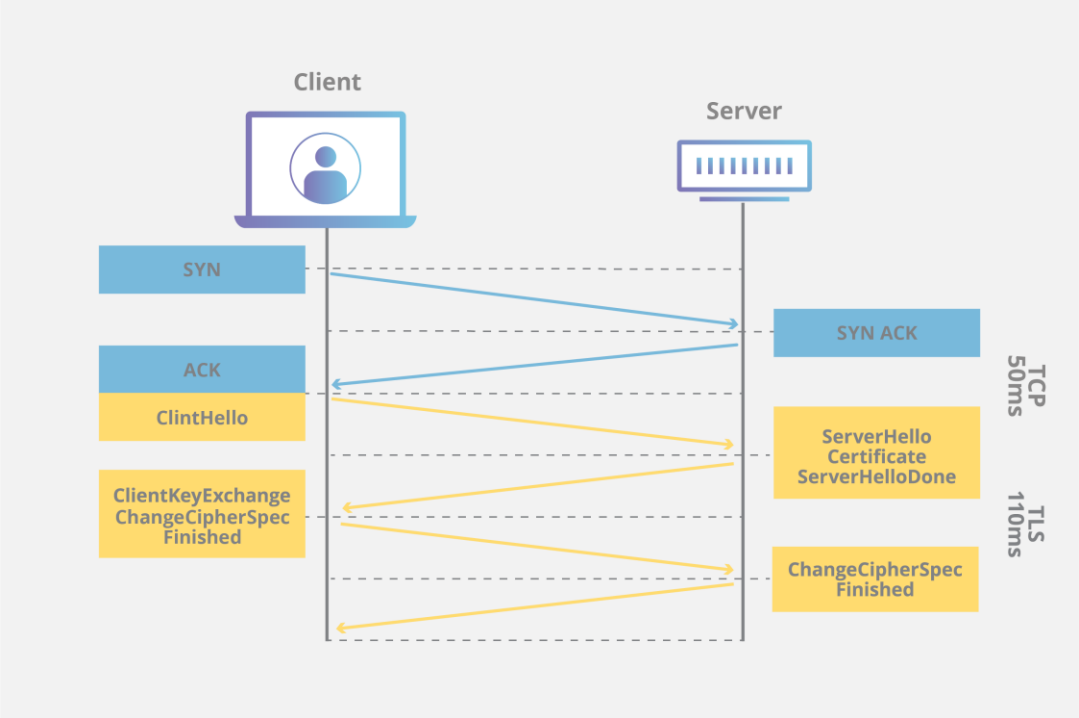
5.1.1. Opis TLS-a

TLS protokol koristi aplikaciju klijent-server za komunikaciju na mreži. On onemogućuje prisluškivanje i neovlašteno mijenjanje sadržaja.

Budući da aplikacije mogu komunicirati sa ili bez TLS protokola, za klijenta je neophodno ukazati serveru kako postaviti TLS vezu. Jedan od glavnih načina za postizanje toga je korištenje različitih „portova“ za TLS vezu, npr. port 443 za HTTPS. Drugi mehanizam je za klijenta da napravi zahtjev serveru za prebacivanje veze na TLS.

Jednom kada klijent i server pristanu koristiti TLS, oni dogovaraju vezu za prijenos tako da koriste protokol za rukovanje. Protokoli koriste rukovanje sa asimetričnom šifrom da uz šifru uspostave i specifične zajedničke ključeve koji se u daljnjoj komunikaciji šifriraju putem simetričnog šifriranja. Tijekom rukovanja, klijenti i server dogovore razne parametre koji koriste za uspostavu sigurnosti veza:

