Metode za enkripciju elektroničke pošte

Galić, Bernard

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:846540

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

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

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

zir.nsk.hr

Image not found or type unknown

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

DIPLOMSKI STUDIJ

METODE ZA ENKRIPCIJU ELEKTRONIČKE POŠTE

Diplomski rad

Bernard Galić

Osijek, 2018.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za obranu diplomskog rada

Osijek, 11.09.2018.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za obranu diplomskog rada

Ime i prezime studenta:	Bernard Galić
Studij, smjer:	Diplomski sveučilišni studij Elektrotehnika, smjer Komunikacije i informatika'
Mat. br. studenta, godina upisa:	D 976, 21.09.2017.
OIB studenta:	39956982213
Mentor:	Izv. prof. dr. sc. Krešimir Grgić
Sumentor:	
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	Prof.dr.sc. Drago Žagar
Član Povjerenstva:	Doc.dr.sc. Višnja Križanović
Naslov diplomskog rada:	Metode za enkripciju elektroničke pošte
Znanstvena grana rada:	**Telekomunikacije i informatika (zn. polje elektrotehnika)**
Zadatak diplomskog rada:	Usluga elektroničke pošte (e-mail) već dugi niz godina predstavlja jednu od najvažnijih i najkorištenijih internetskih usluga. Kao takva, često je izložena napadima i zlouporabi, pa je briga o njezinim sigurnosnim aspektima izuzetno važna. Potrebno je istražiti i analizirati suvremene metode za enkripciju elektroničke pošte. Potrebno je implementirati nekoliko različitih metoda za enkripciju elektroničke pošte, te ih međusobno usporediti.
Prijedlog ocjene pismenog dijela ispita (diplomskog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 2 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 2 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	11.09.2018.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 02.10.2018.

Ime i prezime studenta:	Bernard Galić
Studij:	Diplomski sveučilišni studij Elektrotehnika, smjer Komunikacije i informatika'
Mat. br. studenta, godina upisa:	D 976, 21.09.2017.
Ephorus podudaranje [%]:	8%

Ovom izjavom izjavljujem da je rad pod nazivom**: Metode za enkripciju elektroničke pošte**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Krešimir Grgić

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

Sadržaj

1. UVOD 1
2. ZAŠTITA ELEKTRONIČKE POŠTE 2
2.1. Elektronička pošta 2
2.2. Razvoj zaštite elektroničke pošte 3
2.2.1. Diffie-Hellmanov algoritam 5
2.2.2. RSA algoritam 6
2.2.3. SHA 9
2.2.4. DSA algoritam 13
2.2.5. Simetrični kriptografski algoritmi 15
3. METODE ZA ENKRIPCIJU ELEKTRONIČKE POŠTE 18
3.1. Pretty Good Privacy (PGP) 23
3.2. S/MIME 26
3.2.1. S/MIME certifikati 29
4. IMPLEMENTACIJA METODA ZA ENKRIPCIJU ELEKTRONIČKE POŠTE 31
4.1. Gpg4win 31
4.2. Mozzila Thunderbird 38
4.3. Mailvelope 47
4.4. Protonmail 52
4.5. Usporedba implementiranih metoda za enkripciju 56
5. ZAKLJUČAK 58
LITERATURA 59
SAŽETAK 60
ŽIVOTOPIS 61

1. UVOD

U modernom svijetu postoji sve veća potreba za enkripcijom podataka. Činjenica je da se većina današnjih informacija prenosi elektroničkim putem. Neke od tih informacija mogu biti od izrazite važnosti za pojedinca, društvo, državu pa čak i cijeli svijet. Podatci koji se prenose elektroničkim putem mogu imati veliku novčanu ili stratešku vrijednost što će navesti određene pojedince ili grupacije da na sve moguće načine pokušaju doći do tih podataka. Upravo iz tog razloga, potreba za enkripcijom (šifriranjem) prijenosa podataka je od velike važnosti kako bi se zaštitila opća sigurnost te kako određene važne informacije nebi dospjele u posjed osoba koje se njima žele okoristiti na štetu čitavog društva. Jedan od najčešćih modernih načina prijenosa podataka je elektronička pošta. Ona se koristi u privatne i poslovne svrhe te je jedna od najčešćih meta za krađu podataka iz razloga što većina poslane elektroničke pošte nije enkriptirana, a samim time niti zaštićena. Stručnjaci su s vremenom razvili određene metode kojima se sadržaj elektroničke pošte može zaštiti na način da taj informacije poslane elektroničkom poštom mogu vidjeti samo pošiljatelj i primatelj iste. Standardi na kojima se danas većinom temelji zaštita elektroničke pošte su PGP i S/MIME.

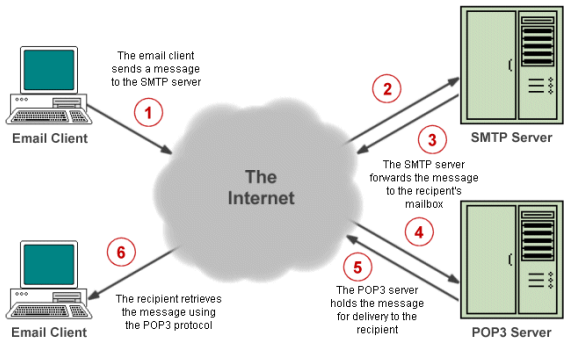
Ovaj diplomski rad će se bazirati na implementaciju i usporedbu navedenih metoda kroz različite primjere. U početnom poglavlju rada biti će ukratko opisana povijest razvoja i struktura elektroničke pošte. Također, nešto detaljnije će biti opisana povijest razvoja enkripcije elektroničke pošte počevši od samih potreba za zaštitom podataka poslanih elektroničkom poštom. U idućem poglavlju će biti detaljno opisani PGP i S/MIME standardi zajedno sa svojom strukturom te algoritmima koje koriste. Algoritmi koje ove 2 metode koriste imaju različitu ulogu pri enkripciji pa će tako većina njih biti detaljno opisana. Zajedno s ovim poglavljem, iduće također čini glavni dio ovog diplomskog rada jer će u njemu biti implementirane metode koristeći različite aplikacije i dodatke za enkripciju elektroničke pošte.

2. ZAŠTITA ELEKTRONIČKE POŠTE

2.1. Elektronička pošta

Elektronička pošta počela se koristiti tijekom 1960ih godina uz ograničenu uporabu dok je sredinom 70-ih godina prošlog stoljeća doživjela svoju uporabu koja je poznata i danas. U početku, uvjet slanja i primanja pošte je bio da su i primatelj i pošiljatelj spojeni na internet inače se komunikacija nije mogla odvijati. Današnja elektronička pošta se bazira na pohrani poruka te je dovoljno da je u trenutku slanja pošiljatelj spojen na internet kako bi se poruka uspješno pohranila na server, a primatelj da je spojen u trenutku kada želi preuzeti poruku pohranjenu na serveru.

Elektronička pošta se sastoji od 2 glavna dijela, a to su zaglavlje poruke te tijelo poruke (sadržaj poruke). Svaka poruka ima samo jedno zaglavlje koje je podijeljeno na polja. Zaglavlje poruke se sastoji od više različitih polja u kojima se nalaze informacije o pošiljatelju, primatelju, datumu, naslovu poruke te drugih informacija o elektroničkoj pošti. Vrijednosti unutar polja se nalaze u 7-bitnom ASCII formatu ili u MIME formatu ukoliko je pošta enkriptirana. Unutar tijela poruke se nalazi sadržaj poruke koji pošiljatelj želi poslati primatelju. Prilikom slanja i primanja poruka se koriste protokoli SMTP, POP3 te IMAP protokol. SMTP (Simple Mail Transfer Protocol) je protokol koji služi za prijenos elektroničke pošte od cilja do odredišta te se koristi od 80-ih godina prošlog stoljeća. POP3 (Post Office Protocol verzija 3) i IMAP (Internet Message Access Protocol) su protokoli koji se koristi za preuzimanje elektroničke pošte. Na slici 2.1. će biti prikazan način funkcioniranja gore navedenih protokola. [1]



Sl. 2.1. *Funkcioniranje protokola kod elektroničke pošte*

Na slici 2.1. je vidljivo kako korisnik sa svoga računala šalje elektroničku poštu putem interneta na SMTP server. SMTP zatim tu poruku prosljeđuje u poštanski sandučić od primatelja.

Primatelj će zatim pomoću POP3 (ili IMAP) protokola preuzeti poslanu poštu te uspješno završiti započetu komunikaciju. [1]

2.2. Razvoj zaštite elektroničke pošte

Pri samim početcima korištenja elektroničke pošte jako je malo bilo govora o privatnosti iste. U početku, korisnici prvih verzija elektroničke pošte su morali vjerovati svojim kolegama ili sistemskim administratorima o očuvanju podataka elektroničke pošte koju su slali. 1973. godine, Roger Shell je prvi puta upotrijebio enkripciju poruka i to na Multics operativnom sustavu, a princip enkripcije je bio prilično jednostavan. Korisnik je mogao odabrati sigurnosni ključ te ga unijeti u program za enkripciju te bi tako njegova poruka bila enkriptirana. Tu istu poruku bi mogao dekriptirati samo u slučaju ukoliko unese isti ključ kojim je poruka bila enkriptirana. Iako je Multics podržavao slanje poruka između dva različita korisnika putem elektroničke pošte, ovakav način enkripcije još uvijek nije bio implementiran za komunikaciju između više korisnika.[2]

1974. godine dolazi do prve enkripcije elektroničke pošte. Sam početak razvoja zaštite elektroničke pošte je vezan uz slanje strogo povjerljivih vojnih informacija. Operacijski sustav na kojem je prva enkripcija elektroničke pošte korištena je bio BBN TENEX, a upravo je zaposlenik Tenex-a Austin Henderson implementirao simetričnu enkripciju prilikom slanja pošte na sustavu za upravljanje porukama zvanom Hermes. Način funkcioniranja ove enkripcije je bio idući. Hermes je naveo korisnika koji šalje poruku da odabere ključ u text formatu te je sadržaj unesenog bio izmiješan kako bi se dobio ključ enkripcije K. S tim ključem, poruka je bila enkriptirana a dobiveni podaci su pretvoreni u ascii format koji se sastojao od blokova od 5 slova odvojenih razmacima. U ovakvom obliku poruka je poslana primatelju kojeg se također tražio ključ K kako bi mogao dešifrirati poruku. Ovakav način enkripcije se naziva simetričnom enkripcijom iz razloga što pošiljatelj i primatelj koriste isti ključ za enkripciju. [2]

1976. godine se prvi puta upotrebljava pojam javnog ključa. Pomoću javnog ključa su se izbjegle situacije u kojima korisnici koji međusobno šalju poruke moraju tražiti sigurne načine kako bi se dogovorili koji ključ će koristiti za enkripciju i dekripciju poruka iz razloga što je do javnog ključa korišten isti ključ za oba procesa. Pomoću javnog ključa, korisnik daje do znanja osobama koje mu žele poslati poruku da tu istu poruku enkriptiraju tim javnim ključem. Dakle, sve osobe

koje žele osobi A poslati poruku će koristiti javni ključ od te osobe A za enkripciju poruke koju joj žele poslati. Za razliku od simetrične enkripcije, ovdje je riječ o asimetričnoj enkripciji iz razloga što će ključ koji će osoba A koristiti za dekripciju primljenih poruka biti različit od onog ključa (javni ključ) koji je korišten za enkripciju. Prednost asimetrične enkripcije je ta što samo osoba koja prima poruku može znati koji koji ključ za dekripciju odgovara pojedinom ključu za enkripciju te na taj način samo ona može pročitati poruku. U tablici 2.1 će biti opisana razlika između simetrične i asimetrične kriptografije.[2] [3]

Tab 2.1. *Usporedba kriptosustava*

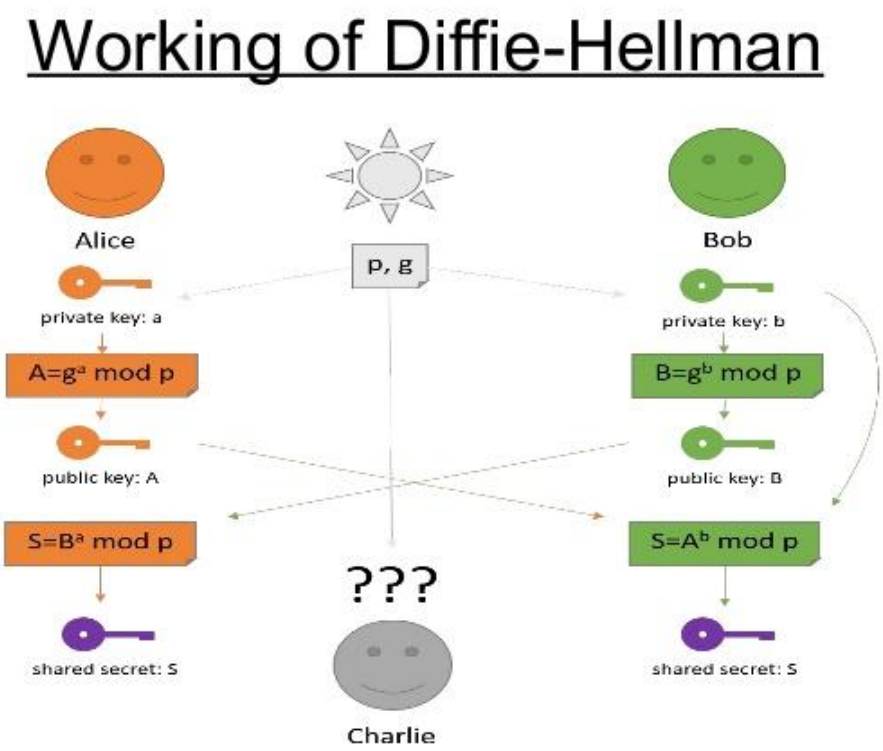
Simetrični kriptosustavi	**Asimetrični kriptosustavi**
Potrebno za funkcionalnost	*Potrebno za funkcionalnost*
Isti algoritam s istim ključem se koristi i za enkripciju i za dekripciju	Za enkripciju i dekripciju se koriste različita 2 ključa i algoritma
Pošiljatelj i primatelj moraju znati o kojem algoritmu i ključu se radi	Pošiljatelj i primatelj moraju imati jedan od 2 ključa koja se koriste (ne oba)
Potrebno za sigurnost	*Potrebno za sigurnost*
Ključ mora biti tajan	Jedan od dva ključa mora biti tajan
Ukoliko je ključ tajan, vjerojatnost da se poruka dešifrira mora biti minimalna	Ukoliko je jedan od ključeva tajan, vjerojatnost da se poruka dešifrira mora biti minimalna
Poznavanje algoritma ne smije biti dostatno za dešifriranje poruke	Poznavanje algoritma te jednog od ključeva ne smije biti dostatno za dešifriranje poruke

U tablici se vidi kako je s asimetričnim kriptosustavima postignuta veća razina sigurnosti iz razloga što se za enkripciju i dekripciju ne koriste isti ključevi i algoritmi. Također, riješen je problem pronalaženja sigurnog načina razmjene ključa između pošiljatelja i primatelja zbog toga što se više ne koristi više isti ključ, a pošiljatelj i primatelj moraju znati samo jedan od dva ključa dok drugi ne moraju znati.

Razvojem javnog ključa počeo je i razvoj različitih algoritama za enkripciju. Jedan od takvih algoritama je Diffie-Helmanov algoritam. U nastavku će biti dan pregled nekoliko algoritama koji se koriste i u metodama za enkripciju elektroničke pošte.

2.2.1. **Diffie-Hellmanov algoritam**

1976. godine znanstvenici Whitfield Diffie i Martin Hellman osmislili su metodu koja je značajno unaprijedila razvoj kriptografije. Iz tog razloga, ova 2 znanstvenika se smatraju začetnicima kriptografije javnog ključa. Na slici 2.2 je prikazana funkcionalnost ovog algoritma. [4]



Sl. 2.2. *Diffie-Hellmanov algoritam*

Dakle, na slici je vidljiva funkcionalnost algoritma prilikom razmjene ključa za enkripciju između 2 osobe. Koraci u algoritmu su [4]:

1. Alice i Bob odabiru brojeve p i g. Inače se koriste jako veliki brojevi, u ovome primjeru za potrebe izračuna su korišteni p = 23 i g = 5
2. Alice odabire svoj privatni ključ (broj) a = 6, a Bobu šalje A izračunatu kao:

$$A = g^a \bmod p$$

$$A = 5^6 \bmod 23$$

$$A = 8$$

3. Bob odabire svoj privatni ključ (broj) b = 15, a Alice šalje B izračunatu kao:

$$B = g^b \bmod p$$

$$B = 5^{15} \bmod 23$$

$$B = 19$$

4. Alice računa $S = B^a \bmod p$

$$S = 19^6 \bmod 23 = 2$$

5. Bob računa $S = A^b \bmod p$

$$S = 8^{15} \bmod 23 = 2$$

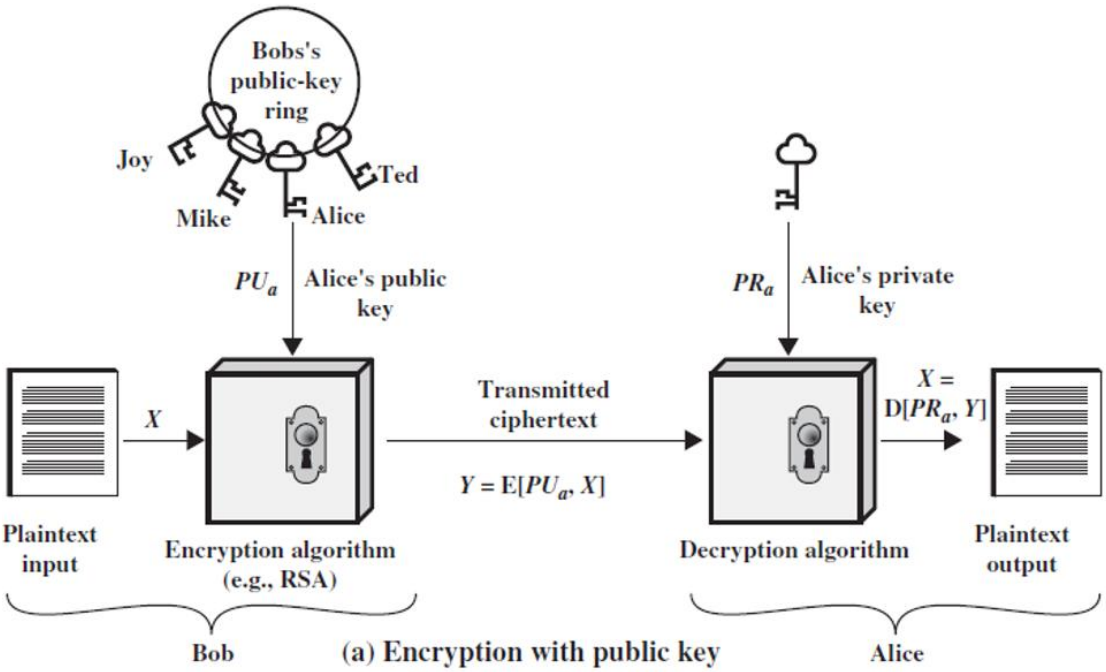
6. Tajni ključ od Alice i Boba je 2

Kod ovog algoritma je vidljivo kako je on temeljen na asimetričnom kriptosustavu za razmjenu poruka. U ovome slučaju, poruka je gore dobiveni tajni ključ za kriptiranje. Ovaj algoritam se u većini slučajeva koristi za sigurnu razmjenu ključa za enkripciju što je i prethodno prikazano. Pošiljatelj i primatelj se ne moraju poznavati, a kanal za razmjenu ključa ne mora biti siguran.

2.2.2. **RSA algoritam**

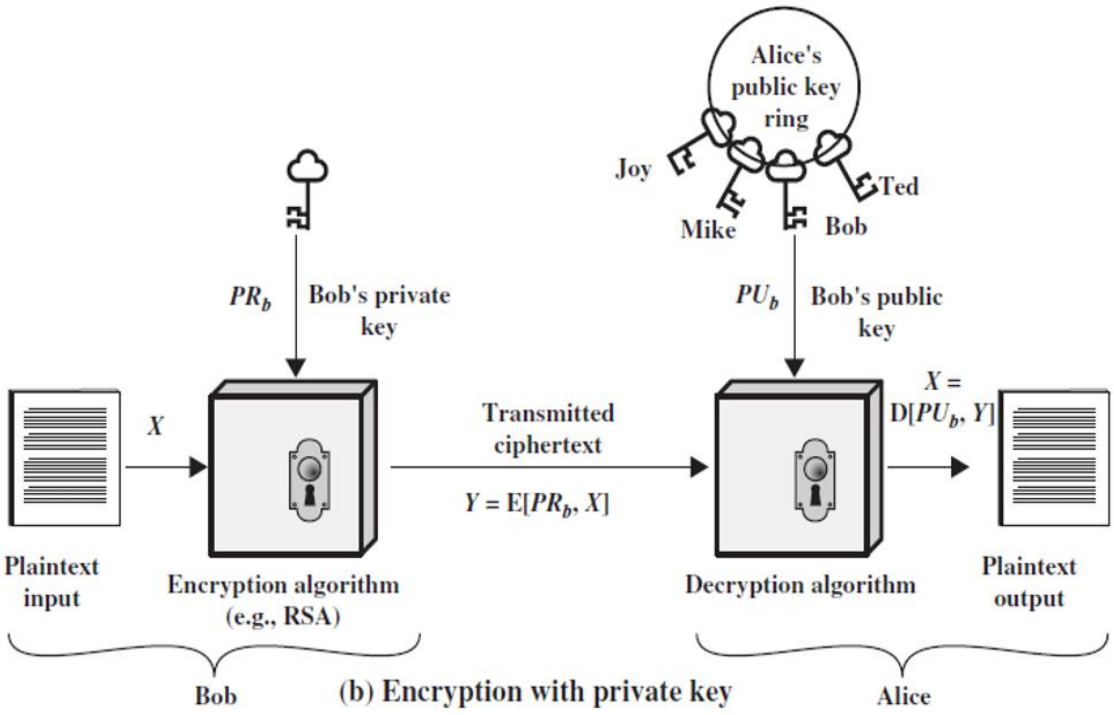
1977. godine je osmišljen prvi i najčešće korišteni kriptosustav koji koristi javni ključ. Zaslužni za to su znastvenici Ronald Rivest, Adi Shamir i Leonard Adleman prema čijim prezimenima je kriptosustav nazvan kao RSA algoritam. Kod ovog algoritma se koriste 2 ključa, a to su javni ključ koji može biti dostupan svima te privatni ključ koji mora ostati tajan. Za enkripciju se može

koristiti i javni i privatni ključ, ali bitno je da se za dekripciju koristi ključ koji nije korišten za enkripciju iste poruke. Na slici 2.3 je prikazana enkripcija javnim ključem, dok je na slici 2.4 prikazana enkripcija privatnim ključem. [3][4]



Sl. 2.3. *Enkripcija javnim ključem*

Na slici je vidljivo kako Bob šalje poruku prema Alice. Pri enkripciji s primjerica RSA algoritmom Bob zna više javnih ključeva, ali će koristiti javni ključ od Alice s obzirom da njoj šalje poruku. Na drugoj strani, Alice će koristiti svoj vlastiti privatni ključ kako bi dekriptirala poruku poslanu od strane Boba. Na slici 2.4 će biti prikazan slučaj gdje će se za enkripciju koristiti privatni ključ.



