Prolećni semestar, 2017/18

Projektni zadatak:

Bezbednost mejl sistema

Ime: Nemanja Kuzmanović
BrojIndeksa: 2851

Predmet: Zaštita i bezbednost informacija
Šifra predmeta: IT381

Profesor: dr Igor Franc
Asistent: Miljan Marković

Beograd 2018.

Sadržaj

1. Cilj projektnog zadatka

Projektni zadatak analizira razmenu mejlova između dva klijenta preko servera na kom se nalazi podignut hMailServer, gde se prvenstveno wiresharkom hvata saobraćaj koji je potpuno ne obezbeđen i otvoren (ne enkriptovan), da bi se potom prvenstveno uveo PGP (Pretty good privacy) koji će enkriptovati i zaštititi samo telo poruke, međutim korisničko ime i lozinka ostaju i dalje vidljivi kroz analizu saobraćaja (plain text), te je neophodno generisati SSL, self signed, koji će biti instaliran na samom serveru, da bi konekcija tj. Razmena lozinke i korisničkog imena bila potpuno enkriptovana kao i telo same poruke, čime bi se potpuno obezbedila razmena mailova između dva klijenta. Dakle, cilj je podići svest o tome koliko su podrazumevana podešavanja ne bezbedna, i kako obezbediti sam server, kao i klijente, za razmenu poruka sigurnim putem, preko ne bezbedne mreže.

2. Uvod

Kroz projekat je korišćen pop3 protokol (port 110 nebezbedan, port 995 bezbedan), kao i SMTP (port 25 nebezbedan, 465 SSL),

POP ili Post Office Protocol (verzija 3)je protokol koji služi za prijem pošte sa e-mail servera. Na taj način što se sva pošta sa servera preuzima i čuva na lokalnom uređaju i sve radnje koje izvršimo nad primljenim porukama se vrše na našem uređaju.

Neki e-mail servisi koji podržavaju POP3, ukoliko bi sa jednog računara proverili poštu i taj računar preuzeo poruke sa servera, bi izgubio poruke, i ne bi se više nalazile na serveru, pošto ih jednom preuzmemo one se brišu sa samog servera, pa bi naš pristup porukama bio ograničen za taj konkretni uređaj na kome smo obavili proveru pošte.

Međutim, danas, u većini slučajeva možemo izbeći ovu situaciju tako što bi u podešavanjima samog mail klienta izabrali odgovarajuću opciju, kojom bi se kopije poruka i dalje čuvale na samom serveru, pa bi nam bio omogućen pristup i sa ostalih uređaja samim porukama na serveru. Naravno ovde postoji boljka, a sastoji se u tome da će se poruke preuzete na jednom uređaju u pročitane na istom na ostalim uređajima i dalje prikazivati kao nove, tj. nepročitane.

Uz to, podešavanja jednog mail klienta će se čuvati samo na uređaju na kome je klient podešen, kao što će pravljenje foldera, markera za sortiranje i samih filtera, takođe biti vezano za lokalni uređaj, jer POP protokol svu poštu sa servera smešta u inbox.

Takođe, sva nova pošta sa prilozima, koji mogu biti neki fajlovi ili nešto slično što sadrži veličinu fajla kao prilog koji se šalje, će se ponovo preuzimati na lokalni uređaj pri svakoj proveri pošte, što naravno nikako nije praktično, pogotovo ukoliko imamo ograničenu ili sporu internet konekciju.

POP3 protokol koristi port 110 za komunikaciju sa mail serverom. A TCP vezu.

Sesija počinje konektovanjem POP3 klijenta na port 110 POP3 servera, pomoću „3 way handshake“-a, server na to šalje pozdravnu poruku, pa potom sledi razmena komandi i odgovora. Klijent se mora identifikovati komandama **USER**<korisničko ime> i **PASS**<lozinka>. Nakon ovoga se izvršava par komandi, slično kao i kod SMTP, a posle svake sledi odgovor POP3 servera. Sesija se zatvara slanjem QUIT komande od strane POP3 klijenta.

POP3 komande:

USER <name> - korisničko ime za ovaj mail server

PASS <password> - šifra

QUIT – kraj sesije u kojoj server iz faze transakcije prelazi u fazu

ažuriranja

STAT – broj i ukupna veličina svih poruka

LIST <message#> - spisak sadržaja poštanskog sandučeta, uključujući

dužinu svake poruke u posebnom redu

RETR message# - preuzmi obeležene poruke

DELE message# - brisanje obeleženih poruka

UIDL – preuzimanje numeričke liste svih poruka i njihove jedinstvene

identifikacione brojeve, ili jedinsteveni ID za određenu poruku

NOOP - drži konekciju otvorenom

RSET – resetuje poštansko sanduče i vraća izbrisane poruke.