- rukovanje počinje kada je omogućeno povezivanje klijenta na TLS, server šalje zahtjev za sigurnu vezu a klijent predstavlja popis sigurnih šifri.
- iz toga popisa server uzima šifru i hash funkcijom podržava i obavještava klijenta o odluci.
- server obično osigurava identifikaciju u obliku digitalnog certifikata. Certifikat sadrži ime servera, certifikatski autoritet (CA) koji jamči autentičnost certifikata i javni ključ servera.
- prije nastavka, klijent potvrđuje valjanost certifikata
- za generiranje ključeva koji se koriste za sigurnu vezu, klijent koristi jedan od ova dva načina:
    1) šifrira slučajni broj sa serverovim javnim ključem i šalje rezultat serveru koji jedino server može dešifrirati sa njegovim privatnim ključem. Obje strane koriste slučajni broj za generiranje jedinstvenog ključa za šifriranje i dešifriranje
    2) koristi Diffie-Hellmanovu metodu za sigurno generiranje jedinstvenog ključa za šifriranje i dešifriranje koji ima dodatna svojstva za naprednu tajnost. Ako serverov privatni ključ bude otkriven u budućnosti, ne može biti korišten kod dešifriranja čak i ako je zapis presrela ili vidjela treća osoba



Slika 12. SSL/TLS rukovanje

(Preuzeto sa: https://www.cloudflare.com/learning/cdn/cdn-ssl-tls-security/)

Time završava rukovanje i počinje sigurna veza koja koristi isti ključ za šifriranje i dešifriranje do završetka veze. Ukoliko neki od navedenih koraka ne uspije, tada TLS rukovanje nije uspjelo i veza nije kreirana.

TLS/SSL nisu dio OSI modela ili TCP/IP modela. TLS je pokrenut „na vrhu nekog pouzdanog protokola“, što bi značilo da je iznad transportnog sloja. On služi za šifriranje viših slojeva. Međutim aplikacije uglavnom koriste TLS protokol kao da je transportni sloj iako aplikacije koje koriste TLS moraju aktivno nadgledati rukovanje i razmjenu autentičnosti certifikata.

5.2. HTTPS protokol

Hypertext Transfer Protocol Secure (HTTPS) je produžetak Hypertext Transfer Protocol (HTTP) za sigurnost komunikacije putem računalne mreže i koristi se na Internetu. U HTTPS-u komunikacijski protokol je šifriran pomoću TLS-a. Radi toga se protokol često naziva HTTP preko TLS-a ili HTTP preko SSL-a pošto je SSL preteča TLS-a.

Glavna zadaća HTTPS-a je autentikacija pristupa web stranicama i zaštita privatnosti i integriteta podataka u razmjeni. Ona štiti od takozvanih „man in the middle“ napada. Dvosmjerno šifriranje u komunikaciji klijent server štiti protiv prisluškivanja i upadanja u komunikaciju. To znači da protokol pruža sigurnu komunikaciju bez uplitanja napadača.

Povijesno gledano, HTTPS veze su uglavnom korištene kod transakcija za plaćanje na webu, e-pošte te kod osjetljivih transakcija u korporativnim informacijskim sustavima. Od 2018. godine korisnici češće koriste HTTPS nego originalni HTTP, prvenstveno da zaštite autentičnost stranice, sigurnost računa, te zadrže privatnost komunikacije i identiteta.

Slika 13. HTTPS protokol

(Preuzeto sa: https://www.x-cart.com/blog/what-is-https-and-ssl.html)

5.2.1. Opis HTTPS-a

Uniform Resource Identifier (URI), shema HTTPS-a ima identičnu sintaksu kao shema za HPPT. Međutim, HTTPS signalizira pregledniku da koristi dodatnu sloj šifriranja SSL/TLS

protokola da bi komunikacija bila sigurna. SSL/TLS je posebno prikladna kod HTTP-a jer može pružiti određenu zaštitu čak i ako je samo jedna strana u komunikaciji autentična. To je slučaj kod HTTP transakcije preko Interneta gdje je samo server autentičan.

HTTPS kreira sigurnosni kanal preko mreže koja nije sigurna. Tako osigurava dovoljnu razinu zaštite od prisluškivanja i „man in the middle“ napada pod uvjetom da se koriste prikladne šifre i da je certifikat servera potvrđen i pouzdan.