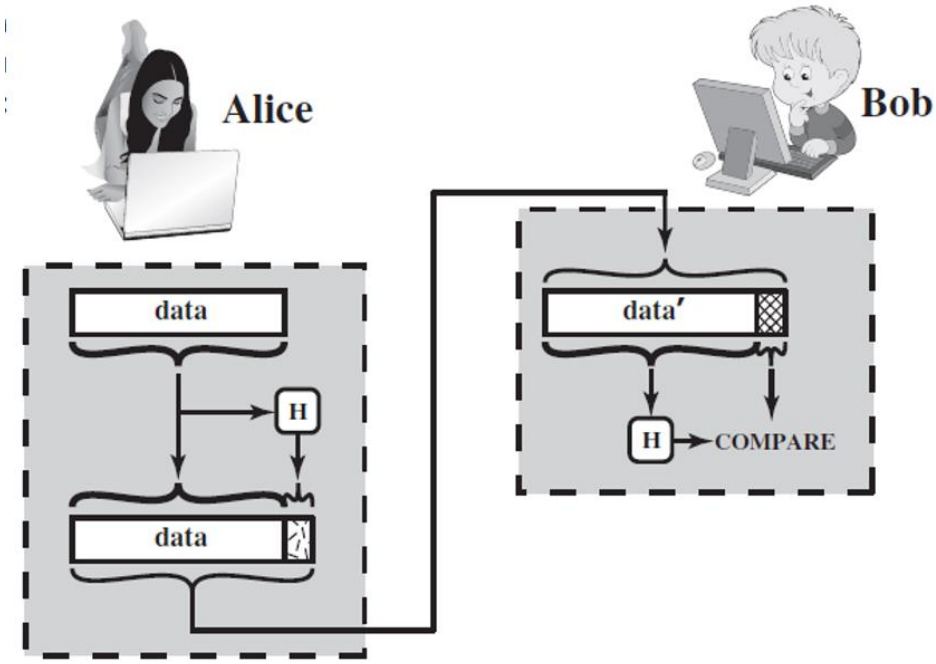
Sl. 2.4. *Enkripcija privatnim ključem*

Koraci u RSA algoritmu su [4]:

1. Izabiru se dva prosta broja p i q, primerice p = 3 i q = 11
2. n = p * q = 3 * 11 = 33
3. Računa se Totientova funkcija φ(n) = (p – 1) * (q – 1) = 2 * 10 = 20
4. Izabire se e u intervalu 1 < e < φ(n) gdje je e relativno prost s φ(n), pr. e = 7
5. Izračuna se d iz (d * e) % φ(n) = 1, pr. d = 3 [(3 * 7) % 20 = 1]
6. Javni ključ je tada (e,n) => (7,33)
   Privatni ključ je tada (d,n) => (3,33)
7. Ukoliko se poruka koju se želi poslati označi kao m, enkripcijska funkcija je tada
   $$c(m) = m^e \bmod \varphi(n)$$
8. Neka je m = 2 pa će se tako prema gornjoj formuli dobiti c(m) = 29
9. Za šifrirani tekst c dekripcijska funkcija je $m(c) = c^d \bmod \varphi(n)$ pa će se tako iz te funkcije za šifrirani tekst c(m) = 29 dekripcijom dobiti početna poruka m = 2

2.2.3. **SHA**

SHA (Secure Hash Algorithm) je jedna od kriptografskih hash funkcija. Kriptografske hash funkcije su funkcije H koje za ulazni podatkovni blok M koji je varijabilne duljine izračunava „hash“ vrijednost h=H(M) koja je fiksne duljine. Podatkovni blok M može biti primjerice nekakva poruka ili datoteka. Hash funkcije zaštićuju integritet podataka iz razloga što one za velik broj ulaznih podataka daju izlaze koji su ravnomjerno distribuirani i prividno slučajni. Kriptografske hash funkcije funkcioniraju na način da primatelj provodi izračunavanje hash vrijednosti te uspoređuje rezultat s primljenom hash vrijednosti. Na taj način se može autentificirati poruka te provjeriti njen integritet. Na slici 2.5 je prikazan način funkcioniranja kriptografskih hash funkcija.



Sl. 2.5. *Kriptografske hash funkcije*

Dakle, na slici 2.5 je vidljivo kako Alice šalje poruku prema Bobu. Bob prima hash vrijednost H te nakon primljene poruke sam računa hash vrijednost H. Ukoliko primljena i izračunata H vrijednost nisu jednake to će značiti da je poruka izmijenjena prilikom prijenosa.

Jedna od najraširenijih kriptografskih hash funkcija je SHA. Postoji više razvijenih verzija ovoga algoritma koje se razlikuju po veličini ulazne poruke, duljini sažete poruke na izlazu, veličini blokova po kojima se ulazna poruka procesuira te veličini riječi. Prikaz razlike između određenih verzija SHA algoritma je prikazan u tablici 2.2. [3]

Tab. 2.2. *Verzija SHA algoritma*

	1995.	2008.	2002.		
	SHA-1	**SHA-224**	**SHA-256**	**SHA-384**	**SHA-512**
Message Digest Size	160	224	256	384	512
Message Size	$< 2^{64}$	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
Block Size	512	512	512	1024	1024
Word Size	32	32	32	64	64
Number of Steps	80	64	64	80	80

Tako će primjerice SHA-1 algoritam kao ulaz uzimati poruku maksimalne duljine 2^{64} bita, a kao izlaz će dati sažetak poruke duljine 160 bita. Ulazna poruka će se procesuirati u blokovima veličine od po 512 bitova. SHA algoritmi su izrazito dobri pri zaštiti podataka jer samo mala promjena ulazne poruke uzrokuje veliku promjenu. U nastavku će biti dan primjer za 2 poruke u kojima će biti izmijenjeno samo jedno slovo.

SHA-1 ulazna poruka „The quick brown fox jumps over the lazy dog“

SHA-1 izlazna poruka u heksadec. obliku : 2fd4e1c67a2d28fced849ee1bb76e7391b93eb12

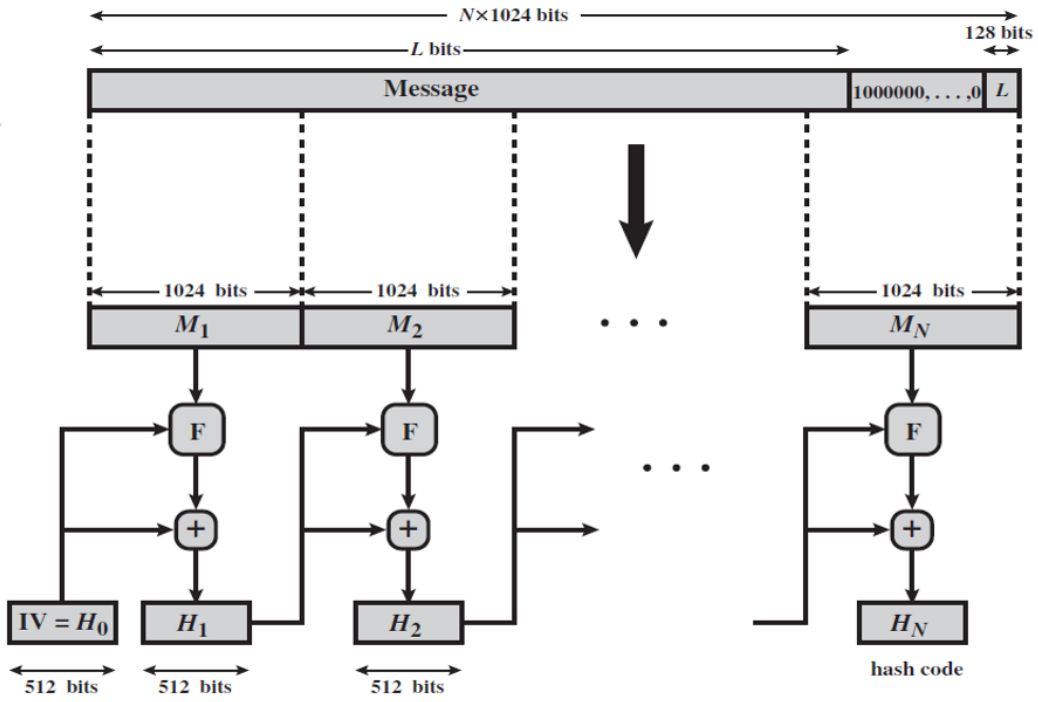
Ukoliko se u ulaznoj poruci izmijeni samo jedno slovo dobije se iduće:

SHA-1 ulazna poruka „The quick brown fox jumps over the lazy dog“

SHA-1 izlazna poruka u heksadec. obliku : de9f2c7fd25e1b3afad3e85a0bd17d9b100db4b3

Iz navedenog je vidljivo kako samo mala promjena u ulaznoj poruci čini veliku promjenu na izlaznoj strani što je idealno za dobru zaštitu podataka.

Način rada SHA algoritma će biti prikazan na primjeru SHA-512 gdje se kao sažetak poruke daje izlaz od 512 bita za maksimalnu duljine ulazne poruke od 2^{128} bita , a poruka se procesuira u blokovima od 1024 bita. Shema procesa je prikazana na slici 2.6. [3]



Sl. 2.6. *Kreiranje sažetka poruke algoritmom SHA-512*

Koraci [3] u kreiranju sažetka poruke su:

1. Dodavanje bitova za nadopunu

Poruci se dodaje duljina nadopune P $\equiv$ 896 (mod 1024). U zagradi se nalazi 1024 iz razloga što se ulazna poruka procesuira u blokovima od 1024 bita. Prvi bit te nadopune će biti 1 dok će ostali bitovi biti 0.

2. Dodavanje duljine poruke

Poruci se dodaje blok od 128 bita u kojima je zapisana duljina originalne poruke bez nadopune (kao 128 bitni cijeli broj)

1. Inicijalizacija hash međuspremnika

Koristi se međuspremnik kojeg se može prikazati kao 8 64-bitnih registara (a,b,c,d,e,f,g,h) u kojeg se spremaju međurezultati i krajnji rezultati hash funkcija. Inicijalni sadržaj tih registara je prikazan u heksadecimalnom obliku:

a = 6A09E667F3BCC908 e = 510E527FADE682D1

b = BB67AE8584CAA73B f = 9B05688C2B3E6C1F

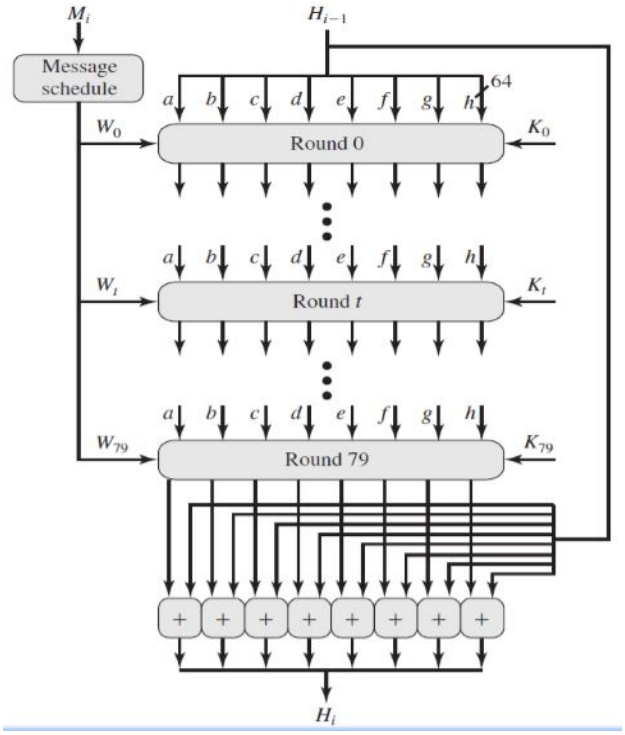
c = 3C6EF372FE94F82B g = 1F83D9ABFB41BD6B

d = A54FF53A5F1D36F1 h = 5BE0CD19137E2179

Ove vrijednosti su dobivene uzimanjem prvih 64 bita decimalnih vrijednosti kvadratnih korijena prvih 8 prostih brojeva. Najznačajniji bit registra je krajnji lijevi bit.

2. Procesiranje poruke u 1024-bitnim blokovima

Glavni dio algoritma je modul koji se sastoji od 80 rundi („number of steps“) te je vidljiv na slici 2.7. [3]



Slika 2.7. *Procesiranje poruke u 80 rundi*

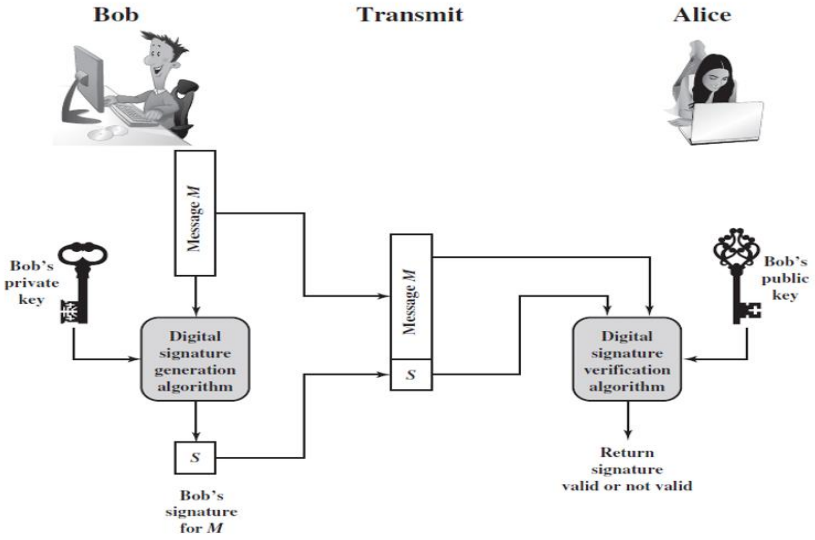
U svakoj od 80 rundi se koristi adtivina konstanta Kt gdje svaka konstanta predstavlja prva 64 bita decimalnog dijela kubnih korijena prvih 80 prostih brojeva. Izlaz iz posljednje runde se zbraja se (modulo 2^{64}) sa ulazom u prvu rundu H_{i-1} kako bi se dobila vrijednost H_i

3. Izlaz

Nakon što se završi s procesiranjem svih 1024-bitnih blokova, izlaz iz posljednjeg bloka predstavlja 512-bitni sažetak poruke

2.2.4. DSA algoritam

Digitalni potpis je zapravo asimetrični kriptosustav temeljen na javnom ključu kojeg su prvog opisali već gore spomenuti Diffie i Hellman 1976. godine . Pomoću njega se može utvrditi autentičnost i integritet neke primljene poruke ili datoteke. Poruka će biti autentična ako se sa sigurnosti može reći tko je njen autor, a ukoliko poruka nije mijenjana putem onda je očuvan i njen integritet. Glavna ideja digitalnog potpisa je da se uz pomoć izvorne poruke i tajnog ključa pošiljatelja generira digitalni potpis tog istog pošiljatelja. Na drugoj strani, primatelj će provjeriti autentičnost potpisa pošiljatelja nakon što dobije izvornu poruku te ukoliko zna javni ključ pošiljatelja. Na slici 2.8. je prikazan način rada digitalnog potpisa. [5]

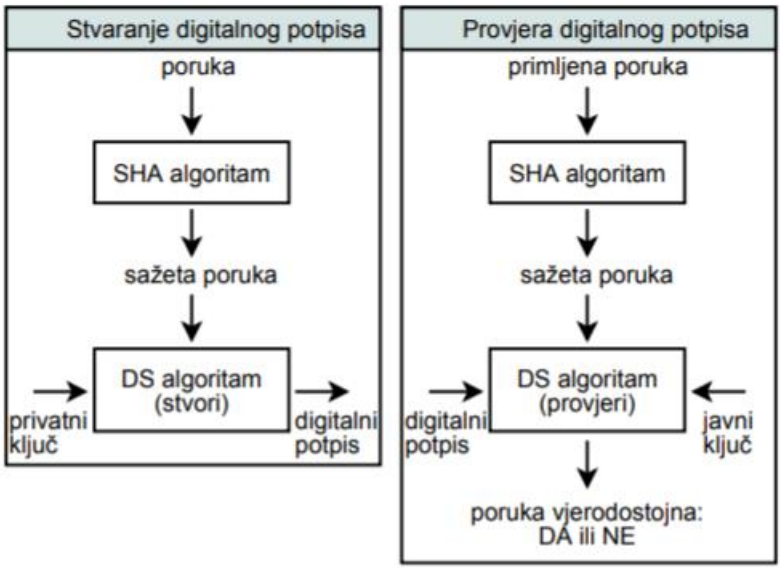


Slika 2.8. *Način rada digitalnog potpisa*

Digitalno potpisivanje se izvodi u sljedeća 3 koraka:

1. Generiranje javnog i privatnog ključa za potpisivanje poruke s pripadajućim algoritmom
2. Potpisivanje poruke s pripadajućim algoritmom
3. Provjera potpisa s pripadajućim algoritmom

Najpoznatiji algoritam korišten za digitalni potpis je DSA algoritam. Ovaj algoritam za digitalno potpisivanje koristi gore navedeni SHA algoritam. Princip rada je prikazan na slici 2.9. [5]



Sl. 2.9. *DSA algoritam*

DSA algoritam se sastoji od 3 koraka: [5]

a) **stvaranje ključeva**

1. odabir 160-bitnog prostog broja q

2. odabir L-bitnog prostog broja p tako da vrijedi: $p = qz + 1$ za neki cijeli broj z, $512 \leq L \leq 1024$, L djeljivo s 64

3. odabir h takvog da vrijedi: $1 < h < p - 1$, $g = hz \bmod p > 1$

4. stvaranje nasumičnog broja x, tako da vrijedi: $0 < x < q$

5. izračun $y = g^x \bmod p$

6. (p, q, g, y) je javni ključ, a x je privatni ključ

b) **potpisivanja poruke**

1. stvaranje nasumičnog broja k, tako da vrijedi: $0 < k < q$

2. izračun r = (g k mod p) mod q

3. izračun s = (k —1 (SHA-1(m) + xr)) mod q, gdje je SHA-1(m) tzv. hash funkcija primijenjena na poruci m

4. ponoviti postupak ako je r = 0 ili s = 0 5. potpis je (r, s)

c) **provjera vjerodostojnosti potpisa**

1. ako 0 < r < q ili 0 < s < q nije zadovoljeno potpis se smatra neispravnim

2. izračun w = s -1 mod q

3. izračun u1 = (SHA-1(m)w) mod q

4. izračun u2 = (rw) mod q

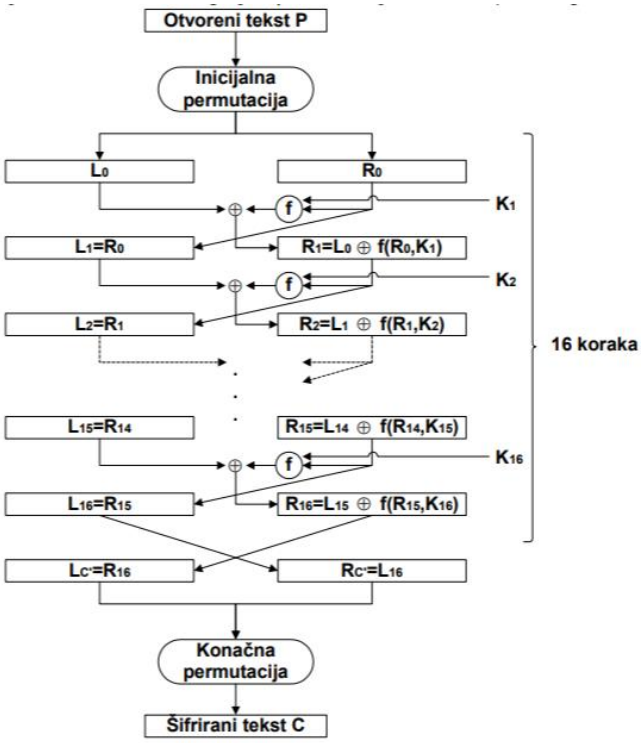
5. izračun v = ((g u1 y u2) mod p) mod q 6. potpis je valjan ako vrijedi v = r

2.2.5. Simetrični kriptografski algoritmi

U metodama za enkripciju elektroničke pošte osim asimetričnih algoritama, hash algoritama i algoritama za digitalni potpis, koriste se i simetrični algoritmi. U nastavku će biti detaljnije opisani algoritmi najznačajniji za metode koje će kasnije biti spominjane a to su 3DES i IDEA.

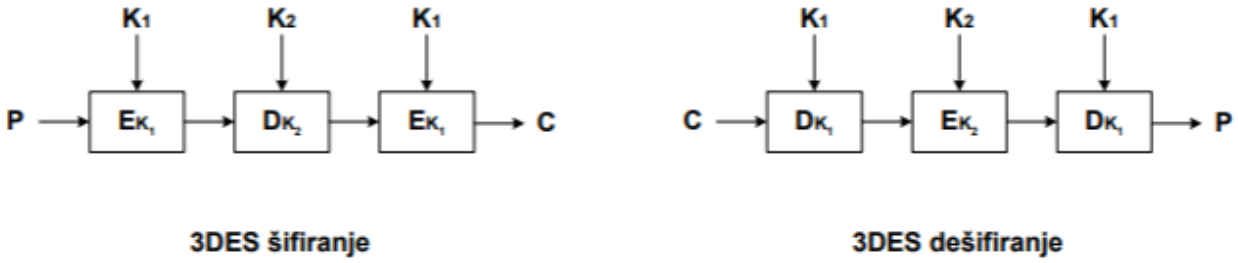
Što se tiče 3DES algoritma, on je inačica DES algoritma. DES (Data Encryption Standard) je razvijen tokom 70-ih godina u IBM-u. Proces u DES algoritmu je prikazan na slici 2.10. [6]

Postupak kod DES algoritma je idući. Poruka se šifrira u 64-bitnim blokovima te na izlazu daje 64-bitni šifrirani tekst. Ključ za šifriranje je duljine 56 bita, ali može biti i duljine 64 bita pri čemu će se svaki 8. bit zanemariti te koristiti za npr. provjeru pariteta. Kao što je vidljivo na slici 2.10, postupak počinje inicijalnom permutacijom, a završava sa konačnom permutacijom koja je zapravo inverzija inicijalne permutacije. Između inicijalne i konačne permutacije se nalazi 16 koraka u kojima se niz znakova poruke dijeli na lijevu i desnu 32-bitnu polovicu gdje desna polovica postaje lijeva polovica, a na lijevoj polovici se provode operacije koje su u svakom koraku parametrizirane drugim 48-bitnim podključem (ukupno 16 različitih podključeva). Ti podključevi se izvode iz osnovnog ključa. [6]



Sl. 2.10. *DES algoritam*

Zbog potražnje za većom zaštitom podataka, razvijen je algoritam 3DES koji koristi 2 ili 3 ključa za šifriranje. Ovaj algoritam se još naziva i TDEA što bi značilo „Triple Data Encryption Algorithm“. Pri ovome algoritmu će se koristiti 3 koraka za enkripciju: enkripcija, dekripcija pa onda opet enkripcija. Dekripcija će koristiti isto 3 koraka, ali će oni biti drugačijeg redoslijeda i to kao dekripcija, enkripcija te onda opet dekripcija. Shema algoritma je prikazana na slici 2.11. [6]

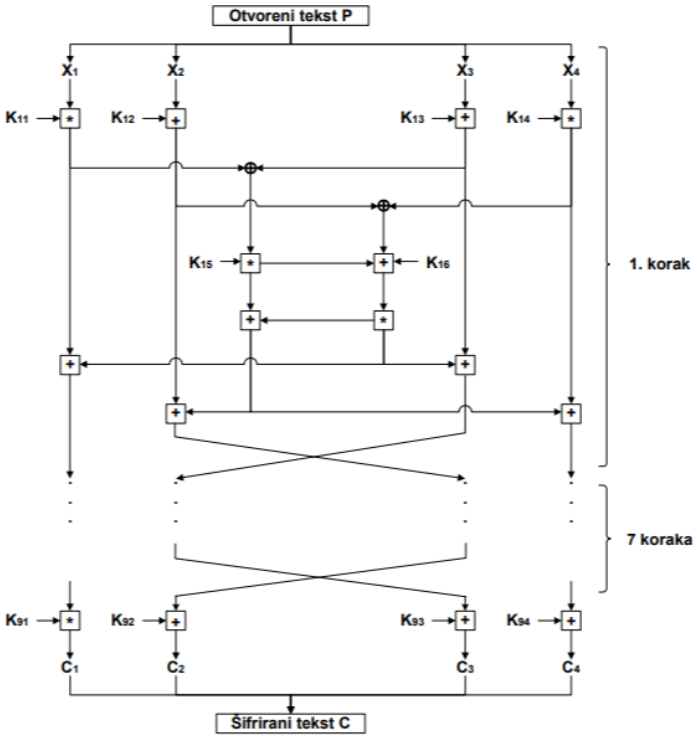


Slika 2.11. *3DES algoritam*

U ovome slučaju 3DES algoritam koristi 2 56-bitna ključa što čini 112-bitni ključ za šifriranje.