Server koristi dve komande najčesće a to su +OK gde vraća da se zahtevana komanda ili radnja izvršila i da je registrovao kao i +ERR gde vraća da nije uspeo da primi ili registruje odgovarajuću akciju.

Command	Responses	Examples
USER name	+OK name is welcome here -ERR never heard of name	USER David +OK Please enter a password
PASS string	+OK maildrop locked and ready -ERR invalid password -ERR unable to lock maildrop	PASS test +OK valid logon
QUIT	+OK	+OK Server closing connection
STAT	+OK nn mm	STAT +OK 2 320
LIST [msg]	+OK scan listing follows -ERR no such message	LIST +OK 2 messages (320 octets) 1 120 2 200 ... LIST 2 +OK 2 200
RETR msg	+OK message follows -ERR no such message	RETR 1 +OK 120 octets < the POP3 server sends the entire message here >
DELE msg	+OK message deleted -ERR no such message	DELE 2 +OK message deleted
NOOP	+OK no transaction	NOOP +OK
LAST	+OK nn	LAST +OK 2
RSET	+OK	RSET +OK maildrop has 2 messages (320 octets)
Additional Commands		
TOP msg nn	+OK top of msg -ERR	TOP 1 10 +OK < first 10 lines of the header >
RPOP user	+OK -ERR	RPOP david +OK enter password

Slika 1 – primer pop3 komandi

Redosled stanja:

Autorizacija – Kada server izrazi spremnost da prihvata komande, klijent treba da se identifikuje da bi mu se omogućio pristup poštanskom sandučetu.

Transakcija – U ovom stanju klijentu je dozvoljeno da izvrši operacije nad sandučetom. Uobičajene operacije su: preuzimanje spiska poruka, preuzimanje samih poruka, označavanje preuzetih poruka spremnih za brisanje.

Ažuriranje – Kada je klijent završio sa operacijama i pošalje QUIT komandu automatski nastupa ovo stanje. Poruke, u prethodnom koraku, označene za brisanje sada se brišu i TCP veza se prekida.

SMTP (engl. Simple Mail Transfer Protocol) predstavlja osnovni protokol sloja aplikacija za elektronsku poštu, koji koristi uslugu pouzdanog transfera podataka protokola TCP. Kao i većina drugih protokola aplikacijskog sloja ima klijentsku i serversku stranu (koja se izvršava na serveru za elektronsku poštu onoga koji šalje i druga strana koja se izvršava na serveru).

Kada server salje poruku drugim serverima, on preuzima ulogu SMTP klijenta, a kad prima poruke ponaša se kao SMTP server.

Klijentska strana na portu 25 uspostavlja TCP konekciju sa serverskom stranom, ukoliko je ne uspostavi, pokušava ponovo. Nakon uspostavljanja konekcije prelazi se na proces sinhronizacije aplikacijskog sloja, tokom koje SMTP klijent navodi adresu pošiljaoca, posle čega počinje slanje poruke, oslanjajući se na uslugu pouzdanog transfera podataka protokola TCP .

Ukoliko ima još poruka postupak se ponavlja, ukoliko nema, prekida se konekcija.

4. Analiza

4.1 Način analize

Kako bi se analizirala sama bezbednost i razmena mejlova, korišćeni su alati:

VirtualBox, i to, 3 mašine, gde je prva server, a druge sve su simulacija klijenata na mreži, koji preko servera razmenjuju poruke.

Na serverskoj mašini se nalazi hMailServer besplatni softver za podizanje mail servera, kao i wireshark koji će osluškivati sav saobraćaj koji prolazi kroz ovaj server.