HTTP protokol može biti šifriran. To uključuje URL zahtjev, parametre upita, zaglavlja i kolačiće. Međutim, zbog adrese domaćina i broja port-a, oni su nužno dio TCP/IP protokola, HTTPS ne može zaštititi njihovo otkrivanje. To znači da čak i kod pravilno konfiguriranoga web servera, oni koji prisluškuju mogu otkriti IP adresu, broj port-a web server, osobe koje komuniciraju, trajanje komunikacije ali ne i sadržaj komunikacije.

Web preglednici znaju kako vjerovati HTTPS web stranicama temeljenim na certifikatu koji su unaprijed ugrađeni u softver. Korisnik mora imati povjerenje u HTTPS vezu na web stranice radi idućih stvari:

- korisnik vjeruje da je softver ispravno implementiran sa ispravno instaliranim certifikatom
- korisnik vjeruje da certifikat jamči jedino za legalne web stranice
- web stranica posjeduje važeći certifikat, što znači da je izvor pouzdan
- certifikat ispravno identificira web stranice
- korisnik vjeruje da je šifriranje dovoljno sigurno protiv napadača

HTTPS je posebno važan kod nesigurnih mreža kao što su javne Wi-Fi pristupne točke jer bilo tko na istoj lokalnoj mreži može otkriti osjetljive informacije koji nisu zaštićene HTTPS protokolom. Osim toga, mnoge besplatne za korištenje a plaćene WLAN mreže uključuju u paket injekcije kako bi prikazali vlastite reklame na web stranicama. Međutim, to može biti zloupotrebljeno na razne načine, kao što je ubacivanje raznih malware-a na web stranice i krađa korisnikovih osobnih podataka.

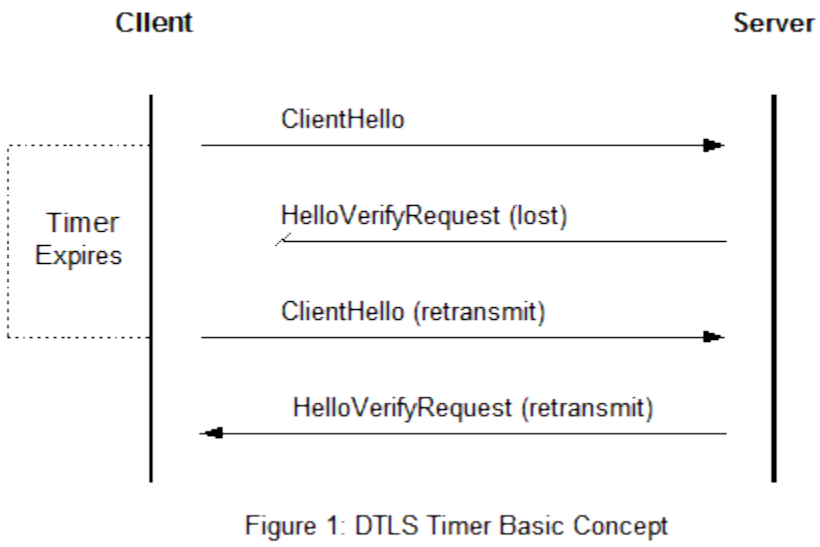
HTTPS je jako bitan kod povezivanja preko anonimne mreže. Zlonamjerni čvorovi mogu uništiti ili promjeniti sadržaj poruke koja prolazi kroz nesigurnu mrežu. Radi toga je Electronic Frontier Foundation i Tor projekt počeo razvijati HTTPS svugdje u Tor preglednik paketu.

Pošto se pojavljuje sve više kriminala i krađe osobnih podataka na Internetu, korištenje HTTPS sigurnosti vrlo je bitno bez obzira na vrstu internetske veze koja se koristi.

5.3. DTLS protokol

Datagram Transport Layer Security (DTLS) je komunikacijski protokol koji pruža sigurnost za aplikacije temeljenje na datagramu dopuštajući im da komuniciraju na način koji je napravljen da bi spriječio prisluškivanje ili krivotvorenje poruka. DTLS je protokol koje se temelji na TLS protokolu te je namijenjen za pružanje sličnih sigurnosti.