Drugi algoritam značajan za metode enkripcije elektroničke pošte koji će se koristiti je IDEA (International Data Encryption Algorithm). Ovaj algoritam je razvijen 1991. godine, a u početku

je služio kao zamjena za gore opisani DES algoritam. Kod njega se poruka šifrira u 64-bitnim blokovima, a ključ za šifriranje je duljine 128 bita. Kod ovog algoritma se koriste 3 algebarske operacije, a to su XOR, zbrajanje modulo 2^{16} te množenje modulo 2^{16+1}. Ovaj algoritam se sastoji od 8 funkcijski identičnih koraka. Ulaz u svakom koraku su četiri 16-bitna bloka generirani u prethodnom koraku. Rezultati funkcije u svakom koraku su isto tako četiri 16-bitna izlazna bloka. Na početku šifriranja poruka se podijeli na četiri 16-bitna podbloka te se na svim tim podblokovima u svakom koraku izvode operacije XOR, zatim modulo zbrajanje te naposlijetku modulo množenje međusobno te sa 6 16-bitnih podključeva. . Između svakog koraka drugi i treći podblok se zamijenjuju. Na samome kraju postupka, 4 podbloka se kombiniraju sa 4 podključa što daje konačni rezultat. Opisani postupak je vidljiv na slici 2.12. gdje će se vidjeti detaljno opisani prvi korak sa svim matematičkim operacijama koje se izvode. Idućih 7 koraka je jednako pa nema potrebe za detaljnim prikazivanjem. [7]



Sl. 2.12. *Shema IDEA algoritma*

3. METODE ZA ENKRIPCIJU ELEKTRONIČKE POŠTE

Između 1984. i 1992. godine grupa pod nazivom „Privacy and Security Research Group“ (PSRG) se bavila načinima na koje elektroničku poštu učiniti sigurnijom. Njihovo istraživanje o navedenoj temi je nazvano „Privacy Enhanced Mail“ (PEM) što bi u prijevodu označavalo elektroničku poštu s poboljšanom sigurnosti. Glavni tehnički problemi koje je ova grupa htjela riješiti su bili:[2][8]

- Prenošenje enkriptirane poruke elektroničkom poštom pomoću SMTP protokola bez da se mijenjaju serveri koji proslijeđuju poruke između web stranica
- Korištenje metoda javnog ključa kako bi se osiguralo da samo primatelj kojemu je namjerena poruka može istu pročitati
- Korištenje metoda javnog ključa kako bi se osigurala autentičnost elektroničke pošte
- Osigurati da primatelj može detektirati bilo kakvu promjenu između primljene i originalne poruke
- Olakšati prijenos javnih ključeva

Iako se sigurnost elektroničke pošte s vremenom sve više razvijala, u kasnim 1980-im godinama software inžinjer i politički aktivist Phil Zimmerman je odlučio još više ubrzati enkripciju elektroničke pošte. Naime, on je izradio vlastitu aplikaciju prema IETF i PKI („Public Key Infrastructure“) standardima te ju nazvao „Pretty Good Privacy“ (PGP). Ovo je jedna od glavnih metoda za enkripciju elektroničke pošte te će biti detaljnije objašnjena i implementirana u nastavku. Osim PGP-a, metoda koja će biti detaljno opisana je S/MIME koja je uz PGP najkorišteniji standard u zaštiti elektroničke pošte. 2 metode na kojima se temelje ova 2 standarda su metoda digitalnog potpisa i metoda enkripcije poruke koristeći pritom gore opisane algoritme. Digitalni potpis će osigurati: [2][8]

- Autentifikaciju – kod elektroničke pošte koja koristi SMTP protokol ne postoji autentifikacija tako da je digitalnim potpisom osigurana autentičnost, tj. tko je stvarni pošiljatelj poruke
- Jedinstvenost – kao i kod potpisa na papiru, svaki korisnik posjeduje svoj jedinstveni digitalni potpis čime se jasno može utvrditi tko je poslao poruku

- Integritet podataka – u ovome slučaju digitalni potpis osigurava integritet poruke na način da one nije mijenjana tokom prijenosa te da je primljena u originalnom obliku

Pojednostavljeni prikaz digitalnog potpisivanje poruke elektroničke pošte je prikazan na slici 3.1 [8]

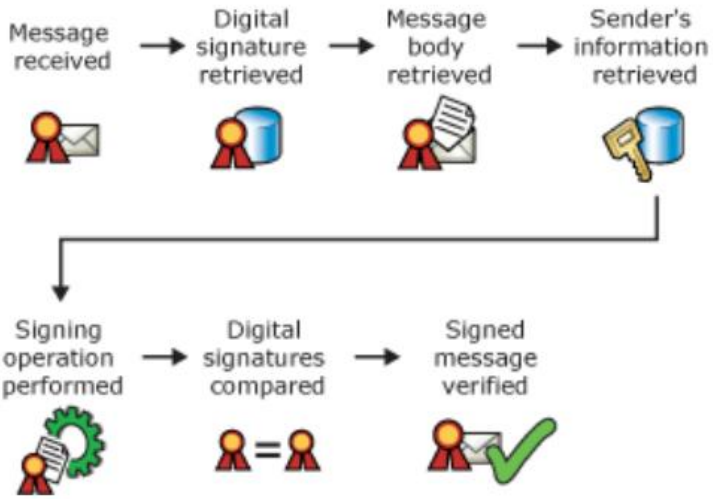


Sl. 3.1. *Digitalni potpis poruke elektroničke pošte*

Postupak digitalnog potpisivanja je sljedeći: [8]

1. Obuhvaćeno tijelo poruke
2. Primljena jedinstvena informacija o pošiljatelju
3. Postupak potpisivanja uz odgovarajući algoritam
4. Digitalni potpis dodijeljen poruci
5. Slanje potpisane poruke

Nakon što se potpisana poruka pošalje, na drugoj strani se vrši verifikacija te poruke što je prikazano na slici 3.2. [8]



Sl. 3.2. *Verifikacija digitalno potpisane poruke*

Postupak verifikacije je sljedeći: [8]

1. Primanje poruke
2. Dohvaćanje digitalnog potpisa poruke
3. Dohvaćanje tijela poruke
4. Dohvaćanje informacija koje identificiraju pošiljatelja
5. Izvršavanje operacije potpisivanja na poruci
6. Uspoređivanje digitalnih potpisa (primljenog i proizvedenog)
7. Ukoliko se u gornjem koraku potpisi poklapaju tada je poruka uspješno verificirana.

Digitalni potpis nije dovoljan za zaštitu elektroničke pošte iz razloga što sadržaj poruke može biti pročitan od strane drugih korisnika. Upravo iz tog razloga uz digitalni potpis se koristi i enkripcija poruka koja osigurava: [8]

- Povjerljivost – enkripcijom poruka se štiti sadržaj koji se nalazi unutar tih poruka na način da će taj sadržaj moći vidjeti samo osoba kojoj je poruka namijenjena
- Integritet podataka – kao i kod digitalnog potpisa, osigurat će se da poruka nije mijenjana tokom prijenosa te da je primljena u originalnom obliku

Enkripcija poruka elektroničke pošte je prikazana na slici 3.3. [8]

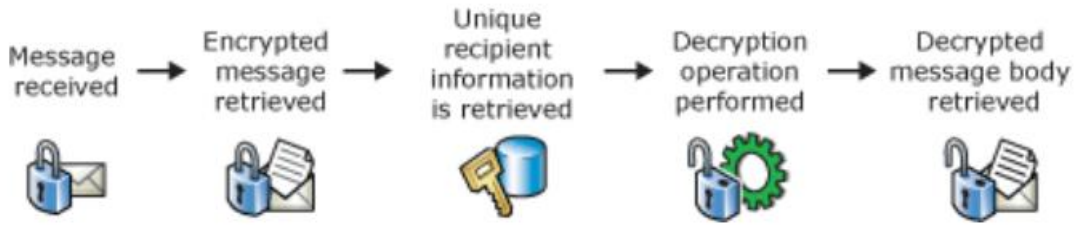


Sl. 3.3. *Enkripcija poruka elektroničke pošte*

Postupak enkripcije je sljedeći: [8]

1. Obuhvaćanje tijela poruke
2. Dohvaćanje informacija koje jedinstveno identificiraju pošiljatelja
3. Obavljanje enkripcije uz korištenje odgovarajućih algoritama
4. Zamjena originalne poruke s porukom enkriptiranom u prethodnom koraku
5. Slanje enkriptirane poruke

Dekripcija poruke elektroničke pošte je prikazana na slici 3.4 [8]

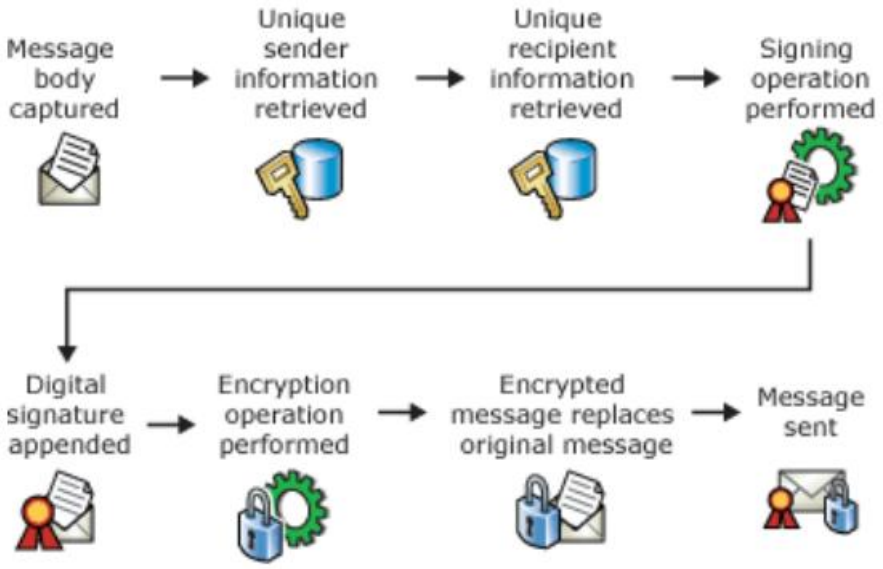


Sl. 3.4. *Dekripcija poruka elektroničke pošte*

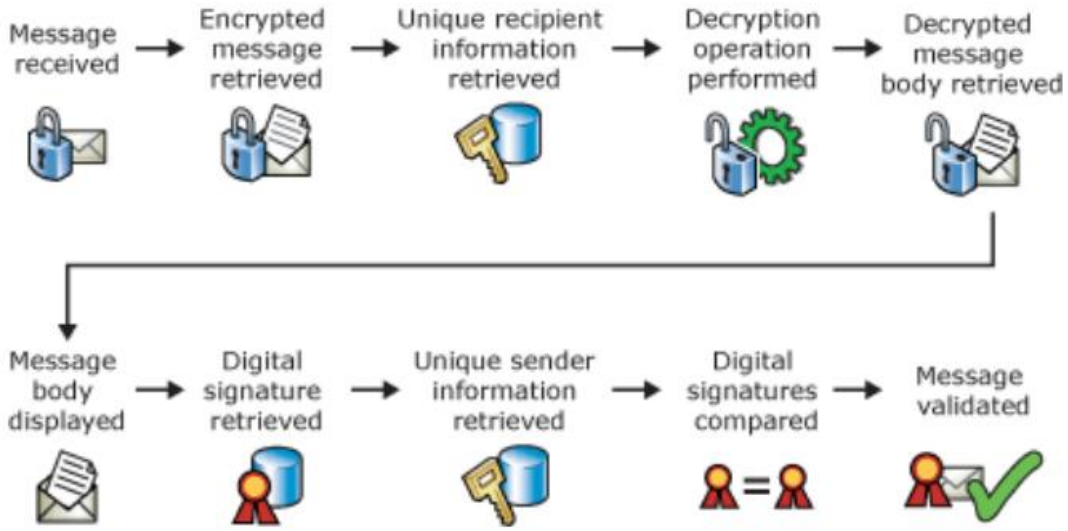
Postupak dekripcije je sljedeći: [8]

1. Primanje poruke
2. Dohvaćanje enkriptirane poruke
3. Dohvaćanje jedinstvene informacije o primatelju
4. Dekripcija poruke uz korištenje odgovarajućih algoritama
5. Dohvaćanje dekriptiranog tijela poruke

Pojednostavljene sheme istovremene enkripcije i digitalnog potpisa te dekripcije i verifikacije digitalnog potpisa su prikazane na slikama 3.5 i 3.6. [8]



Sl. 3.5. *Digitalni potpis i enkripcija elektroničke pošte*

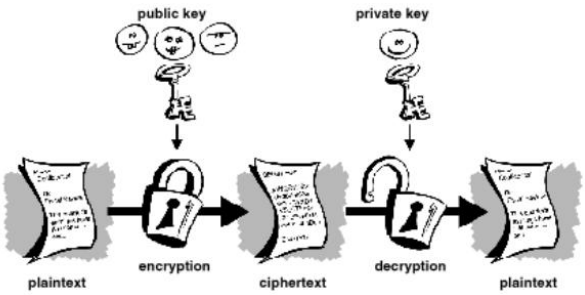


Sl. 3.6. *Dekripcija i verifikacija digitalnog potpisa*

Kao što je gore navedeno, 2 najpoznatija standarda za zaštitu elektroničke pošte su PGP i S/MIME standardi. Oba standarda se temelje na gore opisanim metodama enkripcije i digitalnog potpisa. U nastavku će biti dano detaljnije pojašnjenje oba standarda uz shematske prikaze i algoritme koje koriste za navedene metode.

3.1. Pretty Good Privacy (PGP)

PGP standard je standard koji se temelji na većem broju algoritama,a služi za enkripciju datoteka i poruka te za digitalni potpis. Za asimteričnu enkripciju koriste se RSA,DSS i Diffie Hellmanov algoritam, dok se za simetričnu enkripciju koriste algoritmi poput CAST-128, IDEA i 3DES. Metoda također koristi i SHA-1 za hash kodove. Pojednostavljeni prikaz načina funkcioniranja PGP standarda je prikazan na slici 3.7 [9]



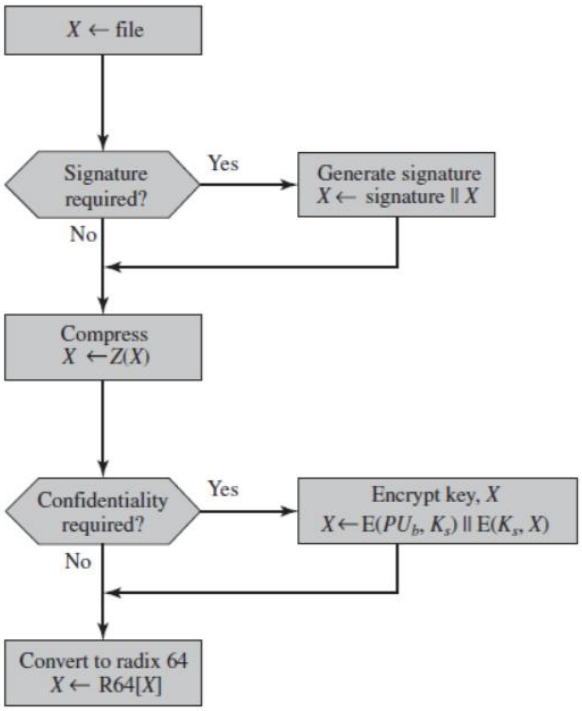
Sl. 3.7. *Funkcioniranje PGP standarda*

U tablici 3.1 će biti prikazane funkcije koje ovaj standard pruža uz pripadajuće algoritme koji se koriste za pojedinu funkciju koju ovaj standard omogućava. [10]

Tab. 3.1. *Funkcije PGP standarda*

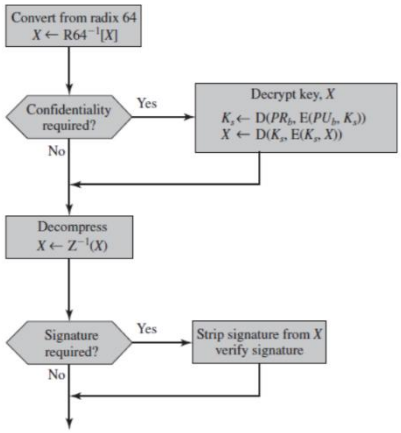
Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key and included with the message.
Compression	ZIP	A message may be compressed for storage or transmission using ZIP.
E-mail compatibility	Radix-64 conversion	To provide transparency for e-mail applications, an encrypted message may be converted to an ASCII string using radix-64 conversion.

Dakle, u tablici je vidljivo kako se PGP standard koristi za digitalni potpis čime će se osigurati autentičnost poruke. Pri digitalnom potpisu će se koristiti gore detaljno objašnjeni algoritmi DSS ili RSA u kombinaciji sa SHA algoritmom za hash kodove. Druga funkcija PGP metode je enkripcija poruke gdje će se koristiti kombinacija simetričnih algoritama (CAST, IDEA ili 3DES) te asimetričnih algoritama kao što su Diffie-Hellman ili RSA. Simetrični algoritmi će biti korišteni za enkripciju poruke, dok će asimetrični algoritmima biti enkriptiran ključ sesije. Iz razloga što metoda koristi i simetričnu i asimetričnu enkripciju, PGP možemo nazvati „hibridnim kriptosustavom“. Treća funkcija PGP metode je kompresija gdje će se koristiti ZIP algoritam. U ovome slučaju poruka će biti kompresirana za skladištenje ili prijenos pomoću ZIP algoritma. Kompresija poruka će utjecati na brzinu prijenosa i zauzimanja memorije, ali će i pojačati sigurnost enkriptirane poruke. Posljednja funkcija PGP metode je kompatibilnost elektroničke pošte. Ovdje će enkriptirana poruka biti konventirana u ASCII pomoću Radix-64 konverzije čime će se osigurati transparentnost za aplikacije elektroničke pošte. Na slici 3.8 će biti prikazan algoritam za slanje poruke PGP metodom. Kod slanja poruke će biti nuđena mogućnost kreiranja digitalnog potpisa te će nakon odrađenog, ili neodrađenog u slučaju da nema zahtjeva za digitalnim potpisom, biti kompresirana poruka. Nakon kompresije odrađivat će se dio zaslužan za pouzdanost, tj. enkripcija ključa. Na samome kraju datoteka će se konvertirati pomoću radix-64 konverzije u ASCII format. [9] [10]



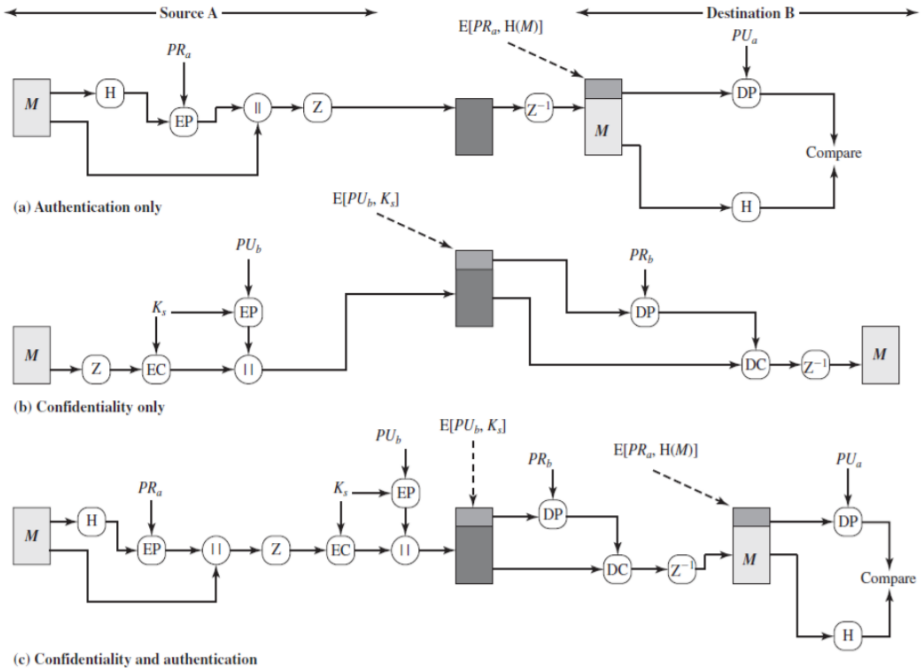
Sl. 3.8. *Algoritam slanja poruke PGP metodom*

Na slici 3.9 će biti prikazana situacija primanja poruke korištenjem PGP metode. Kod primanja poruke postupak će biti obrnut nego kod slanja. Vršit će se dekripcija ključa, zatim dekompresija te skidanje digitalnog potpisa s originalnog teksta.[10]



Sl. 3.9. *Algoritam primanja poruke PGP metodom*

Shematski prikaz načina pružanja usluge privatnosti i autentičnosti je prikazan na slici 3.10. [10]



Sl. 3.10. *Privatnost i autentičnost pomoću PGP-a*

Oznake na shemi imaju sljedeće značenje: [10]

Ks - sjednički ključ korišten za simetričnu enkripciju

PRa - privatni ključ korisnika A, korišten za asimetričnu enkripciju

PUa - javni ključ korisnika A, korišten za asimetričnu enkripciju

EP - šifriranje asimetričnim kriptosustavom

DP - dešifriranje asimetričnim kriptosustavom

EC - šifriranje simetričnim kriptosustavom

DC - dešifriranje simetričnim kriptosustavom

H - hash funkcija

|| - nadovezivanje

Z - kompresija pomoću ZIP algoritma

$R64$ - konverzija u radix 64 ASCII format

3.2. S/MIME

S razvojem zaštite elektroničke pošte pojavio se i problem slanja kompleksnijeg sadržaja. U ovome slučaju se radi o slanju fotografija, glazbe i videa putem elektroničke pošte. Problemi koje je trebalo riješiti zbog ograničenja prilikom slanja poruka koristeći SMTP protokol su: [2]

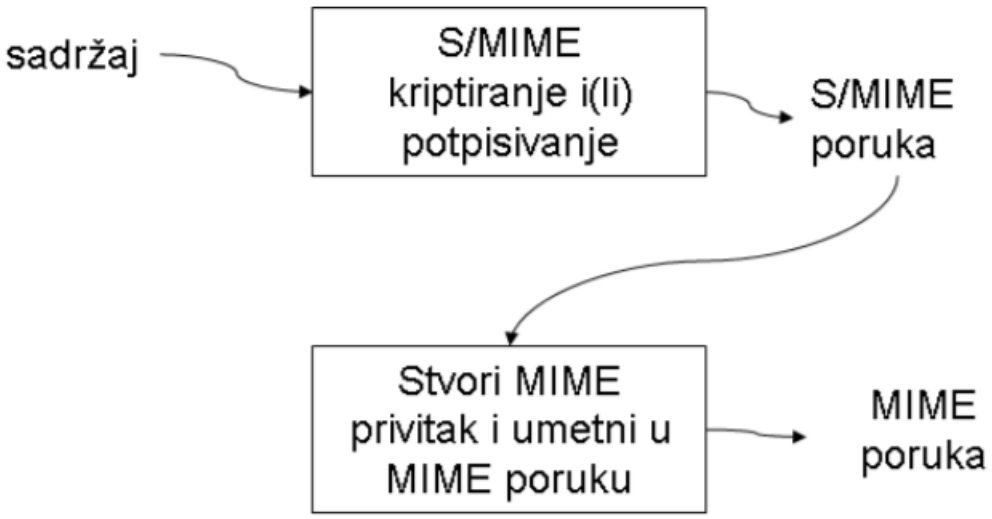
- SMTP ne može prenositi izvršne datoteke, niti druge oblike binarnih datoteka (stoga se koriste i različite sheme za konverziju binarnih datoteka u tekstualni oblik, npr. Uuencode/UUdecode sa UNIX-a)
- SMTP ne može prenositi tekst koji sadrži znakove izvan 7-bitnog ASCII skupa znakova
- SMTP poslužitelji mogu odbiti poruku veću od određenog limita
- SMTP gatewayi koji prevode između ASCII i EBCDIC skupa znakova ne koriste konzistentni način mapiranja, što može rezultirati pogreškama kod prevođenja

Iz tog razloga IETF je omogućio slanje takvih vrsta datoteka unutar poruka elektroničke pošte što je omogućilo i današnje umetanje priloženih datoteka u poruku. Taj standard je nazvan MIME (Multipurpose Internet Mail Extensions). Formati koji su podržani u standardu MIME su prikazani u tablici 3.2. [10]

Tab. 3.2. *Podržani formati u MIME standardu*

Type	Subtype	Description
Text	Plain	Unformatted text; may be ASCII or ISO 8859.
	Enriched	Provides greater format flexibility.
Multipart	Mixed	The different parts are independent but are to be transmitted together. They should be presented to the receiver in the order that they appear in the mail message.
	Parallel	Differs from Mixed only in that no order is defined for delivering the parts to the receiver.
	Alternative	The different parts are alternative versions of the same information. They are ordered in increasing faithfulness to the original, and the recipient's mail system should display the "best" version to the user.
	Digest	Similar to Mixed, but the default type/subtype of each part is message/rfc822.
Message	rfc822	The body is itself an encapsulated message that conforms to RFC 822.
	Partial	Used to allow fragmentation of large mail items, in a way that is transparent to the recipient.
	External-body	Contains a pointer to an object that exists elsewhere.
Image	jpeg	The image is in JPEG format, JFIF encoding.
	gif	The image is in GIF format.
Video	mpeg	MPEG format.
Audio	Basic	Single-channel 8-bit ISDN mu-law encoding at a sample rate of 8 kHz.
Application	PostScript	Adobe Postscript format.
	octet-stream	General binary data consisting of 8-bit bytes.