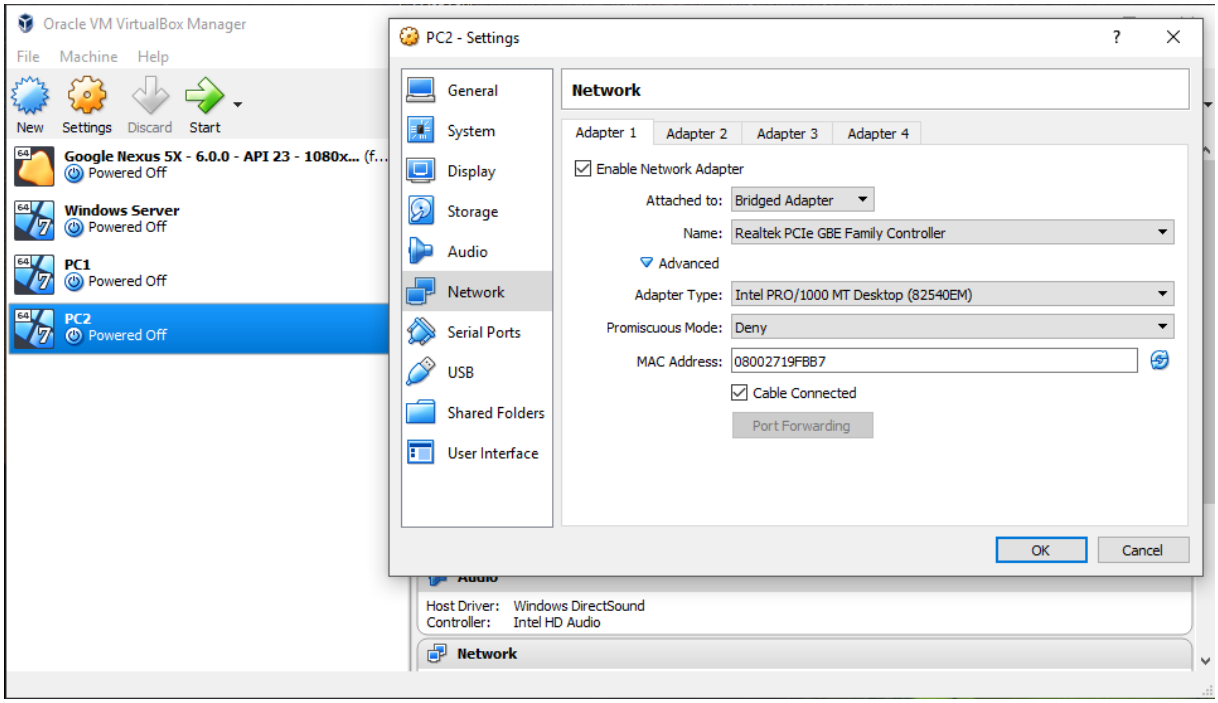
Sve tri virtualne mašine imaju windows 7 sistem podignut na njima.

Druge dve mašine, koje simuliraju klijente, imaju instaliran Thunderbird, PGP4Win, kao I ekstenziju za thunderbird Enigmail.

Dakle, kao što je već napomenuto, ideja je da se na serveru podigne hmailServer a u okviru njega da se konfiguriše domen, dva korisnička naloga, preko kojih će se oni ulogovati preko Thunderbirda, kako bi razmenili mailove, što će kasnije i biti demonstrirano detaljno.

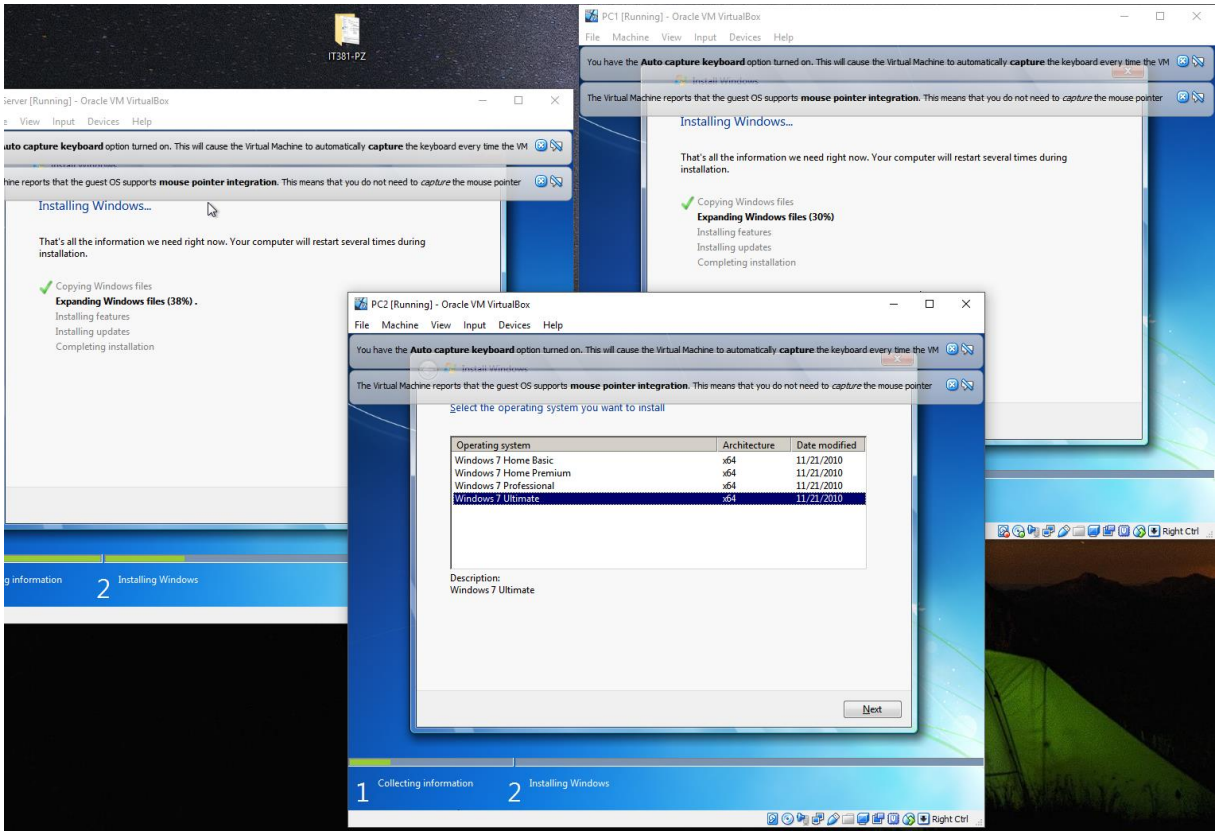
Mašine su podešene u bridged modu, kako bi lokalno između sebe komunicirale, sve imaju 4GB rama (4096 MB), i dodeljena su im 2 procesorska Thread-a.