DTLS 1.0 definiran je kao delta verzija od TLS 1.1. Iduća verzija je DTLS 1.2. Verzija DTLS 1.1 je preskočena iz razloga da se usklade brojevi sa TLS verzijom.



Slika 14. DTLS protokol

(Preuzeto sa: https://www.vocal.com/networking/datagram-transport-layer-security-dtls/)

5.3.1. Opis DTLS-a

Osnovni koncept DTLS-a je konstruirati TLS preko datagrama transporta. Razlog da se TLS ne može konstruirati direktno u datagram je to što paketi mogu biti izgubljeni ili promijeniti

redoslijed. TLS nema unutarnje objekte za obradu ove vrste nepouzdanosti. Svrha DTSL-a je samo da izvrši minimalne promjene na TLS-u da se riješi taj problem. Kad je potrebno učiniti neke preinake, uvijek se gleda da se očuva stil TLS-a.

Nepouzdanost stvara probleme za TLS na dvije razine:

- TLS ne dopušta neovisno dešifriranje pojedinačnih zapisa. Budući da provjera integriteta ovisi o rednom broju, ako zapis N nije primljen, tada provjera integriteta na zapis N+1 će biti provedena za krivi redni broj i time neće uspjeti
- Sloj TLS rukovanja predstavlja pouzdano dostavljanje poruke te zaustavlja proces ukoliko su poruke izgubljene.

Zaključak

U današnje vrijeme spyware, spam te razne internetske prevare sve više se mogu naći na naslovnicama novina. Tehnologija se mijenja iz dana u dan tako da svatko tko koristi računalo mora biti na oprezu i imati neko sigurnosno riješenje. Korištenje Interneta može biti dosta opasno jer skoro svi naši podaci su dostupni i ako zaštita nije dobra ili je zastarjela, napadači se mogu time okoristiti. Ako koristimo prave alate i sigurne stranice možemo izbjeći mnogobrojne probleme i opasnosti radi kojih možemo izgubiti puno i naći se u zamci.

Ukoliko se želimo dobro zaštititi preporučljivo je redovito ažurirati sigurnosne zaštite te izbjegavati stranice kod kojih nije prisutan HTTPS protokol. Kada kreiramo lozinku nikako nije dobro koristiti ime, prezime, datum ili godinu rođenja jer je to vrlo opasno. Najbolje je koristiti velika i mala slova, brojke i ostale znakove tako da lozinka ne tvori smislenu riječ. Takve lozinke gotovo je nemoguće probiti.

Iako Internet „nije za svakoga“, potrebno je biti oprezan jer mali postotak korisnika Interneta su prevaranti, lopovi i kriminalci koji se žele okoristiti tuđom ne pažnjom. Naravno da korištenje Interneta ne treba izbjegavati no treba biti na oprezu i paziti kakve stranice otvaramo. Kao što pazimo kome otvaramo ulazna vrata na kući tako moramo i paziti koje stranice pregledavamo. Ljudi u od davnina naučili raditi razne stvari pa tako se može i naučiti kako paziti na sigurnost u web komunikaciji.

Literatura

1. Radovan, Mario (2011) Računalne mreže (2): Prijenos, mrežne usluge i zaštita, Rijeka: Digital point, 2011.
2. D. Pleskonjić, N. Maček, B. Đorđević, M. Carić(2007). Sigurnost računarskih sistema i mreža. Beograd, Mikro knjiga
3. Stephen A. Thomas: SSL & TLS Essentials, Securing the Web, Wiley Computer Publishing, 2000.
4. Rolf Oppliger, Ralf Hauser, David Basin: SSL/TLS Session-Aware User Authentication, Computer, ožujak 2008., pp. 59-65.
5. RSA Security Technology Backgrounder, Enhancing One-Time Passwords for Protection Against Real-time Phishing Attacks, dostupno na adresi: www.rsa.com/rsalabs/technotes/One-TimePWWP.pdf (4.12.2008).
6. A. Dujella, M Maretić (2007.) Kriptografija, Tisak

Popis priloga