Povećanjem korištenja ovog standarda razvila se i potreba za zaštitom podataka prenesenih ovih standardom. Tako je IETF osmislio sigurnosno proširenje MIME-a pod nazivom S/MIME gdje slovo S označava riječ „Secure“ što bi u prijevodu označavalo sigurnost. Za razliku od PGP-a koji se više koristi za osobne potrebe korisnika, S/MIME češće ima komercijalnu i institucionalnu primjenu. Prva verzija S/MIME standarda se veže uz 1995. godinu te je razvijena od strane RSA Data Security korporacije. Druga verzija je izašla 1998., a treća verzija 1999. godine. Treća verzija S/MIME standarda je danas najčešće korištena verzija te su u nju uvedene nove funkcionalnosti kao što su mogućnost RSA enkripcije prilikom slanja poruke (do 3. verzije RSA enkripcija se koristila samo za provjeru potpisa), zaštićena potvrda primitka, sigurnosne oznake poruka te višestruka zaštita iste poruke. S/MIME omogućuje enkripciju javnog ključa i digitalno potpisivanje poruka elektroničke pošte. Kriptirane i digitalno potpisane poruke se omataju posebnim MIME formatom te se kao takve umeću u novu MIME poruku. Pojednostavljeni shematski prikaz načina funkcioniranja S/MIME standarda je prikazan na slici 3.11. [11]



Sl. 3.11. *Omatanje S/MIME poruke MIME formatom*

S/MIME format poruka se temelji na binarnom CMS (Cryptographic Message Syntax) formatu. CMS je standard koji opisuje sintaksu za zaštitu sadržaja elektroničkih poruka, a koristi se za enkripciju, autentifikaciju, sažimanje i potpisivanje poruka. U sintaksi je opisan način na koji se upisuje sadržaj u poruku kao i tip sadržaja, korišteni, algoritmi za enkripciju/dekripciju, podatci o certifikatu i ostale bitne stvari. Ovaj format prepoznaje 6 tipova podataka [11], a to su :

1. Data
2. Signed-data
3. Enveloped-data
4. Digested-data
5. Encrypted-data
6. Authenticated-data

Funkcionalnost S/MIME standarda je slična kao i kod PGP-a te se koriste isti algoritmi prilikom zaštite elektroničke pošte. Algoritmi koji se koriste u ovom standardu su prikazani u tablici 3.3.[10]

Tab. 3.3. *Algoritmi korišteni u S/MIME standardu*

Function	Requirement
Create a message digest to be used in forming a digital signature.	MUST support SHA-1. Receiver SHOULD support MD5 for backward compatibility.
Encrypt message digest to form a digital signature.	Sending and receiving agents MUST support DSS. Sending agents SHOULD support RSA encryption. Receiving agents SHOULD support verification of RSA signatures with key sizes 512 bits to 1024 bits.
Encrypt session key for transmission with a message.	Sending and receiving agents SHOULD support Diffie-Hellman. Sending and receiving agents MUST support RSA encryption with key sizes 512 bits to 1024 bits.
Encrypt message for transmission with a one-time session key.	Sending and receiving agents MUST support encryption with tripleDES. Sending agents SHOULD support encryption with AES. Sending agents SHOULD support encryption with RC2/40.
Create a message authentication code.	Receiving agents MUST support HMAC with SHA-1. Sending agents SHOULD support HMAC with SHA-1.

Iz tablice je vidljivo kako će se koristiti gotovo isti algoritmi kao i kod gore navedene PGP metode. Za digitalni potpis će se koristiti Diffie-Hellman postupak koristeći DSS ili RSA algoritam, dok će se za enkripciju poruke koristiti 3DES. Upravo zbog korištenja istih algoritama može se uočiti sličnost sa PGP metodom. Slično kao i kod PGP standarda, S/MIME omogućuje korisniku 2 metode zaštite, a to su digitalni potpis i enkrpicija poruke.

3.2.1. **S/MIME certifikati**

Korištenje S/MIME standarda podrazumijeva preuzimanjei i instalaciju certifikata. Pomoću certifikata se osigurava autentičnost, privatnost i integritet podataka. Certifikati se mogu preuzeti na stranicama ovlaštenim za preuzimanje certifikata što će biti prikazano u sljedećem poglavlju. S/MIME standard zahtjeva korištenje certifikata formata X.509v3, a primjer takvog certifikata je dan na slici 3.12 [11]

```
Certificate:
   Data:
       Version: 3 (0x2)
       Serial Number: 1 (0x1)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Validity
           Not Before: Aug  1 00:00:00 1996 GMT
           Not After : Dec 31 23:59:59 2020 GMT
       Subject: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
                OU=Certification Services Division,
                CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus (1024 bit):
                   00:d3:a4:50:6e:c8:ff:56:6b:e6:cf:5d:b6:ea:0c:
                   68:75:47:a2:aa:c2:da:84:25:fc:a8:f4:47:51:da:
                   85:b5:20:74:94:86:1e:0f:75:c9:e9:08:61:f5:06:
                   6d:30:6e:15:19:02:e9:52:c0:62:db:4d:99:9e:e2:
                   6a:0c:44:38:cd:fe:be:e3:64:09:70:c5:fe:b1:6b:
                   29:b6:2f:49:c8:3b:d4:27:04:25:10:97:2f:e7:90:
                   6d:c0:28:42:99:d7:4c:43:de:c3:f5:21:6d:54:9f:
                   5d:c3:58:e1:c0:e4:d9:5b:b0:b8:dc:b4:7b:df:36:
                   3a:c2:b5:66:22:12:d6:87:0d
               Exponent: 65537 (0x10001)
       X509v3 extensions:
           X509v3 Basic Constraints: critical
               CA:TRUE
   Signature Algorithm: md5WithRSAEncryption
       07:fa:4c:69:5c:fb:95:cc:46:ee:85:83:4d:21:30:8e:ca:d9:
       a8:6f:49:1a:e6:da:51:e3:60:70:6c:84:61:11:a1:1a:c8:48:
       3e:59:43:7d:4f:95:3d:a1:8b:b7:0b:62:98:7a:75:8a:dd:88:
       4e:4e:9e:40:db:a8:cc:32:74:b9:6f:0d:c6:e3:b3:44:0b:d9:
       8a:6f:9a:29:9b:99:18:28:3b:d1:e3:40:28:9a:5a:3c:d5:b5:
       e7:20:1b:8b:ca:a4:ab:8d:e9:51:d9:e2:4c:2c:59:a9:da:b9:
       b2:75:1b:f6:42:f2:ef:c7:f2:18:f9:89:bc:a3:ff:8a:23:2e:
        70:47
```

Sl. 3.12. *Primjer certifikata X.509v3 formata*

Na gornjem primjeru se vide brojni podaci koje dobiveni certifikat sadržava kao što su: [11]

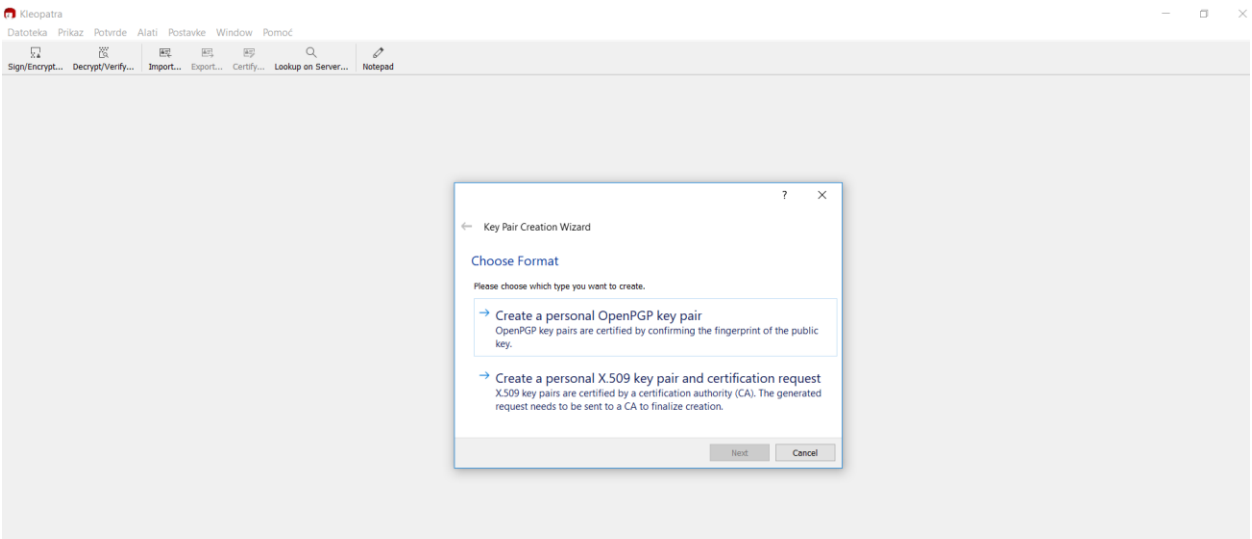
- Javni ključ
- Naziv i identifikacijsku oznaka
- Verziju X.509 preporuke
- Serijski broj certifikata
- Naziv certifikacijskog centra koji izdaje certifikat
- Algoritam koji se primjenjuje za kompresiju i digitalni potpis
- Algoritam koji se koristi za enkripciju te parametre
- Prvi i zadnji dan dok vrijedi certifikat
- Digitalni potpis certifikacijskog centra

4. IMPLEMENTACIJA METODA ZA ENKRIPCIJU ELEKTRONIČKE POŠTE

4.1. Gpg4win

Gpg4win je besplatni paket za MS Windows platformu koji sadržava GnuPG enkripcijski softver, softver za upravljanje certifikatima, ekstenziju za Microsoft Outlook te druge funkcionalnosti. On je temeljen na OpenPGP (inačica PGP-a) standardu, a od verzije 2 podržava i S/MIME standard. Unutar ovoga paketa će biti korištena aplikacija Kleopatra u kojoj će biti pokazan jedan od načina sigurne razmjene elektorničke pošte. Postupak je sljedeći:

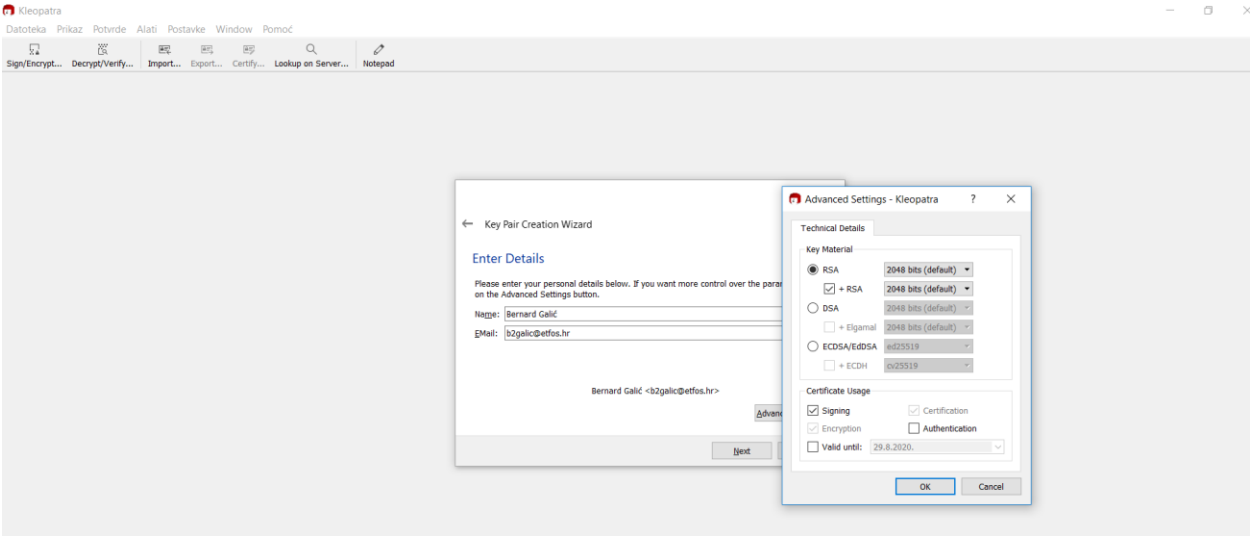
Nakon što se aplikacija Kleopatra otvori potrebno je kreirati par ključeva kao što je vidljivo na slici 4.1.



Sl. 4.1. *Odabir metode za kreiranje ključeva*

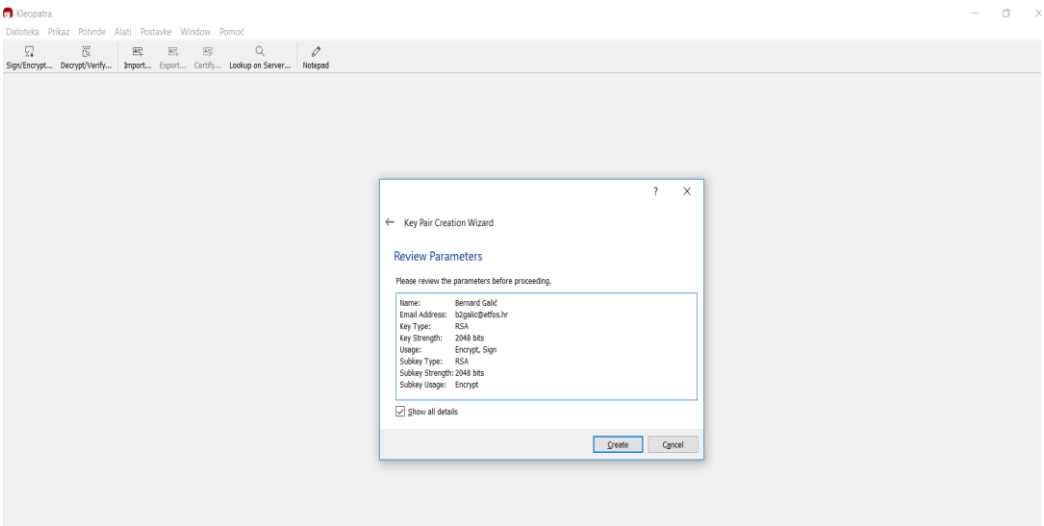
Aplikacija Kleopatra nudi dvije mogućnosti kreiranja ključeva kao što je vidljivo na gornjoj slici. U prvoj opciji će se kreirati OpenPGP par ključeva gdje će se vršiti autentifikacija na temelju „otiska prsta“ javnog ključa korisnika. Što se tiče ove opcije, kod nje nije potrebno dobiti ovlast o certifikatu od strane neke „više“ agencije nego se cijeli koncept temelji na „Web of trust“ metodi što bi u prijevodu označavalo mrežu povjerenja. U drugoj opciji se koristi S/MIME standard gdje kreirani ključevi moraju biti autentificirani od strane ovlaštene organizacije (CA – Certificate Authority) što će biti objašnjeno u nastavku.

Nakon odabira prve opcije za kreiranje OpenPGP para ključeva na ekranu će se tražiti od korisnika da unese određene podatke što je vidljivo na slici 4.2.

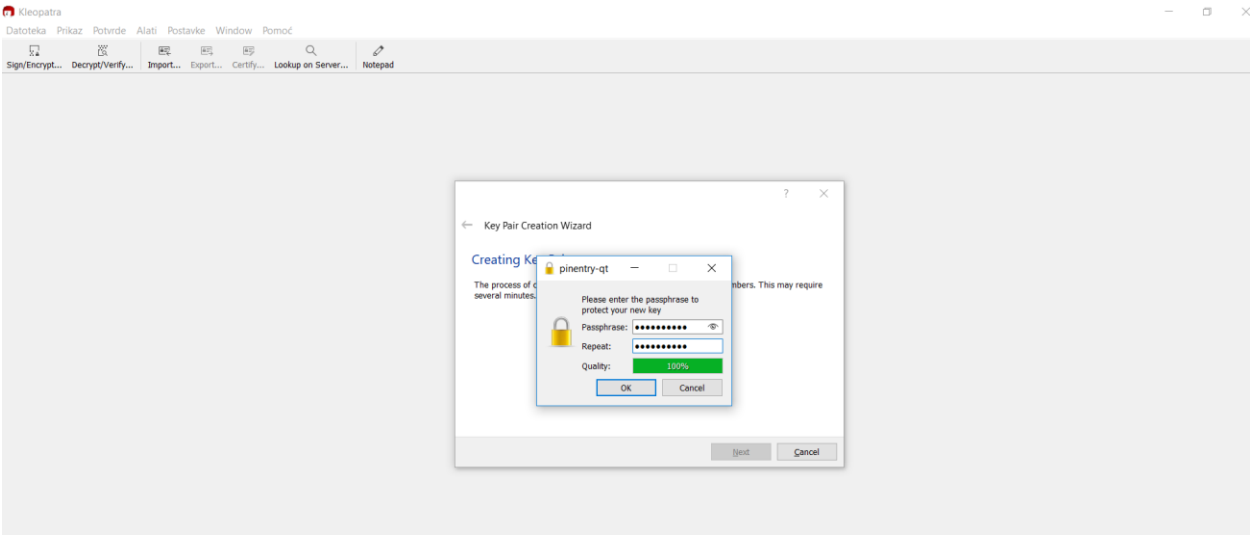


Sl. 4.2. *Unos podataka i odabir naprednih parametara*

Kao što je vidljivo na slici, korisnik unosi svoje ime i adresu elektroničke pošte, a postoj ii mogućnost postavljanja naprednih parametara kao što su algoritmi koji će se koristiti te namjene certifikata. Pritiskom na OK izvršit će se posljednji pregled certifikata prije kreiranja te će se tražiti unos lozinke koja će štititi par kreiranih ključeva. Navedeno je prikazano na slikama 4.3. i 4.4.

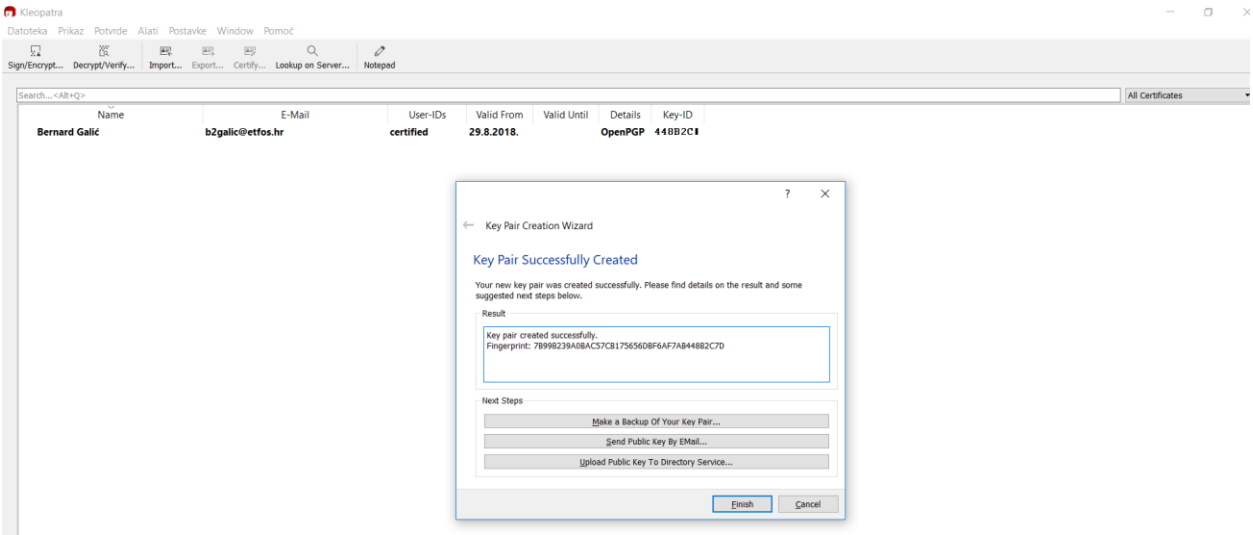


Sl. 4.3. *Prikaz parametara certifikata prije samog kreiranja*



Sl. 4.4. *Unos osobne lozinke*

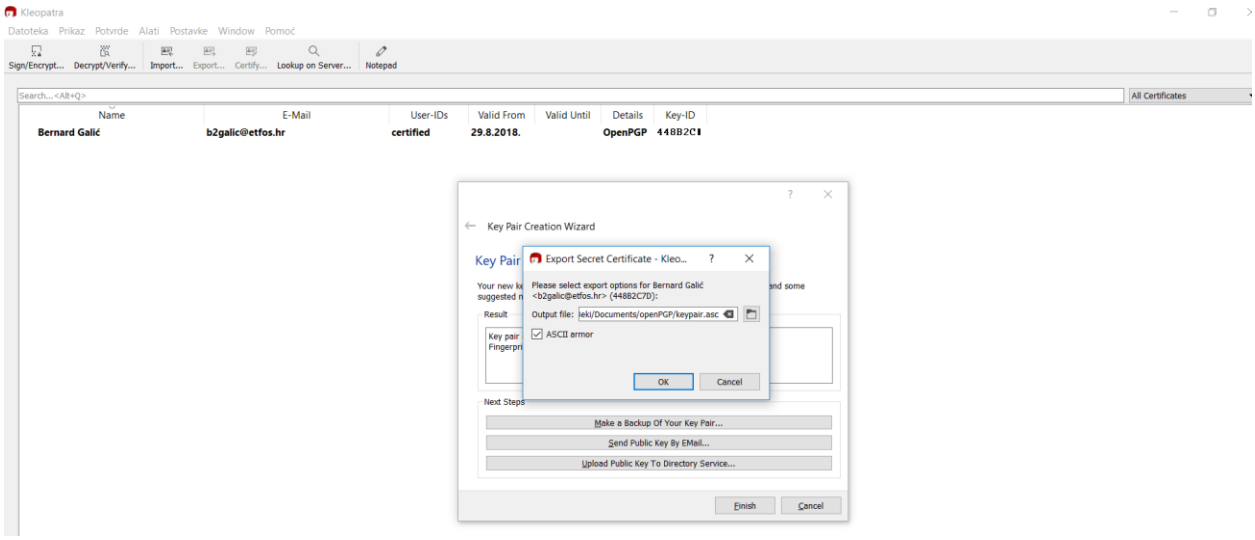
Nakon što se što se pritisne tipka OK, izvršit će se kreiranje navedenog certifikata. Dobiveni certifikat će biti vidljiv unutar aplikacije Kleopatra te će se nuditi mogućnost spremanja pohrane javnog i privatnog ključa na računalo, slanje javnog ključa putem elektroničke pošte ili postavljanje javnog ključa na javni server. Prikaz prozora koji će se otvoriti nakon kreiranja certifikata je prikazan na slici 4.5.