4.2 Podizanje potrebnih alata



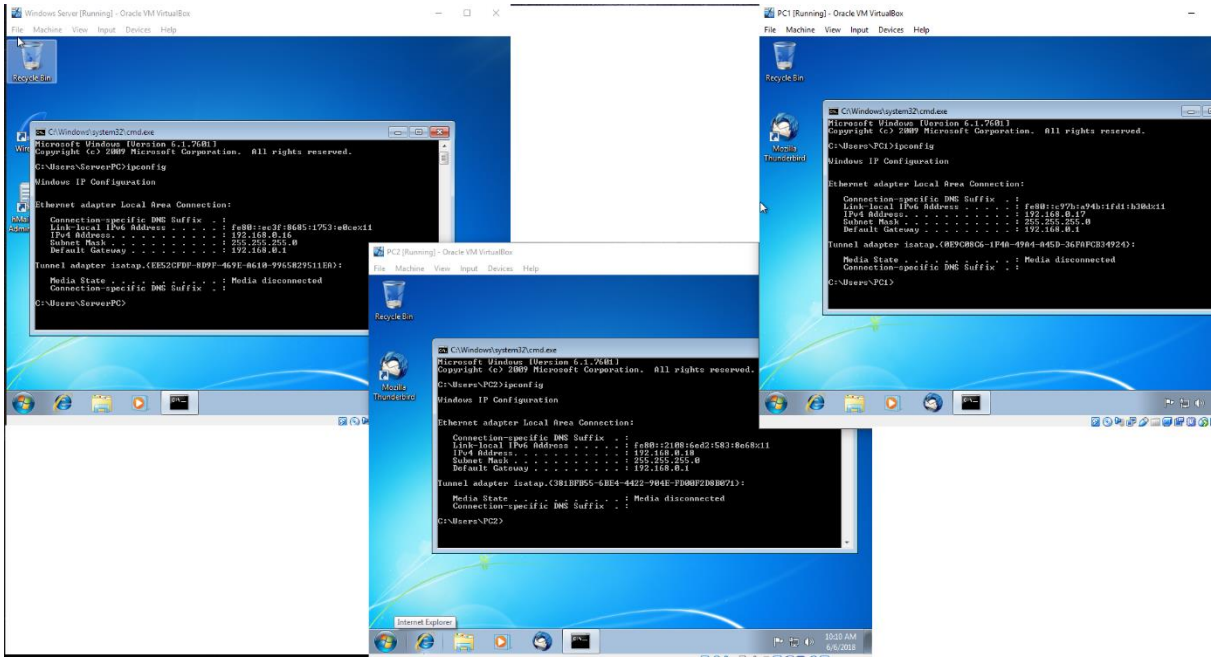
Slika 2 – Virtualna mašina

U besplatnom opensource alatu VirtualBox, instancirane su, dakle, tri mašine, Windows Server, i dve klijentske mašine. Sve mašine su povezane preko Bridged Adaptera. Dodeljeno im je 4096MB rama, kao i 2 procesorska Threada.



Slika 3 – Podizanje mašina

Sve tri mašine imaju Windows 7 OS podignut na njima. Jedina razlika je što je jedna od njih nazvana drugacije i simulira Windows Server, a takođe ima na sebe malo drugacije alate od ostale dve mašine (Wireshark, hMailServer,OpenSSL).



Slika 4 – IP adrese mašina i aplikacije

Server mašina ima ip adresu – 192.168.0.16

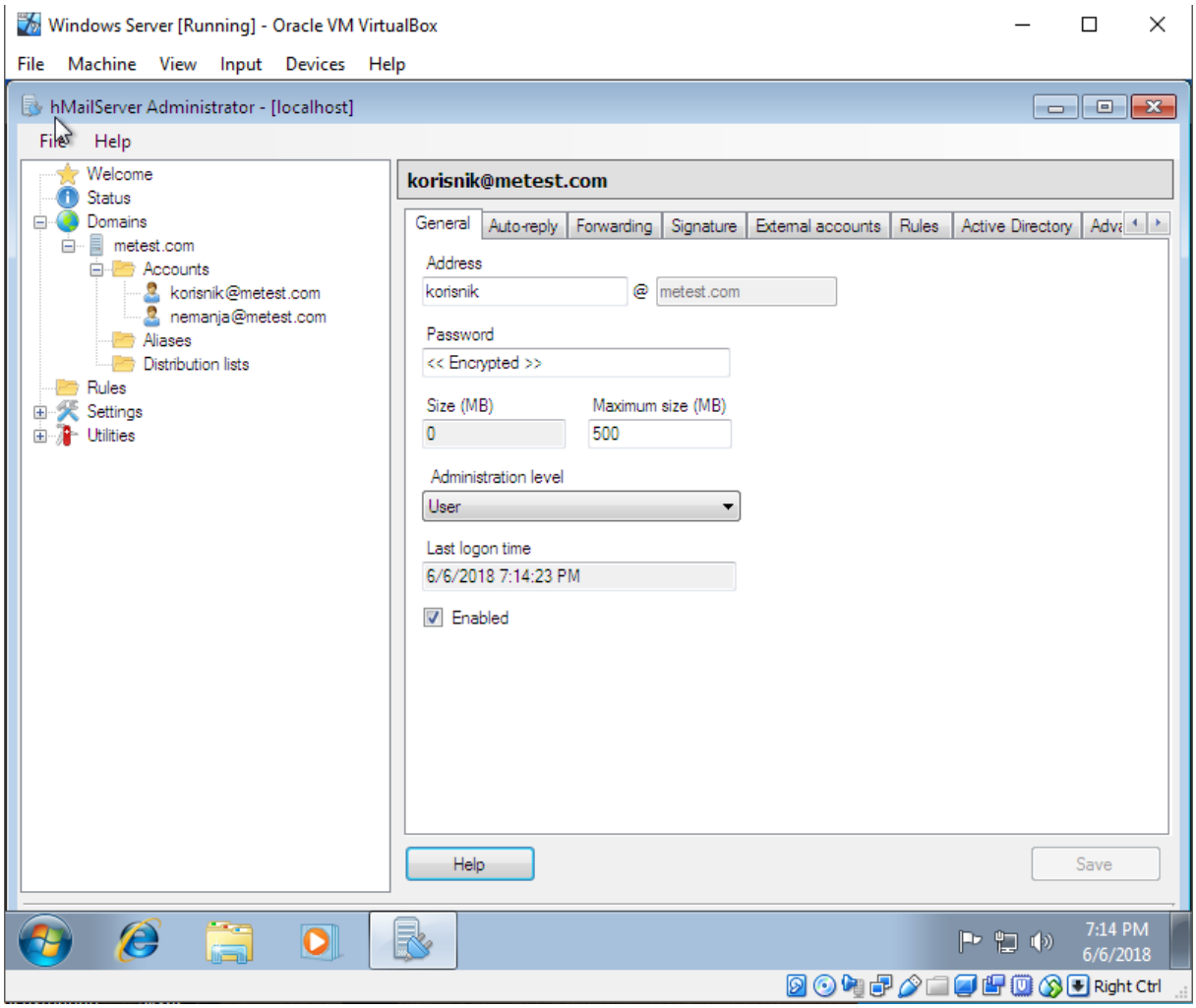
Klijentska mašina sa korisnikom Nemanja Kuzmanovic nemanja@metest.com ima adresu – 192.168.0.17

Klijentska mašina sa korisnikom Korisnik Korisnikovic korisnik@metest.com ima adresu – 192.168.0.18

Što će kasnije ponovo biti i prikazano više puta kroz analizu.

Pored toga se može videti na desktopima, da su na dve mašine instalirani Thunderbird mail klijenti, a na jednoj hMailServer, kao i Wireshark.