Sl. 4.5. *Kreirani certifikat*

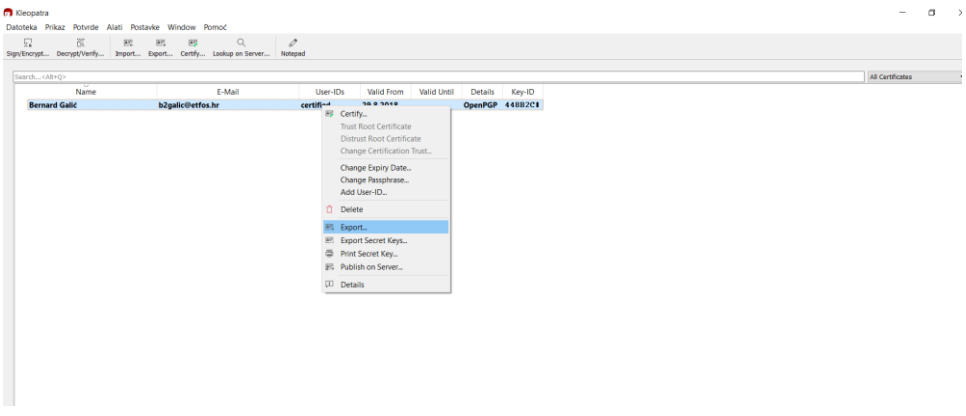
Na prozoru se vidi tzv. otisak prsta koji je vezan uz kreirani certifikat te se sastoji od 40 znakova. Ovaj otisak je jedinstven te vrijedi da nijedan drugi kreirani certifikat neće imati istu kombinaciju znakova. Odabirom prve opcije radi se „backup“ kreiranih ključeva na računalu. Poželjno je nakon spremanja privatnog ključa na računalo, taj ključ pohraniti negdje izvan

računala, primjerice na USB stick te obrisati sve tragove datoteke koja sadrži privatni ključ s računala. Kao što je u prethodnim poglavljima objašnjeno, privatni ključ je ključ koji zna samo korisnik te mu služi za čitanje enkriptirane elektroničke pošte. Javni ključ je dostupan svima te pošiljatelj mora znati javni ključ primatelja kako bi mu mogao poslati enkriptiranu poštu. Na slici 4.6. je vidljiv prozor koji se otvara prilikom pohrane ključeva.



Sl. 4.6. *Pohrana ključeva*

Prilikom pohrane se odabire gdje će se na računalu smjestiti datoteka u kojoj će biti pohranjeni ključevi te se odabire opcija „ASCII armor“ kako ključevi nebi bili pohranjeni u binarom formatu već u ASCII iz razloga da se mogu otvoriti u text editorima. Osim na ovaj način, ključevi se mogu pohraniti bilo kad kasnije na način da se odabere željeni certifikat te pomoću desnog klika odradi pohrana kao što je prikazano na slici 4.7.



Sl. 4.7. *Drugi način pohrane ključeva*

Kao što je gore navedeno, korisnik može poslati zaštićenu elektroničku poštu drugom korisniku samo ako ima njegov javni ključ. U gore prikazanoj metodi pohrane, javni ključ je moguće pohraniti u formatu .asc ili .gpg kako bi se mogao pročitati u nekom text editoru. Za potrebe ove simulacije, korišten je javni ključ osobe Mateo Miličić koji je poslan putem elektroničke pošte. Prilikom slanja javnog ključa drugoj osobi, u tijelo poruke se može kopirati kompletan javni ključ ili se javni ključ može poslati kao prilog. Javni ključ osobe Mateo Miličić je prikazan na slici 4.8.