4.3 Analiza podrazumevanih (default) podešavanja



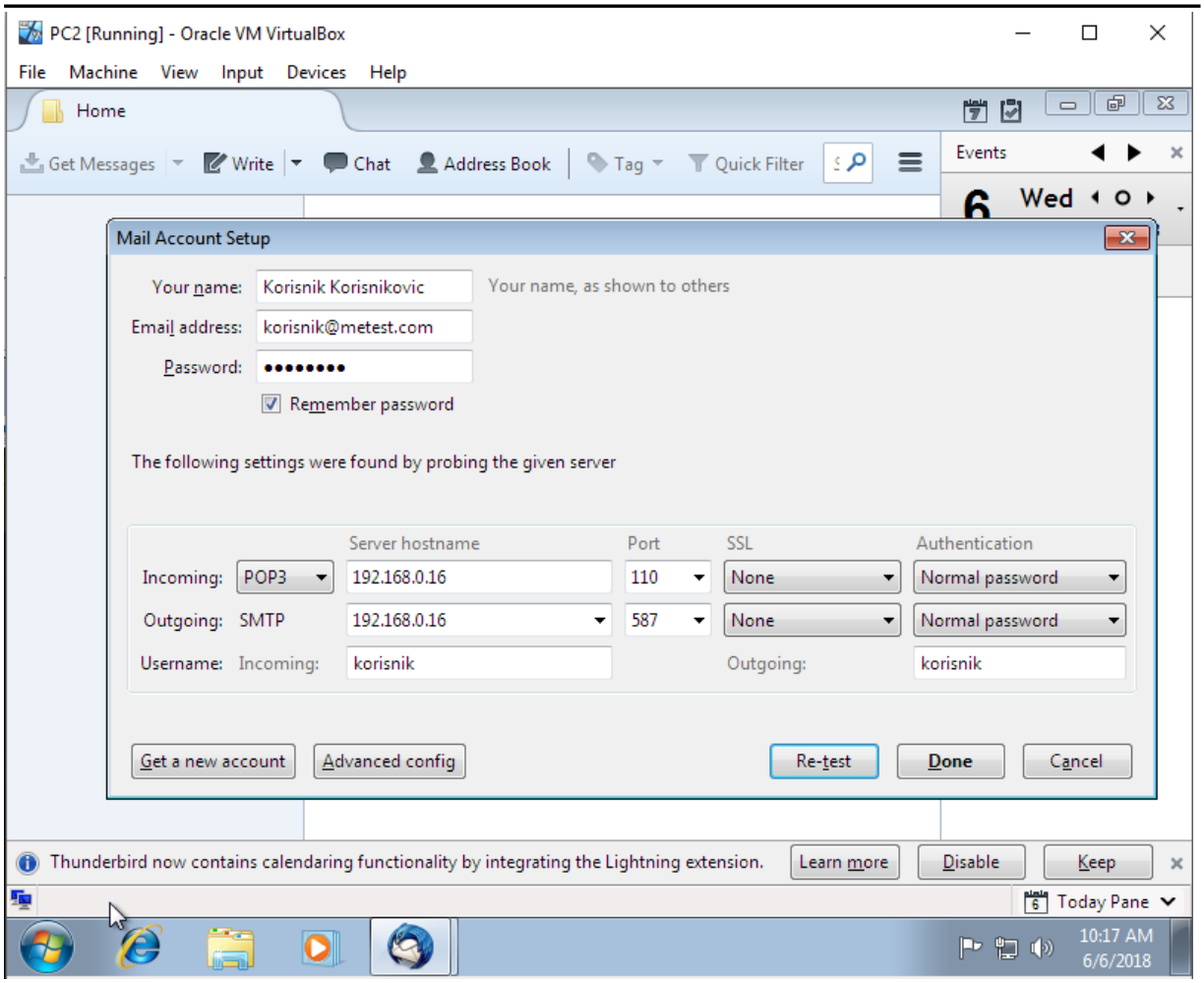
Slika 5 – Podizanje domena i dodavanje naloga

Sledeći korak je instalacija hMailServera, koja je krajnje jednostavna. Potrebno je podesiti i lozinku kojom ćemo pristupadi hMailServer administratorskim alatima za podešavanje i konfiguraciju samog servera.

Nakon toga u delu aplikacije Domains, podižemo novi domen, pod nazivom metest.com (od dve reči Met Test, metest).

Dodajemo dva korisnika:

1. Nemanja Kuzmanovic nemanja@metest.com
2. Korisnik Korisnikovic korisnik@metest.com



Slika 6 – Podešavanje Thunderbird korisničkih naloga

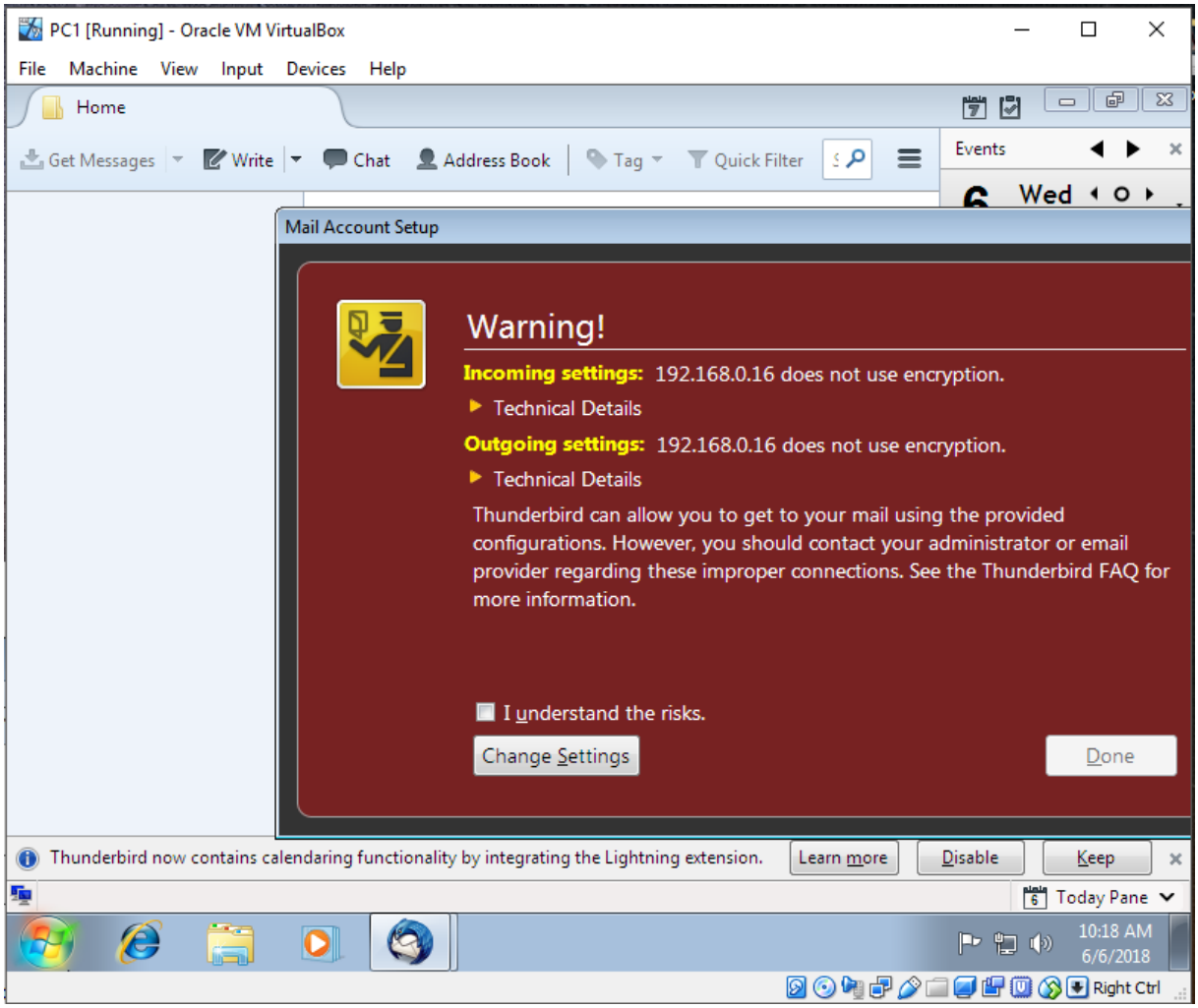
Sada ćemo na obe mašine otvoriti Thunderbird klijenti logovati se na naloge sa našin prethodno definisanim kredencijalima.

NAPOMENA:

Iz razloga što DNS nije podešen tako da obrađuje metest.com domen, te Thunderbird sam pretražuje prvo internet, umesto samog domena, korišćena je IP Adresa Servera koja je 192.168.0.16

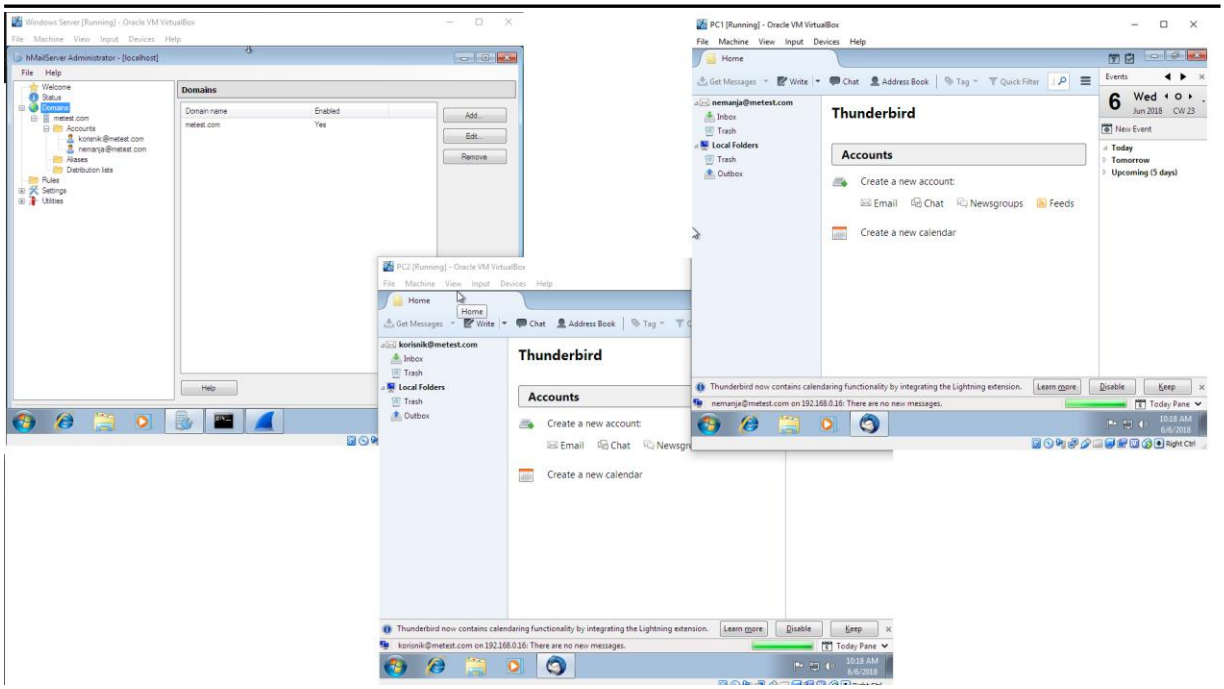
Kao što se na slici može videti Korisnik se loguje preko svoje email adrese i lozinke: korisnik