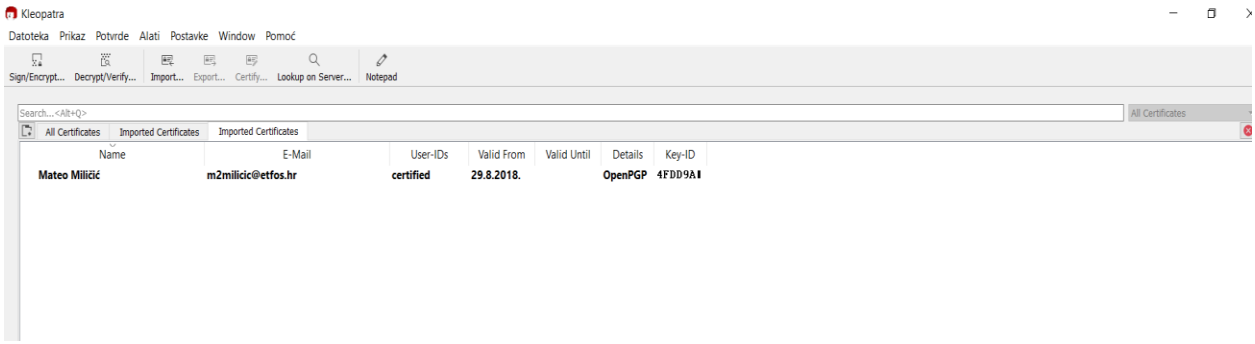

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQENBFuGrpABCADUZBRCX2I6CC/zRJarLcl31dsgljHEZ1YWtakT+hFHSUrzAlSI
I8X7YbMs92X2NxaEs2AZyDUJphaUsG2KDqTCewGJyLfKEJNJMeKEj2GSO8IxvCap
W9S0djbiO6rEPe9mxeLeimhGQYFGcopo5zId7qeJF2qTUcNzPwU2HGLPGBhj+ns3
/HIhkEElL4vMjxwKCu9kZ5tGTzTfjw6rhenhzAWtCkdbiWYsrhdwhKsB/0H69zDa
TOESqHeu3n/GaC8s6W6Icfv7kPpfkID9BpM9aILNxz5RjaAoNDOgE9KwYp38NXz8
dMn6jVID7ADcz7Y028T9THNOk1Z5Ta0wZoqHABEBAAG0JE1hdGVvIE1pbGnEjWnE
hyA8bTJtaWxpY2ljQGV0Zm9zLmhyPokBTgQTAQgAOBYhBK905gzzjnD5/AO9Mob/
1H1P3ZpDBQJbhq6QAhsDBQsJCAcCBhUKCQgLAgQWAgMBAh4BAheAAAoJEIb/1H1P
3ZpDHlsH/iw5L8inEGrq5fvlNnJ7G8dfM2XAFDd7os4HASs48v1RCC6Z5sCdZDV8
QABqaEmHCRJ3MwJD4s2h08qPXhvT9hoJYdR9wECQKPGSTLZvEAXfIgIC63tXQBJo
UDY99uLesKbrek/b36p5XOS6VtMy6yjw16glKfE+j5jqYLPQESoMmrG2+2Jij0pi
dB3kIe99MUuQmLGFu4wC9Gz1Lm0U1KNTwc7lE0sghCD2hkCEUaPrv+DdmbyXfLKG
Kn+/OscsbCj6IIO9PGI1ZUqj9Q/VSNdPPAmEat3kgJiVkWUktksJua+W6Rdy0xqc
BtMdOHYWLHCCX4GpaYPT54QjJMMoHQ+5AQ0EW4aukAEIANWfOQ8+lJ8DJ6OqvMu3
i5bkawydVyK0cyRjM7shuCnLric8e72SPhdCgrS3nQnFhy7ufBv1ClYzSn+DLIBv
HYjqYcbVCcE584eswCc4EsqPv3cpBZPYwN6oj8uQk6qmURSbDDGRvcXicjgQCtep
66mcAJKw8D9QeFFWIKPLf12YO7QvGWf1A3eDg/SKtMNs5O1HxjKOPXrgCbCnDuLh
EWgon9/0rMGbueWmQ6FpdEOomEiI9WtpSaasMTev+0vINwSKlAzR+zPE1rxuTXO0
nxu4Ts/1ro9v/dxH1ge7u+e9cq5UIyKktz13kkXESCNcZLBHvNLPll86vnGr518A
0U0AEQEAAYkBNgQYAQgAIBYhBK905gzzjnD5/AO9Mob/1H1P3ZpDBQJbhq6QAhsM
AAoJEIb/1H1P3ZpDzkIH/RM3ckh3IrOoulcLIs+9sTvZldyZ0+xtEmU9pyRECgci
6Zc2jcAC6wqcgWKYR1TtkZ3vK4k8a+k2e1EgqBchBOFyVr+lI2pGbI0KYAQRV9fC
uD7aQ/DRKr166QK5ipKatkHELGCkARdkXV7tKoR3w0p9DqdHGuAEKjquN9Ybfhl/
wE15wXIc0VELjKN5ieduwBUG0Chs27Y8nHgAmQRbVWHgCgu2tqQiM5BsvbWKuSLc
ERF7h5wXWtfkmmlhF/oojap1HXFaGo6wCYrky6Pc4Q88ryXdULFrIfdn+SSMRXFE
nsPPZe0Bidai1T0OpFTIHqXLHU7oKArnaUaQRfblXpM=
=PM0R
-----END PGP PUBLIC KEY BLOCK-----
```


Sl. 4.8. *Izgled javnog ključa*

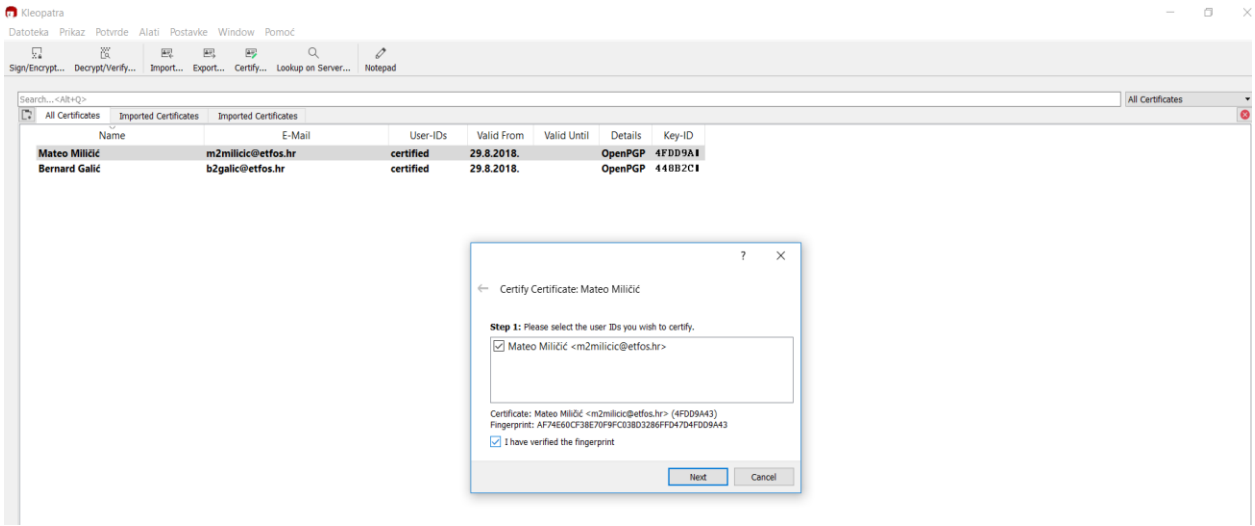
Nakon što se zna javni ključ od osobe s kojom se želi komunicirati, datoteka koja sadržava javni ključ se mora pohraniti u aplikaciju Kleopatra. Postupak je jednostavan, u aplikaciji Kleopatra se odabere „File“ te zatim „Import“ nakon čega se odabire željena datoteka koja sadržava javni ključ drugog korisnika. Nakon što se navedeno obavi, u Kleopatri se unutar „Imported

Certificates“ može vidjeti certifikat korisnika s kojim želimo komunicirati. Prikaz navedene radnje je vidljiv na slici 4.9.



Sl. 4.9. *Pohranjeni certifikat drugog korisnika*

Kako bi bili sigurni da se u ovome slučaju radi baš o korisniku s kojim želimo na siguran način razmjenjivat elektroničku poštu, u tom slučaju mu se možemo obratiti na neki drugi način te mu reći da provjeri „fingerprint“ koji se nalazi unutar certifikata. Ukoliko se niz znakova koji drugi korisnik pročita poklapa sa onime što imamo unutar certifikata kojeg smo pohranili tada je sigurno da se radi o toj osobi te se može potvrditi autentifikacija što je prikazano na slici 4.10.



Sl. 4.10. *Autentifikacija certifikata*

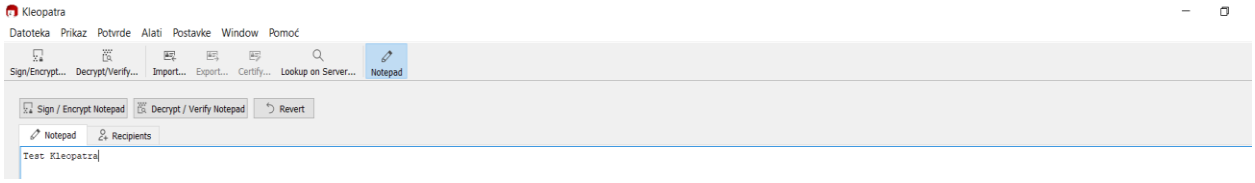
Nakon što je poznat javni ključ osobe kojoj se želi poslati poruka tada je moguće i izvršiti enkripciju poruke koja će se poslati tome korisniku. Postupak se vrši na način da se u aplikaciji Kleopatra odabere izbornik Notepad gdje će se unijeti primatelj poruke. Unutar polja Notepad se upisuje željena poruka što će u ovome slučaju biti „Test Kleopatra“. Nakon odabira tipke Sign/Encrypt upisuje se osobna lozinka korisnika te se vrši potpis i enkripcija poruke. Postupak

je prikazan na idućim slikama. Na slici 4.11. se vidi odabir primatelja za kojeg se vrši enkripcija poruke.



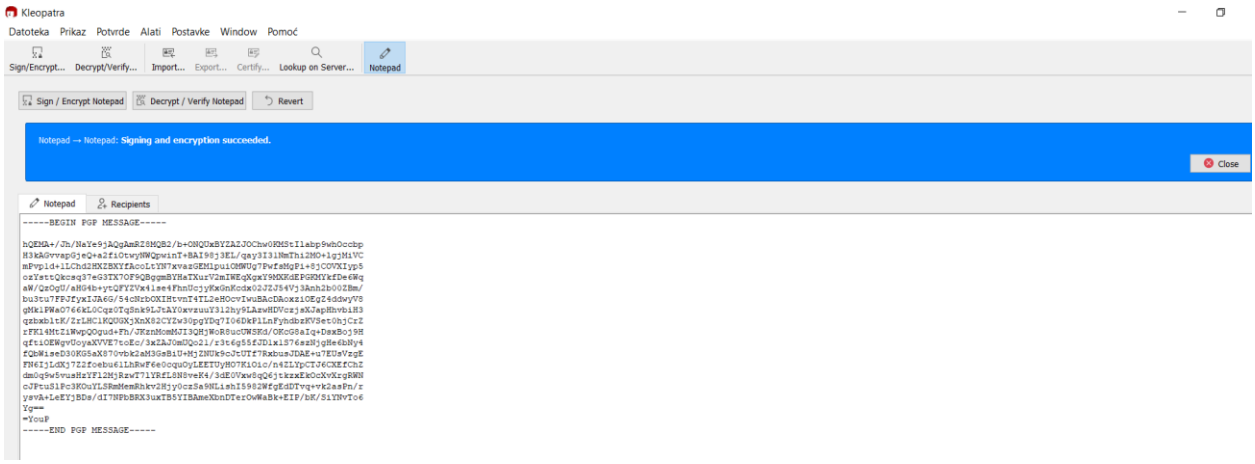
Sl. 4.11. *Odabir primatelja za kojeg se vrši enkripcija poruke*

Na slici 4.12. je prikazan prozor u kojeg se unosi tijelo poruke koje se želi enkriptirati i poslati.



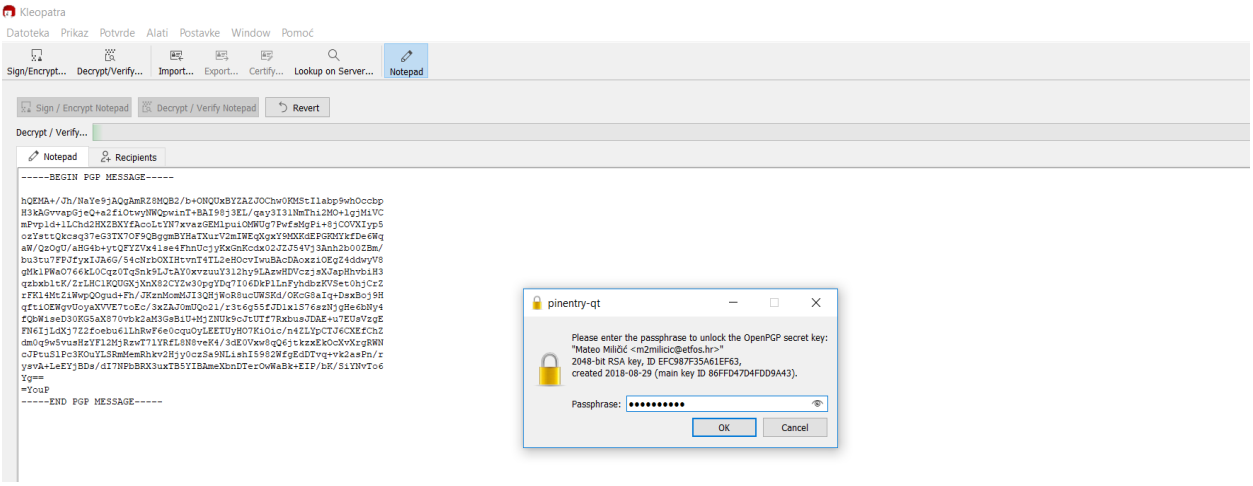
Sl. 4.12. *Unos poruke „Test Kleopatra“*

Nakon enkripcije tijelo poruke će se sastojati od niza nepovezanih znakova što je vidljivo na slici 4.13.



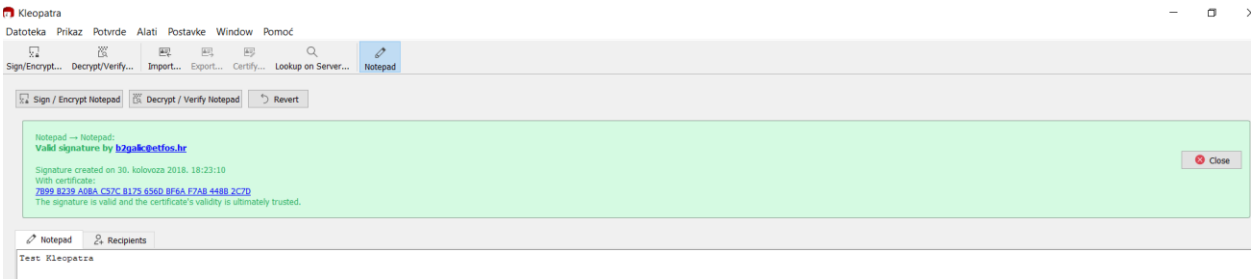
Sl. 4.13. *Poruka dobivena enkripcijom*

Nakon što se dobije enkriptirana poruka kao što je prikazano na slici 4.13. tada se takva poruka može poslati korisniku kojemu je namijenjena na način da se ili kopira tekst u tijelo poruke elektroničke pošte ili da se pošalje kao privitak. Korisnik će biti u mogućnosti dekriptirati poruku samo ako se njegov privatni ključ poklapa sa javnim ključem kojeg smo pohranili unutar njegovog certifikata, tj. ako pohranjeni javni ključ primatelja i privatni ključ čine odgovarajući par. Postupak dekripcije je prikazan na slikama 4.14 i 4.15.



Sl. 4.14. *Dekripcija poruke od strane primatelja*

Na ovoj slici je vidljivo kako primatelj mora unijeti svoju lozinku koja štiti njegov privatni ključ kako bi mogao dekriptirati poruku. Nakon unosa lozinke dobije se originalna poruka što je vidljivo na slici 4.15.

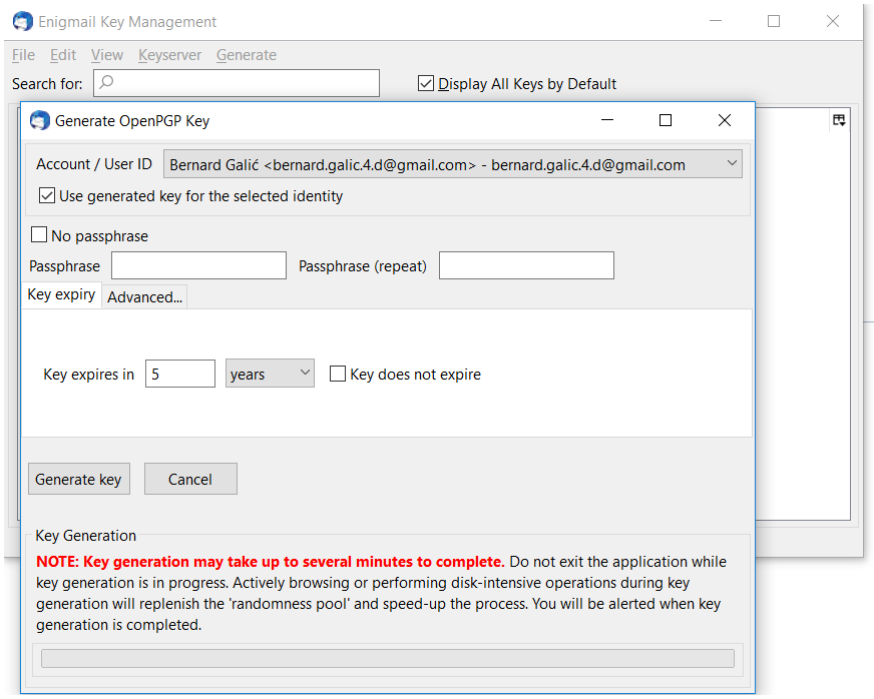


Sl. 4.15. *Dekriptirana poruka*

4.2. Mozzila Thunderbird

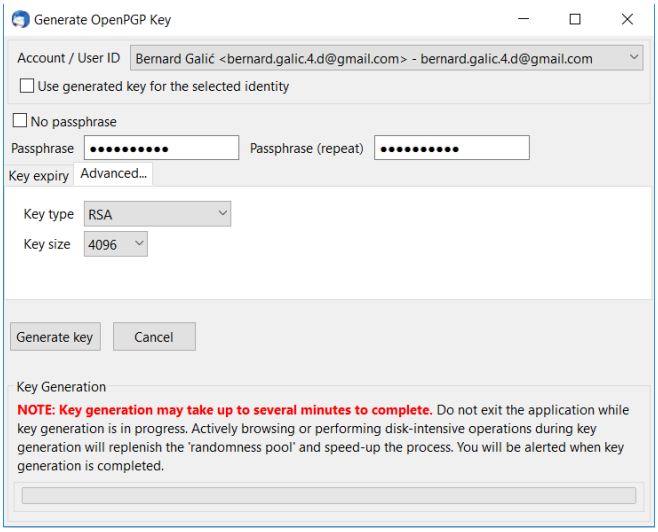
Druga metoda u kojoj će biti prikazana enkripcija elektroničke pošte putem PGP standarda je razmjena pošte putem klijenta imena Thunderbird. Ovaj internetski klijent služi između ostaloga

i kao sredstvo za razmjenu elektroničke pošte. Nakon besplatnog preuzimanja sa službene stranice, klijent se instalira na računalo te je spreman za korištenje. Unutar klijenta je moguće dodavanje više računa elektroničke pošte istovremeno te će tako u ovome primjeru biti korištena 3 računa elektroničke pošte. Prvi korak koji treba napraviti je preuzeti dodatak pod imenom Enigmail te instalirati OpenPGP software kao i u gornjem primjeru. Nakon preuzimanja svih potrebnih dodataka potrebno generirati par ključeva pomoću preuzetog dodatka Enigmail što je prikazano na slici 4.16.



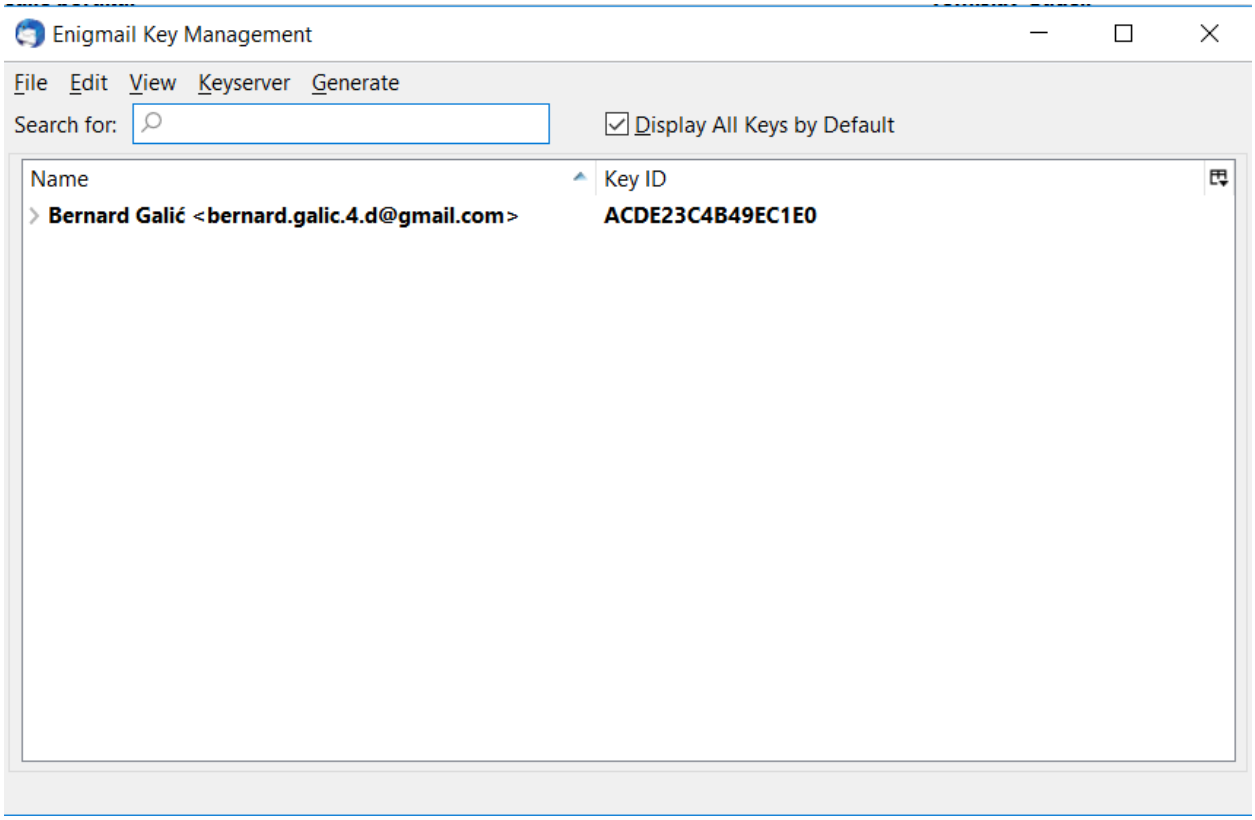
Sl. 4.16. *Generiranje ključeva*

Nakon odabira računa za kojeg se želi kreirati par ključeva, upisa lozinke i odabira trajanja ključa u naprednim postavkama se može odabrati i željeni tip ključa (algoritam) kao i njegova veličina što je prikazano na slici 4.17.



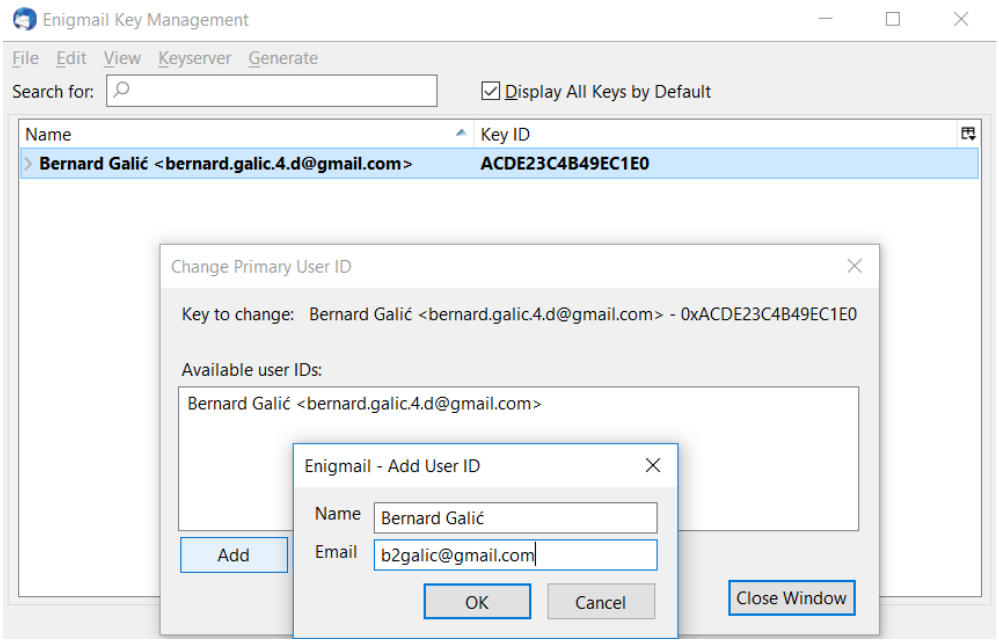
Sl. 4.17. *Napredne postavke generiranja ključa*

Kao i kod aplikacije Kleopatra, i unutar klijenta Thunderbird je moguće vidjeti kreirane ključeve za pojedini account zajedno s podatcima o tim ključevima te pripadajućim „otiskom“ kao na slici 4.18.



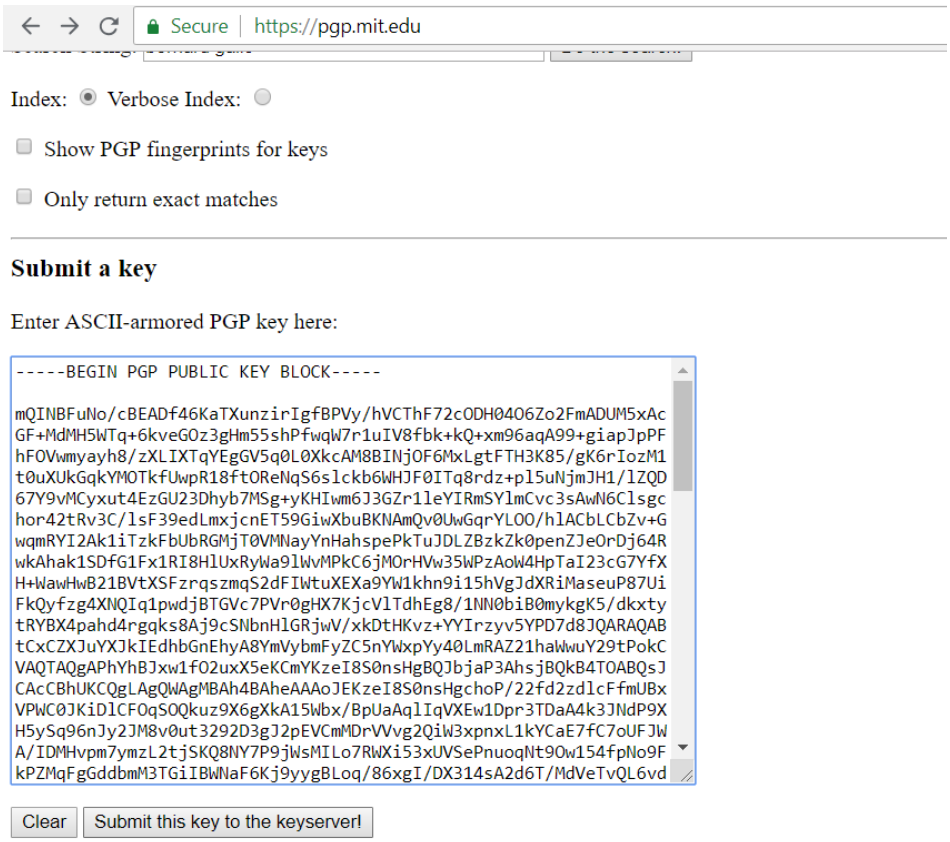
Sl. 4.18. *Prikaz kreiranih ključeva*

Jedna od dodatnih mogućnosti je i dodavanje istih ključeva na više različitih adresa elektroničke pošte. Kako bi korisnik koji koristi više računa izbjegao kreiranje ključeva za svaki račun posebno, unutar Thunderbird-a je moguće istom kreiranom paru ključeva pridružiti više različitih adresa elektroničke pošte pa će tako kreiranom ključu osim adrese bernard.galic.4.d@gmail.com biti pridružena i adresa b2galic@gmail.com. U primjeru Kleopatre je prikazano pohranjivanje privatnog i javnog ključa na računalo pa se tako i u ovome slučaju preporuča pohraniti privatni ključ na neko sigurno mjesto, dok javni ključ nije potrebno posebno štititi iz razloga što on mora biti poznat drugim korisnicima kako bi mogli sigurno komunicirati. Jedan od načina razmjene javnih ključeva je putem takozvanih Key servera gdje će se ključevi pohranjivati. Jedan od najpoznatijih takvih servera je razvijen od strane poznatog sveučilišta MIT koji će biti korišten i u ovome primjeru. Na slici 4.19 će biti prikazano dodavanje drugog računa na kreirani par ključeva.



Sl. 4.19. *Pridruživanje računa kreiranom paru ključeva*

Na slici 4.20. je prikazana pohrana javnog ključa na server.



Sl. 4.20. *Dodavanje javnog ključa na server*

Nakon što se na server doda javni ključ u tražilici je taj ključ moguće naći ili putem imena i prezimena korisnika (User ID) ili putem računa elektroničke pošte kojoj je taj ključ pridružen što je vidljivo na slici 4.21.



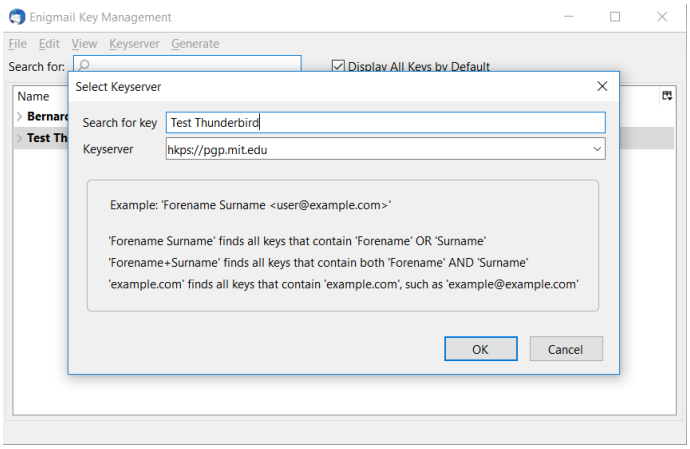
Sl. 4.21. *Rezultati pretrage javnog ključa na serveru*

Za potrebe ovog primjera kreirana je testna adresa testdipl123@gmail.com. Njen javni ključ je također dodan na isti server, a na idućim slikama će biti prikazano pretraživanje tog javnog ključa kao i dodavanje ključa unutar Thunderbird-a kako bi se mogla odvijati sigurna komunikacija s tim korisnikom. Na slici 4.22. je prikazan rezultat pretrage javnog ključa testnog računa.



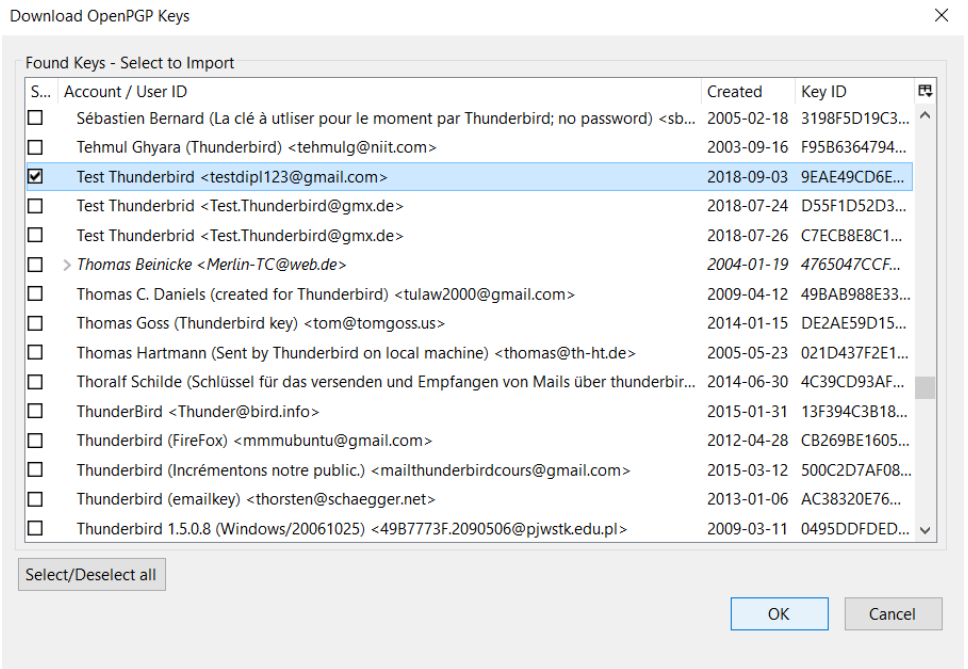
Sl. 4.22. *Rezultati pretrage javnog ključa testnog računa na serveru*

Nakon uspješne pretrage željenog javnog ključa taj isti ključ je moguće dodati unutar Thunderbird-a kako bi se moglo sigurno komunicirati s korisnikom tog ključa što je prikazano na slici 4.23.



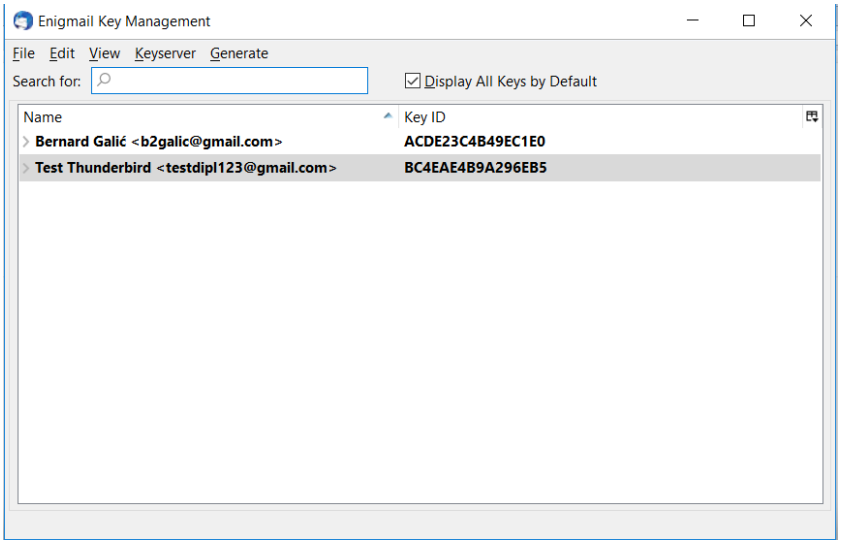
Sl. 4.23. *Dodavanje ključa unutar Thunderbird-a*

Nakon unosa željenog teksta u tražilicu, otvara se lista sa svim ključevima koji su rezultat pretrage, odabirom na željeni ključ potrebno je pritisnuti tipku „Ok“ prema slici 4.24.



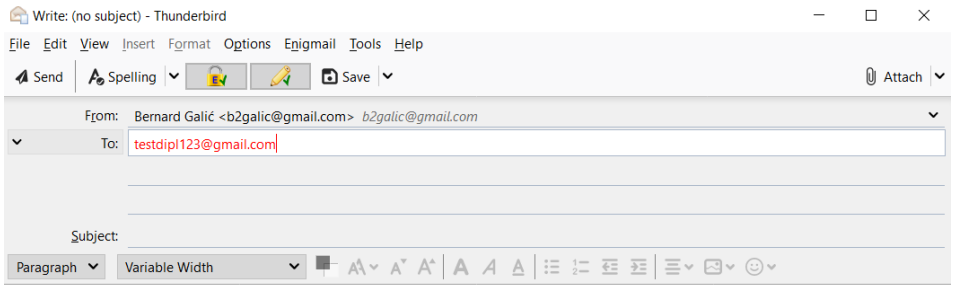
Sl. 4.24. *Preuzimanje željenog javnog ključa*

Nakon pretrage i odabira željenog javnog ključa na serveru taj isti ključ (certifikat) je moguće pronaći unutar Enigmail prozora zajedno s vlastitim ključevima (certifikatom) prema slici 4.25.



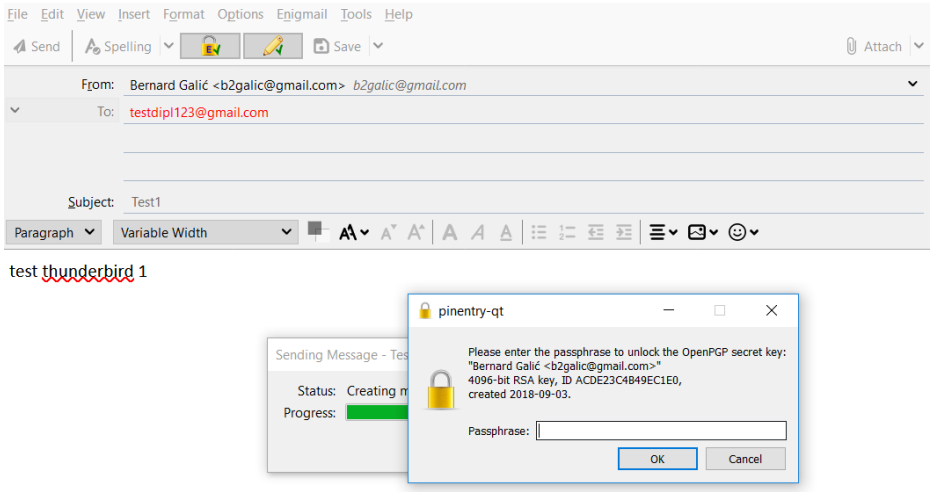
Sl. 4.25. *Preuzeti javni ključ (certifikat)*

Nakon što je pohranjen javni ključ korisnika s kojim se želi razmijeniti enkriptirana elektronička pošta moguća je i razmjena enkriptiranih poruka što je prikazano na sljedećim slikama. Osim enkripcije, prilikom slanja poruke u klijentu Thunderbird moguće je i odabrati digitalni potpis kako bi primatelj autentificirao tko je stvarni pošiljatelj poruke, a prozor gdje se unosi tijelo poruke i odabire enkripcija i digitalni potpis je prikazan na slici 4.26.



Sl. 4.26. *Slanje enkriptirane i digitalno potpisane pošte*

Prilikom slanja pošiljatelj mora opisati svoju lozinku kako bi enkriptirao sadržaj elektroničke pošte. Poruka koja se šalje je „test thunderbird 1“ što je vidljivo na slici 4.27.



Sl. 4.27. *Enkripcija elektroničke pošte*

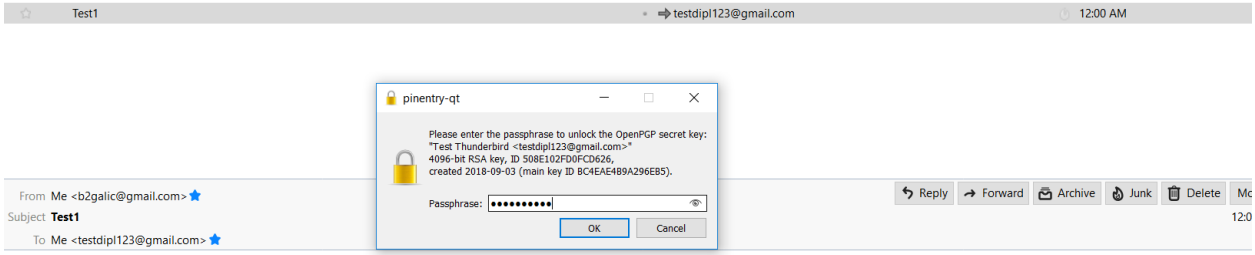
Prema slici 4.28., nakon enkripcije saržaj elektroničke pošte će biti niz nepovezanih znakova kao i u prethodnim primjerima.

```
-----BEGIN PGP MESSAGE-----

hQIMA1COEC/Q/NYmARAAjRiQDEblq5q7ApyArTKJnVlaUitZCTlnOhJrDQpI9Y7g
GUXojzXPE9NOT2j89f44uauFP+H6rzF93BpNDsbdVdtHJ6tE7lY6eWPagg3Ys3y+
ZazRdmfdlmPuB6bdS528w2nu6MpnUBWV8H/SmBAY6+F9kg2SugU08E1VrDNgExhX
0hKzCO8PJUwuXuoW/FWPiUKcqmCGA39VafO5rPFeWe+ba2kdvqRGeLuBORh9Z94U
5Y8bKuVUZ31CvVqMmdbvvqiryEswUwVTfSaN07evqQnoFZzvitoFwrDOXBUFehgL
DdT7PeH3Rr1VQOehGob8ALIRr1rH7vTIk4NSvm3DmJiSds/5Null0+E12Gzrd4TR
Xm0W4zTRi5lNuCd9liQ16D8pU9DRBYO6MmvFkRQtoJNcxPOt6qvxZ4LXi6bqjS0T
BeVhRSKnL3WM65a0DaRmm6dh8GTjwYx7/PL3IfomPtcXmr2ROA9bATbdxu+urstq
yZ30NCR5ouV49pDhYpl8nBuB6Gtx0NiSTH7FRxNnS7IVyCzYnB0C0WUXueClcXir
APcE6WZ5gYcc41orQMkqR48O4MThofp0clEZmZ1Alus6vFJk+1JD3V3Rqz+RWHFU
L9qE1PG0r3Tn0s2bQ7PTFb1BEnuu672zIdDYwKQeVr/Fcv3WcBHeKuorCePftHCF
AgwDwkbjbpTV4zIBD/4nOh37E53QGK8lSutEdX5qU90JWi6XbnEtu0JGopi/U3+j
PZ/uB2pbER3dJepdLszUaRsaMKiZYJl5g+aVOfN6v3t6QNb7HqNnUR5JS+5/16MD
4Qft7EgBaSFeegH6qNdJJs29CR9GRzdUBD4+EajTgNhWKHIPmvHGAPLI25XO56ag
MJP9aPzsFZKirrRpwqzZTrCQ3PhGokxvxh6njsGEYeFjuuedzDp5F1DJgIeuzig6
olRVMX4wla2GVBG4t5MrBlbwU6XsPESmY6HbtXoBJpLuJah+TWB1WGZlZEsAb0Jn
DMV7ALrIGrrMVthVOiFd9GS0RMRdhRinbF0Z3WJ/Xaz58qhRmPTltMB8vtfD9ZIA
wIgwiNzGBOFl3+55vJe7PzQuDbjCFJpiGbYjqv7yPSOnBNbLzq8oQ/wxwyTf7XEn
ns/dUx+j3MGuM0hh1hqTsg/8IQxs6DCcIupZekz8uAFM2cH6FwKcDbkTCdh8v9k0
0iP/g7dNCRcA2c1MHv+A1mcN+uOjGwD1KcBHPDBJ04jo06NqH8C6EqGJhaEOouc6
0mK1vNx3Evk9kG3ijARIRskBf9BrHKaAoUqQXVWCDHzRXmhqED/MlXsZr9eHkVMJ
JT64zQ3zSzN9U3jk34LrYx7Pjh58JipRAtlg++In53VW7C9HyGezdpEiIa2QH9Lp
AXw1nXutghsF4KND9spTRvwwnHpLw0r0JBvvnP72+DNYONZWu3japlyYKqRGZ65W
jX2bZ68epimVip+pgv4MZ0S/d3LWOLE+4WGM+bZQ/SP77n8ojZE9yvSjLE60iX/v
jyKo8CyKqRlzLWCOWMoVwn3NbfdM3txVyRp3IGZ99nE2XEF23ki2FTfvWg9uVnQ2
jvLAiNroBRdCOPzg3bcBlTQV4seGUDA1LZJ/LRZvgStmgmUFSKmlFa6PZD95ehC9
YZ2DKqvWZC1Cwx8edVmX4MrNl0+oMAf95gUgTb/9z1hyvOhcqQaBh36+7VIkWmf+
vxenVNYgQwqayf1e5ynMhEFT7jOsx0wcVWzWPiwPP0wGFptVU5oAMG2YmzDJDKiT
v9LrId7sQ+G2uwNYmreNUKurNMBzqhriwRZL6UvhGWWjB6U5XCrPyXwOipD5vprV
9/KDmcEcNYjKKfLn/FceNwSCGhOx54uQ2EERuGMc4oHFbAqzjVS39MupdKoxscW/
KyvxomnMYIWnRJm20vhfduzWK629qMZ4SJnTgeDLMYIXUs7gE619+AchF9ZxzDrB
BcK4mjQNp7eXicjAq2Qdif4z6cAmiTMLARpYY/efSrLRlwHIdj9v9glw/kCl1a7m
oVAuOl5n4jYDuLZeDjKKGzWVBjcsUnUPq+tXpheiMtLBNn5trYAWns3G0fyA1neD
L/QGWosCVmNjVsfsRhb50iL8ZExFGMgchUlTHS6S1+sKRBlWFoq7QEeUBo2M/1RI
wA92hmxGcpXJtegl8JNAjwIiSiYau2w6z1eRreNXdYWy6/fWgCvIvzKIzFoiKiey
spMvlAIalfl0N7oQlvQrYiofN+KGau5JxxI1TVB0I2XwFIDNS8//j7HFw8esRpKP
```

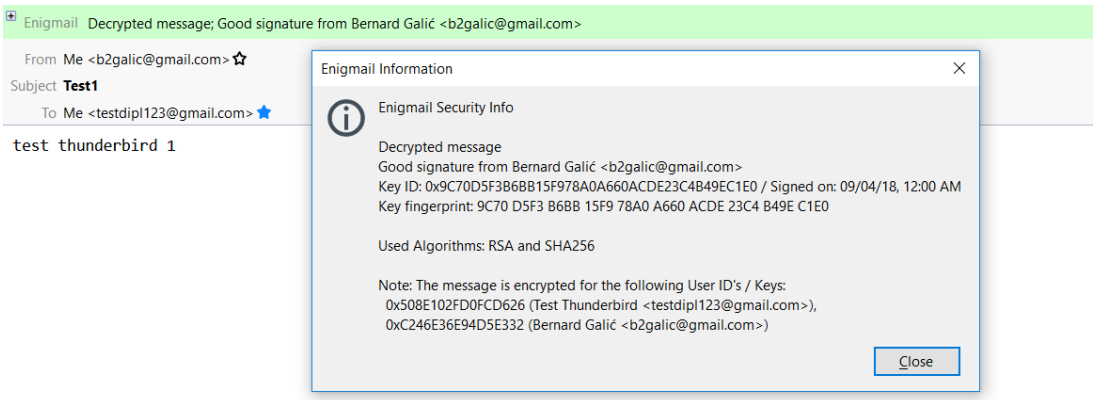

Sl. 4.28. *Prikaz enkriptiranog sadržaja elektroničke pošte*

Po primitku enkriptirane elektroničke pošte korisnik računa testdipl123@gmail.com će morati unijeti svoju lozinku kako bi mogao otvoriti elektroničku poštu prema slici 4.29.. Nakon dekripcije, primatelj vidi originalnu poruku te može pogledati detalje o primljenom sadržaju.



Sl. 4.29. *Dekripcija elektroničke pošte*

Unutar informacija o primljenoj dekriptiranoj poruci, korisnik može vidjeti algoritme koji su korišteni te podatke o pošiljatelju prema slici 4.30.

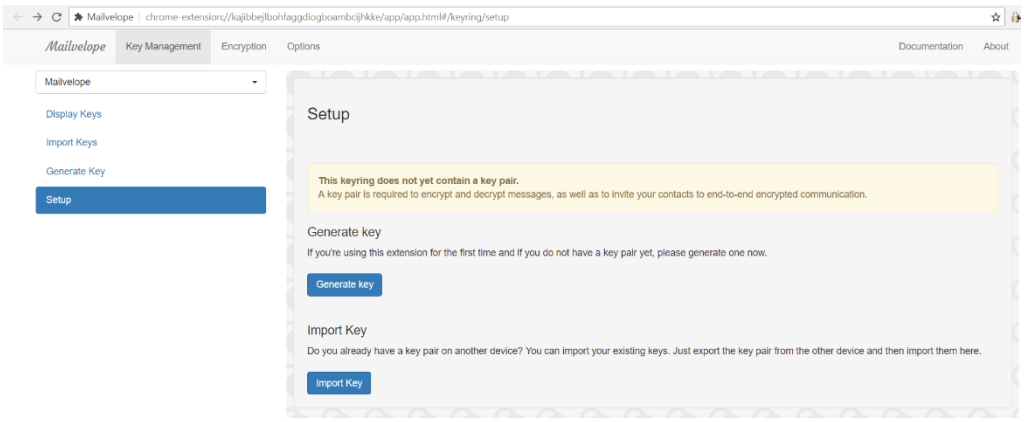


Sl. 4.30. *Dekriptirani sadržaj elektroničke pošte*

4.3. Mailvelope

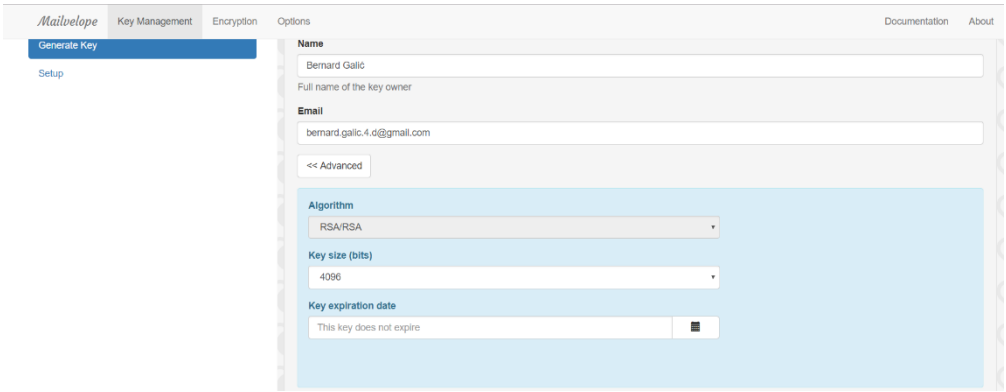
Sljedeći primjer metode za enkripciju elektroničke pošte je različit od prijašnjih primjera iz razloga što on ne zahijeva preuzimanje dodatnog software-a na računalo. U ovome primjeru je korišten internetski preglednik Google Chrome te Gmail klijent u kojemu se na početku mora instalirati dodatak pod nazivom Mailvelope. Dodatak Mailvelope se temelji na PGP standardu te je postupak sličan kao i u prethodnim primjerima što se tiče generiranja te pohrane privatnih i javnih ključeva. Nakon generiranja privatnog i javnog ključa biti će potrebno dodati javni ključ korisnika s kojim se želi komunicirati. Javni ključ drugog korisnika se može dodati ručno ili pretragom na odgovarajućem serveru za pohranu javnih ključeva. Prilikom slanja poruke u prozoru za slanje se prikazuje Mailvelope ikona na koju je potrebno kliknuti za enkripciju. Zatim se otvara novi prozor koji je neovisan o klijentu kojeg se koristi te se u njemu unosi primatelj za kojeg se sadržaj poruke enkriptira kao i sadržaj poruke. Nakon klika na gumb „Encrypt“ poruka se prikazuje u enkriptiranom obliku te je spremna na slanje. Primatelj će morati unijeti odgovarajuću lozinku koja štiti njegove ključeve kako bi mogao pročitati poruku. Na idućim slikama će biti prikazan postupak enkripcije putem ove metode.

Na slici 4.31. je prikazan prvi korak generiranja ključeva unutar Mailvelope-a.



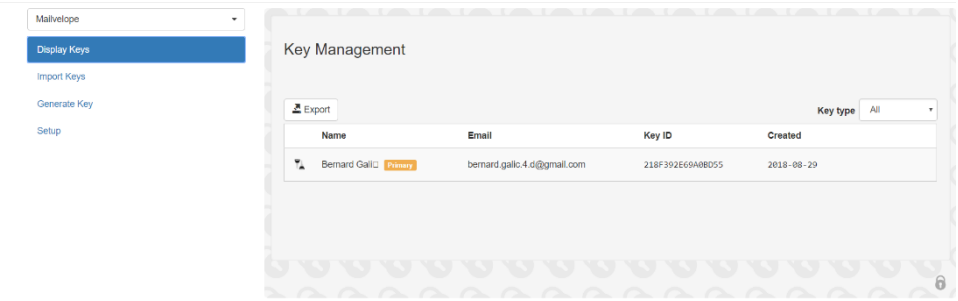
Sl. 4.31. *Generiranje ključeva unutar Mailvelope-a*

Prema slici 4.32. je vidljiv prozor u kojem korisnik podešava parametre prije same kreacije ključeva.



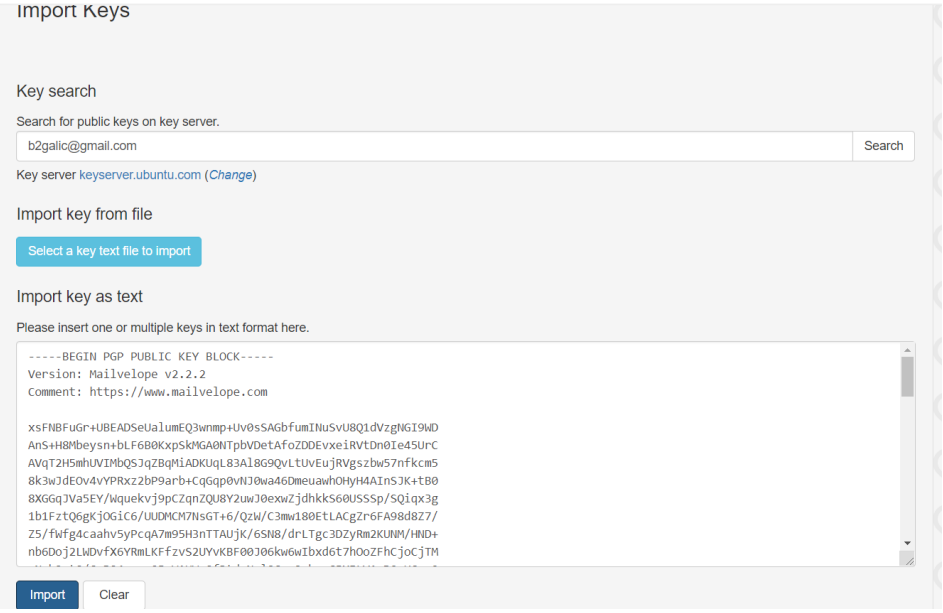
Sl. 4.32. *Napredne postavke kod kreiranja ključeva*

Nakon kreiranja, moguće je vidjeti kreirane ključeve te podatke o njima prema slici 4.33.



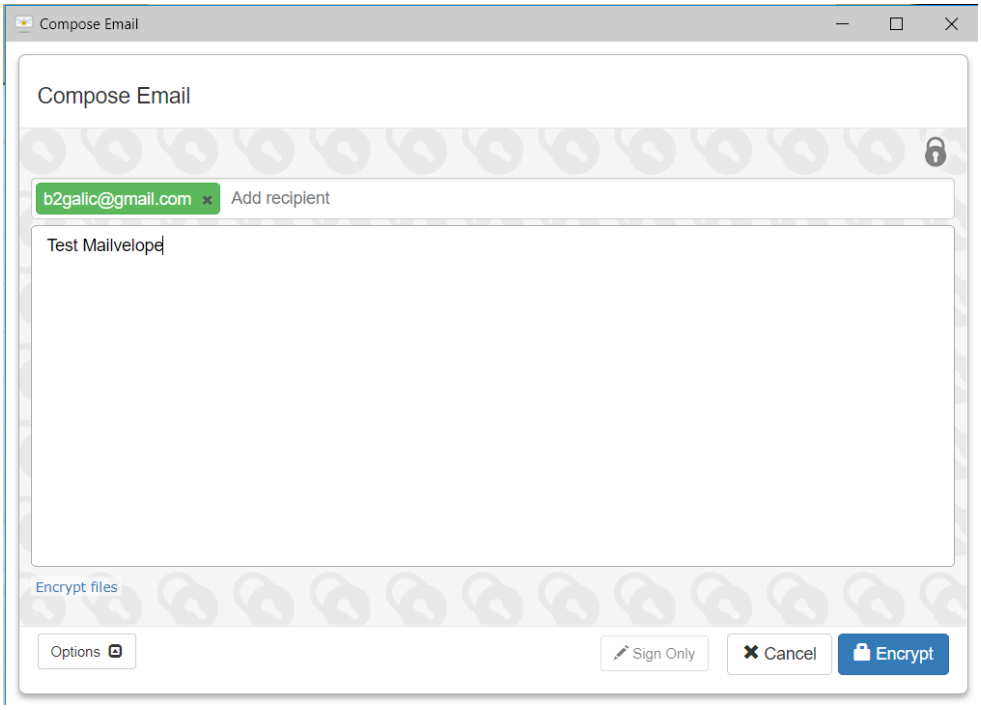
Sl. 4.33. *Prikaz kreiranih ključeva*

Kao i u prethodnim primjerima, korisnik mora dodati javni ključ osobe s kojom želi komunicirati. U ovome slučaju korisnik će dodati javni ključ u tekstnom obliku prema slici 4.34.



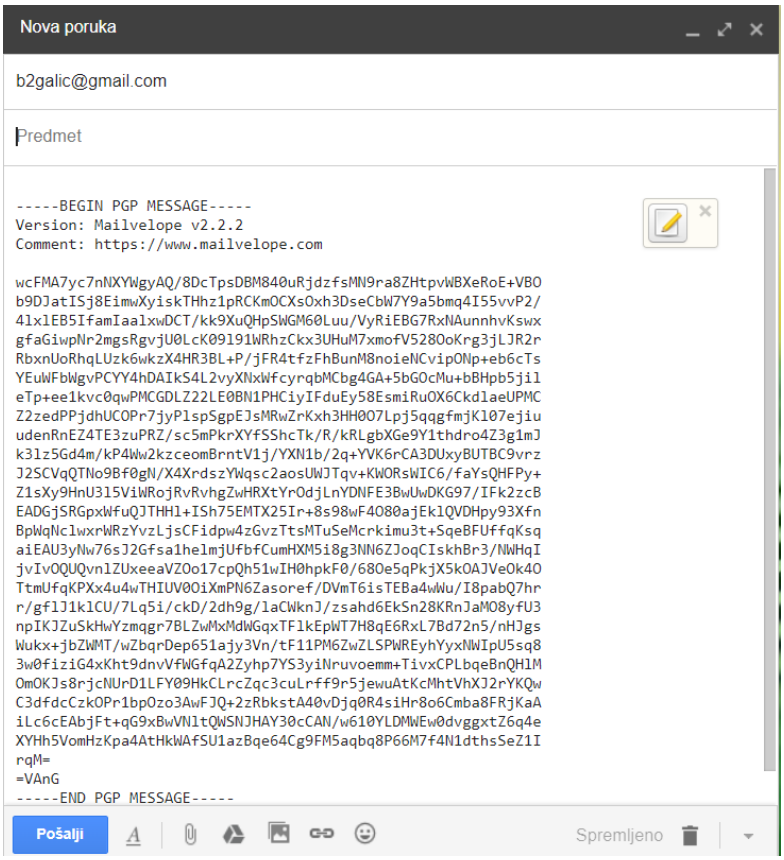
Sl. 4.34. *Dodavanje javnih ključeva drugih korisnika*

Na slici 4.35. je vidljiv prozor unutar kojeg korisnik unosi primatelja i tijelo poruke.



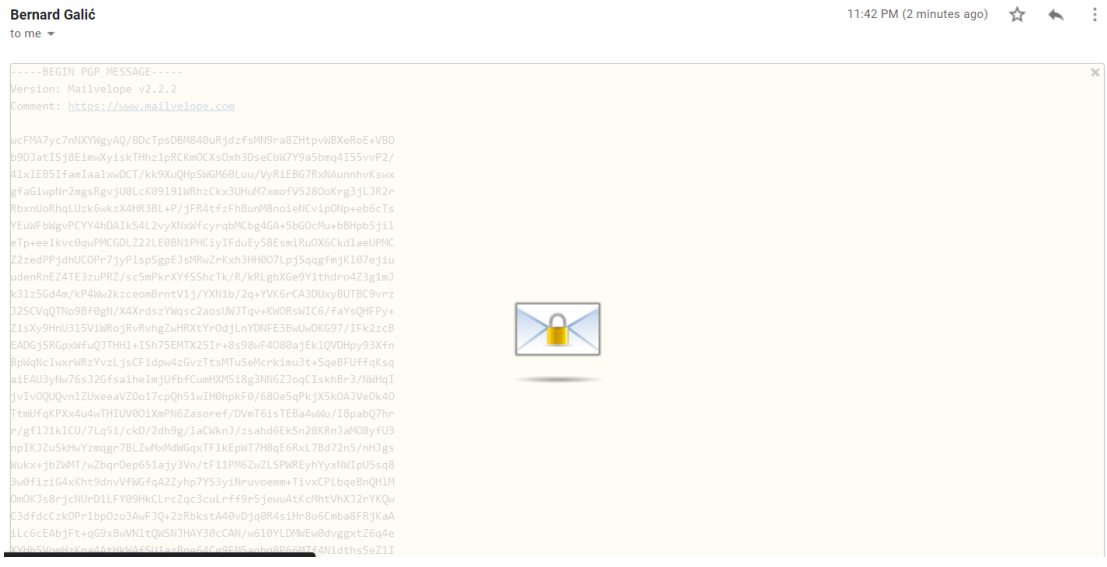
Sl. 4.35. *Slanje poruke*

Prije samog slanja, poruka se enkriptira u niz nepovezanih znakova kao u prethodnim primjerima. Navedeno je prikazano na slici 4.36.



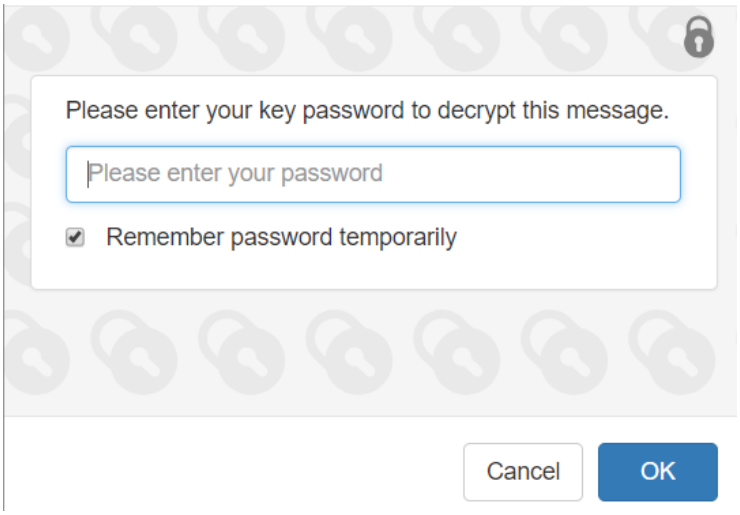
Sl. 4.36. *Prikaz enkriptirane poruke prije slanja*

Na drugoj strani, primatelj će primiti poruku u obliku prikazanom na slici 4.38.



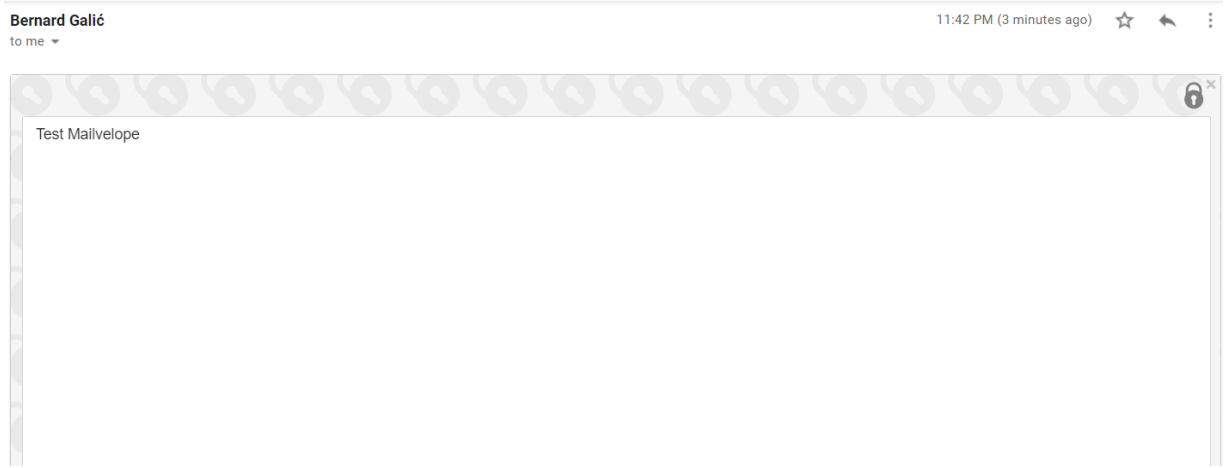
Sl. 4.37. *Prikaz primljene enkriptirane poruke*

Kako bi otvorio poruku, primatelj mora unijet lozinku koja štiti njegove ključeve (Slika 4.38.) .



Sl. 4.38. *Unos lozinke za dekripciju primljene poruke*

Ukoliko je primatelj unio ispravnu lozinku tada će vidjeti originalnu dekriptiranu poruku (Slika 4.39.).

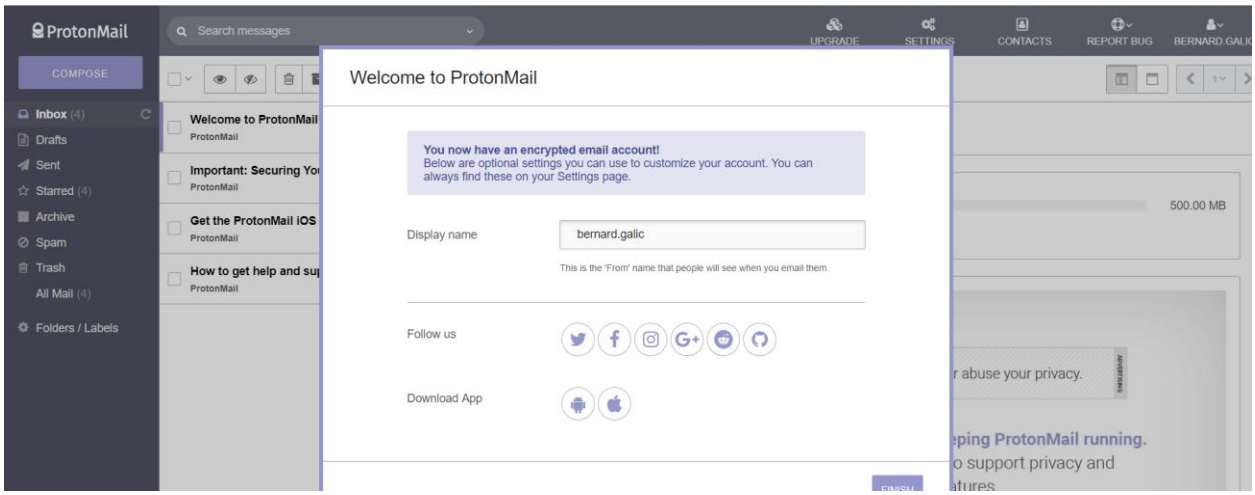


Sl. 4.39. *Dekriptirana poruka*

Pomoću Mailvelope-a se mogu slati i privitci na način da se unutar Mailvelope-a dodaju dokumenti koji se žele enkriptirati te klikne na gumb Encrypt. Postupak dekriptiranja je jednak.

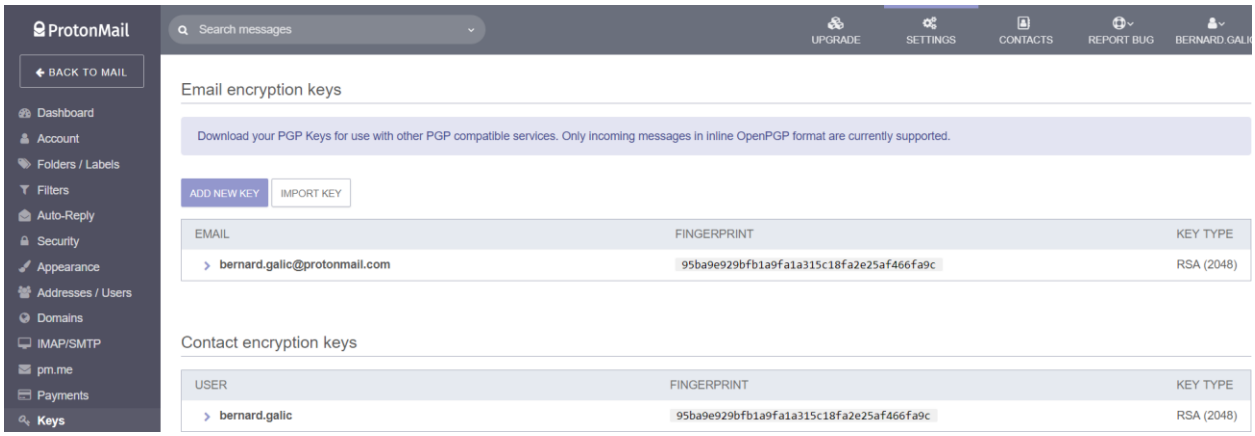
4.4. ProtonMail

Kao posljednji primjer će biti prikazan klijent elektroničke pošte s ugrađenim alatima za enkripciju elektroničke pošte. Kod ovog klijenta nije potrebno preuzimati posebne dodatke ili software za enkripciju pošte te je izrazito jednostavan za korištenje. Radi se o ProtonMail-u koji je razvijen 2014. godine u Švicarskoj. Prilikom odlaska na službenu stranicu nudi se mogućnost kreiranja računa koji će imati nastavak @protonmail.com. Prilikom kreiranja moguće je odabrati besplatnu verziju ili veriju koja se naplaćuje, a u ovome primjeru će biti korištena besplatna verzija koja također nudi mogućnost potpune enkripcije elektroničke pošte. Nakon uspješnog kreiranja te ulaska na račun pojavljuje se iduća slika gdje se može postaviti ime koje će biti vidljivo drugim korisnicima prilikom komunikacije (Slika 4.40).



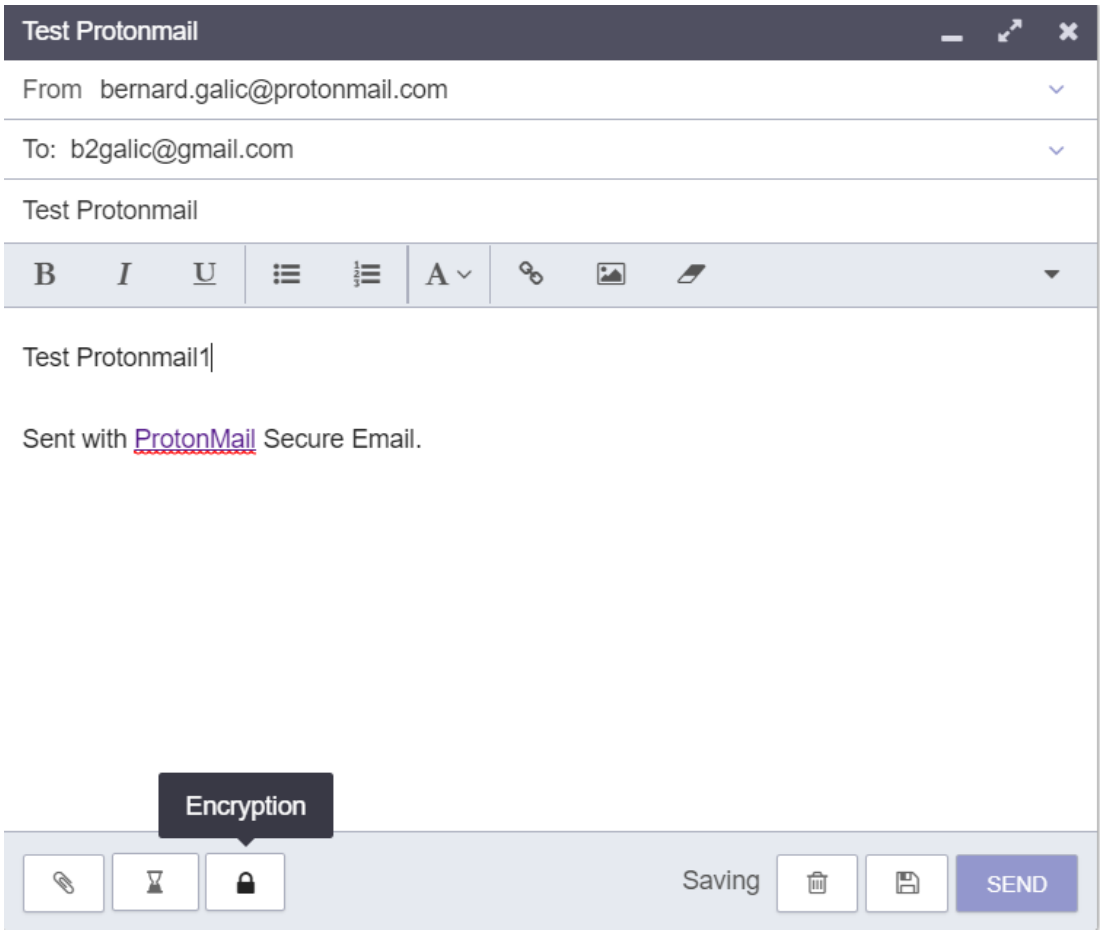
Sl. 4.40. *Prikaz uspješno kreiranog računa*

Kao i kod prijašnjih i u ovome slučaju se radi o korištenju privatnih i javnih ključeva samo što se ovdje prilikom kreiranja računa automatski kreiraju i ključevi za taj račun. Ovaj pristup čini enkripciju izrazito jednostavnom za korištenje iz razloga što nije potrebno skidati dodatke te kreirati ključeve samostalno već je u postavkama dovoljno postaviti lozinku koju će primatelj koristiti za dekripciju pošte koja mu je poslana. Tu lozinku je potrebno podijeliti s osobom kojoj se želi poslati enkriptirana pošta na način koji pošiljatelju odgovoara. Protonmail omogućuva i dodavanje novih ključeva kao i unos postojećih ključeva što je vidljivo na slici 4.41.



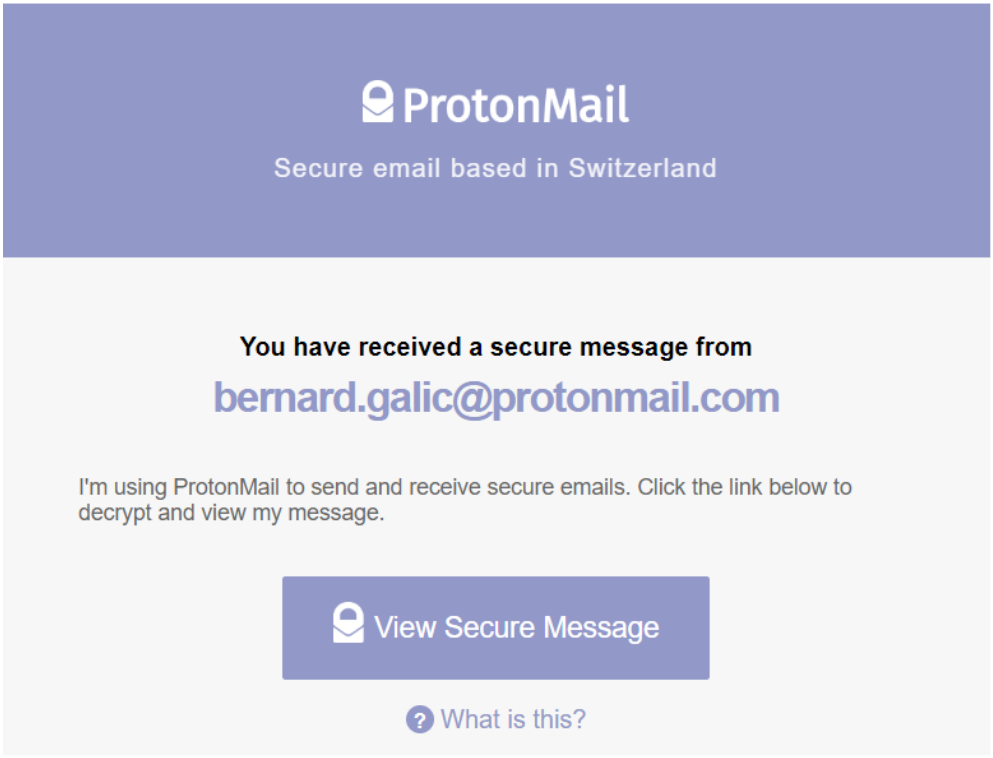
Sl. 4.41. *Prikaz kreiranih ključeva*

U nastavku će biti prikazan primjer enkripcije i dekripcije elektroničke pošte putem ovog klijenta. Na slici 4.42. je prikazan prozor koji se otvara prilikom kreiranja poruke.



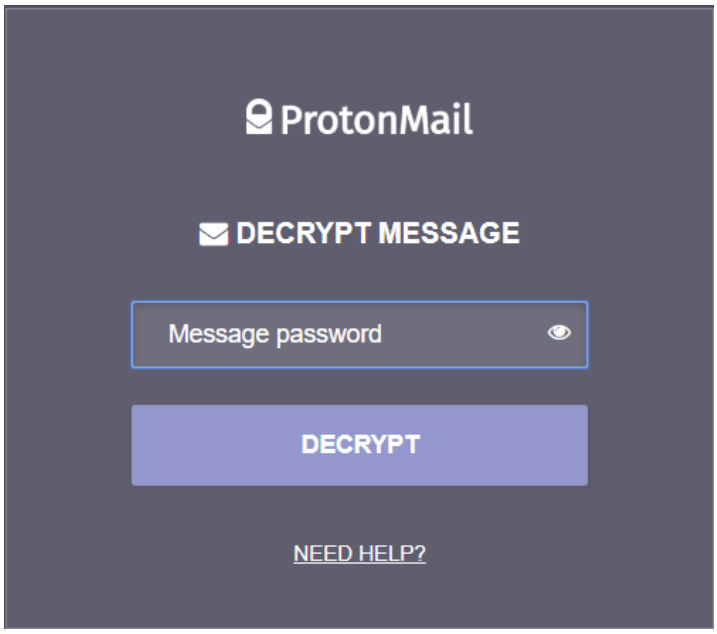
Sl. 4.42. *Slanje enkriptirane poruke*

Kao što je vidljivo na slici 4.42., ProtonMail omogućuje slanje poruke i korisnicima koji nemaju kreiran račun na službenoj stranici ProtonMail-a te je dovoljno samo unijeti primatelja, napisati sadržaj poruke te kliknuti na ikonu lokota čime će se izvršiti enkripcija. Nakon što primatelj primi poruku neovisno u kojem pregledniku elektroničke pošte će mu se prikazati poruka vidljiva na slici 4.43.



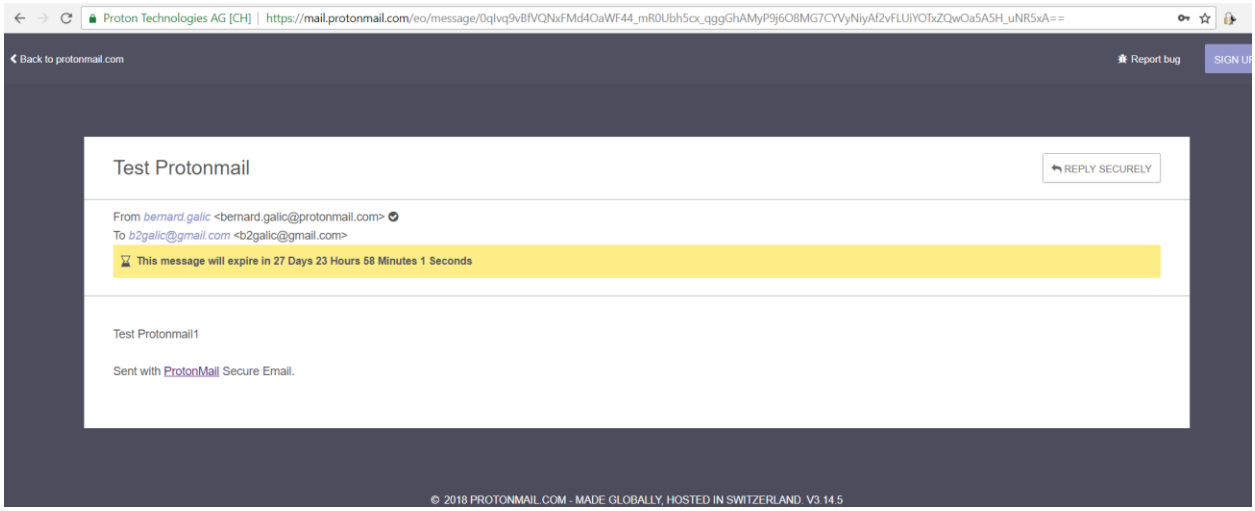
Sl. 4.43. *Primljena poruka korisnika Protonmail-a*

Klikom na „View Secure Message“ korisniku će se otvoriti prikaz vidljiv na slici 4.44. U tome prikazu primatelja enkriptirane pošte će se tražiti da unese lozinku koju je prethodno saznao od pošiljatelja kako bi na siguran način mogao pročitati poruku. Nakon unosa ispravne lozinke, poruka će se dekriptirati te će korisniku biti vidljiv prikaz na slici 4.45.



Sl. 4.44. *Unos lozinke za dekripciju*

Nakon unosa ispravne lozinke primatelju se otvara originalna dekriptirana poruka (Slika 4.45).



Sl. 4.45. *Dekriptirana poruka*

4.5. Usporedba implementiranih metoda za enkripciju

U prethodnim primjerima su prikazane različite metode za enkripciju i dekripciju elektroničke pošte. U svim primjerima je korišten PGP standard, ali svi primjeri također omogućuju i korištenje S/MIME standarda. Princip je gotovo jednak, osim što dobivanje certifikata (para ključeva) kod S/MIME standarda se ne vrši preko „Web of trust“ metode već se certifikat kreira na određenim stranicama koje su pod kontrolom ovlaštenih organizacija za dodjelu certifikata kao što su Comodo te DigiCert. U ovome radu korišten je PGP standard jer u njemu korisnik ima potpunu kontrolu nad kreiranim ključevima dok su kod S/MIME standarda ključevi pod kontrolom ovlaštenih organizacija. Kao prvi primjer je prikazana enkripcija i dekripcija u aplikaciji Kleopatra. Od prikazanih primjera ovaj je najkompliciraniji za korištenje jer zahtjeva preuzimanje i instalaciju software-a na računalo. Nakon instalacije, postupak kreiranja ključeva je prilično jednostavan kao i dodavanje ključeva korisnika s kojima se želi komunicirati. Na primjeru je prikazana enkripcija unutar „Clipboard-a“ aplikacije Kleopatra, međutim ova aplikacija je također kompatibilna sa Microsoft Outlook klijentom. Na sljedećem prikazanom primjeru se radi o klijentu razvijenom od strane Mozzile po nazivu Thunderbird. Nakon besplatnog preuzimanja software-a i instalacije na računalo korisnik može dodati svoje postojeće račune u klijent, a dobra stvar je što istovremeno može biti prijavljen na više računa. Kako bi unutar ovog klijenta bila vršena enkripcija i dekripcija potrebno je instalirati dodatak Enigmail te OpenPGP kao i u prošlom primjeru. Ukoliko unutar OpenPGP-a već postoje kreirani ključevi za određene račune elektroničke pošte Thunderbird će ih prepoznati. Za potrebe testiranja, gore u primjeru su kreirani novi parovi ključeva kako bi se pokazao postupak kreiranja novih i dodavanja postojećih ključeva. Također, u primjeru su korišteni i serveri na kojima se pohranjuju javni ključevi korisnika. Kasnije je javne ključeve drugih korisnika moguće dodati u dodatak Enigmail kako bi se mogla vršiti sigurna komunikacija. Ovaj klijent je izrazito jednostavan za korištenje te omogućuje dobru zaštitu. Na trećem primjeru je korišten dodatak za Gmail pod nazivom Mailvelope. Korištenje ove vrste enkripcije je jednostavnije nego na prethodna 2 primjera iz razloga što se ne mora skidati poseban software na računalo već se nakon instalacije dodatka Mailvelope generiraju ključevi te dodaju ključevi osoba s kojima želimo razmijenjivati enkriptirane poruke. Kao posljednji primjer je prikazan klijent elektroničke pošte pod nazivom ProtonMail. Kako on ne zahtijeva preuzimanje dodataka i posebnog software-a na računalo, djeluje kao najjednostavniji za korištenje. Korisnik mora kreirati račun na službenoj stranici te odabrati lozinku kojom će štititi poruke. Njegovi ključevi su automatski generirani prilikom

kreiranja računa pa tako nije potrebno ručno generirati ključeve kao u prethodnim primjerima. U tablici 4.1. je dan pregled korištenih metoda.

Tab 4.1. *Pregled korištenih metoda*

METODA	POTREBNA INSTALACIJA SOFTWARE-a	KORIŠTENI STANDARD	POTREBNA INSTALACIJA EKSTENZIJA ILI DODATAKA	MOGUĆNOST KORIŠTENJA S/MIME STANDARDA	ENKRIPCIJA DOKUMENATA
Kleopatra	Da	OpenPGP	Ne	Da	Da
Thunderbird	Da	OpenPGP	Da	Da	Da
Mailvelope	Ne	OpenPGP	Da	Ne	Da
Protonmail	Ne	OpenPGP	Ne	Ne	Da

4. ZAKLJUČAK

Tema ovoga rada je bila prikazati prijetnje kojima je izložena elektronička pošta te metode kojima se te prijetnje mogu otkloniti. Na početku rada je ukratko objašnjena sama struktura elektroničke pošte te povijest razvoja zaštite iste. Kako bi se određeni standardi pri zaštiti mogli razumjeti potrebno je znati algoritme na kojima se ti standardi temelje. Iz toga razloga u nastavku su detaljnije objašnjeni najčešće korišteni algoritmi za zaštitu elektroničke pošte zajedno s shemama i matematičkim formulama koje ih opisuju. Upravo ti algoritmi čine temelj danas 2 najkorištenija standarda u zaštiti elektroničke pošte, a radi se o PGP i S/MIME standardu. Većina end-to-end enkripcije se vrši pomoću ova 2 standarda pa je tako detaljno objašnjen svaki od standarda te je dan pregled funkcija koje vrše s pripadajućim algoritmima. Tema ovoga rada je implementirana u sljedećem poglavlju gdje je dan prikaz suvremenih metoda za enkripciju elektroničke pošte. Koristeći različite aplikacije, dodatke i klijente za elektroničku poštu dolazi se do zaključka kako danas postoji velik broj načina na koje korisnik može zaštititi poruke koje šalje. U svim slučajevima je korišten PGP standard kako bi korisnik imao potpunu kontrolu nad kreiranim ključevima, dok bi kod S/MIME standarda ta kontrola bila pod nadzorom ovlaštenih organizacija za dodjelu certifikata (ključeva). Rad je pokazao kako postoje klijenti s već ugrađenim metodama enkripcije koji su najjednostavniji za korištenje,dok postoje i nešto složenije metode zaštite koje podrazumijevaju preuzimanje određenog software-a ili dodataka. Neke od metoda se temeljiti na povjerenju između 2 korisnika o tajnom ključu, kod nekih će tajnost poruka ovisiti o sigurnoj pohrani privatnog ključa, dok će kod nekih tajnost biti u potpunosti u rukama ovlaštenih organizacija. Niti jedna od metoda nije u potpunosti savršena, ali je bitno da korisnik koji želi na siguran način razmijenjivati elektroničku poštu odabere metodu koja mu najviše odgovara te ju dobro prouči uzimajući u obzir sve prednosti i nedostatke koje ta metoda pruža.

LITERATURA

[1] https://techcloud.in/list-of-email-protocols-an-overview (lipanj 2015.)

[2] H.Orman: Encrypted Email: A Brief History of Secure Email (2015.)

[3] Doc.dr.sc Krešimir Grgić, dipl.ing: Sigurnost računalnih sustava,3.predavanje: Asimetrični kriptosustavi (AK.God 2015./2016.), URL:

https://loomen.carnet.hr/pluginfile.php/522291/mod_resource/content/3/SRS%20-%20PR05%20-%20Bezicne%20mreze%20i%20e-mail.pdf

[4] D.Chauhan: RSA and Diffie-Hellman algorithms, URL:

https://www.slideshare.net/daxeshchauhan/rsa-and-diffie-hellman-algorithms-64170629

[5] CARNet Digitalni potpis, CCERT-PUBDOC-2007-02-182, URL:

https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2007-02-182.pdf

[6] CARNet DES algoritam, CCERT-PUBDOC-2003-06-24, URL:

https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2003-06-24.pdf

[7] CARNet IDEA algoritam, CCERT-PUBDOC-2003-06-25, URL:

https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2003-06-25.pdf

[8] Microsoft Docs: Understanding S/MIME (2017.), URL:

https://docs.microsoft.com/en-us/previous-versions/tn-archive/aa995740(v=exchg.65)

[9] Network Associates,Inc.:Introduction to Cryptography: How PGP works (1999.), URL:

https://users.ece.cmu.edu/~adrian/630-f04/PGP-intro.html

[10] Doc.dr.sc Krešimir Grgić, dipl.ing: Sigurnost računalnih sustava, 5.predavanje: Sigurnost elektroničke pošte(AK.God 2015./2016.), URL:

https://loomen.carnet.hr/pluginfile.php/522291/mod_resource/content/3/SRS%20-%20PR05%20-%20Bezicne%20mreze%20i%20e-mail.pdf

[11] CARNet S/MIME standard, CCERT-PUBDOC-2009-05-263, URL:

https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2009-05-263.pdf

SAŽETAK

Postoje mnoge prijetnje što se tiče sigurnosti elektroničke pošte. Iz tog razloga, stručnjaci su razvili razne metode za zaštitu elektroničke pošte. Većina metoda se temelji na PGP i S/MIME standardima. Ovi standardi koriste različite algorimte za različite funkcije koje pružaju. PGP i S/MIME se koriste u većini suvremenih metoda za enkripciju elektroničke pošte kao što su aplikacije, dodatci ili klijenti elektroničke pošte s ugrađenom enkripcijom. Svaka od metoda ima svoje prednosti i nedostatke, a korisnik bi treba pronaći metodu koja mu najviše odgovara te ju dobro proučiti kako bi mogao na siguran način koristiti elektroničku poštu.

Ključne riječi: Sigurnost, elektronička pošta, PGP, S/MIME, enkripcija, metode

ABSTRACT

Email encryption methods

There are many threats in case of E-mail security. Therefore, experts have developed different methods for E-mail protection. Most of these methods are based on PGP and S/MIME standards. These standards use different algorithms for different functions that they provide. PGP and S/MIME are used in most of the modern methods for E-mail encryption such as applications, add-ons or E-mail clients with embedded encryption. Each of these methods have their pros and cons and user should find and learn a method that suits him the most so he can use E-mail in a safe way.

Keywords: Security, E-mail, PGP, S/MIME, encryption, methods

ŽIVOTOPIS

Bernard Galić je rođen 25. prosinca 1993. godine u Osijeku. Osnovnoškolsko obrazovanje završava u OŠ Bilje gdje sudjeluje na brojnim županijskim natjecanjima iz matematike, fizike i geografije. Nakon završetka osnovne škole, 2008. godine upisuje Prirodoslovno-matematičku gimnaziju u Osijeku koju završava 2012. godine s odličnim uspjehom. Nakon toga upisuje Fakultet elektrotehnike, računarstva i informacijskih tehnologija u Osijeku gdje 2015. uspješno završava preddiplomski studij elektrotehnike, smjer Komunikacije i informatika čime stječe zvanje inženjera prvostupnika elektrotehnike. Obrazovanje nastavlja na diplomskom studiju elektrotehnike, smjer Komunikacije i informatika, modul Mrežne tehnologije.