Biramo POP3 protokol sa Incoming saobraćaj, jer ćemo preko njega preuzimati poštu, a slaćemo preko SMTP.



Slika 7 – Upozorenje

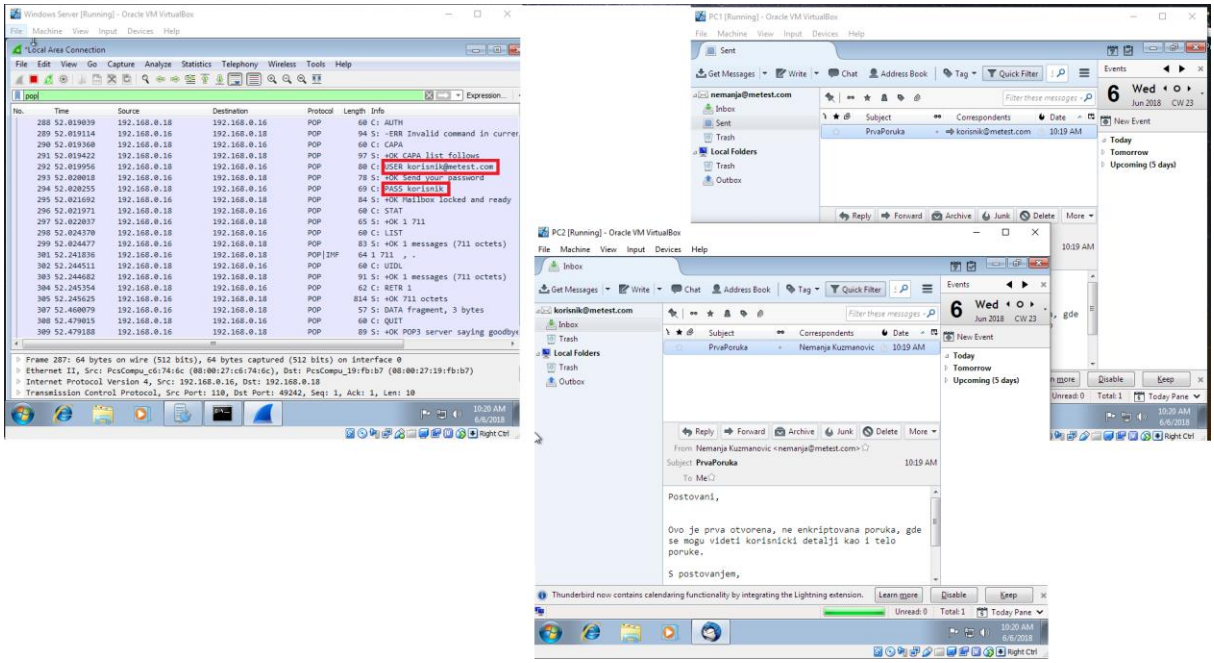
Iz razloga što nemamo odgovarajuće sertifikate na serveru podešene, dobićemo upozorenje da domen ne koristi enkripciju. Idemo na I understand the risks, i zatvaramo prozor na Done. Kasnije će se i sertifikati instalirati.



Slika 8 – Podešeni nalozi

Konačno, imamo podešene naloge na klijentskim mašinama, kao i server, nad kojim je podignut domen, i dva naloga, spremna za razmenu poruka i prvu analizu. Takođe, na server mašini, imamo pokrenut Wireshark, koji osluškuje saobraćaj u pozadini za kasniju analizu.

Na desnoj gornjoj mašini se nalazi nemanja, dok je na donjoj u sredini korisnik.

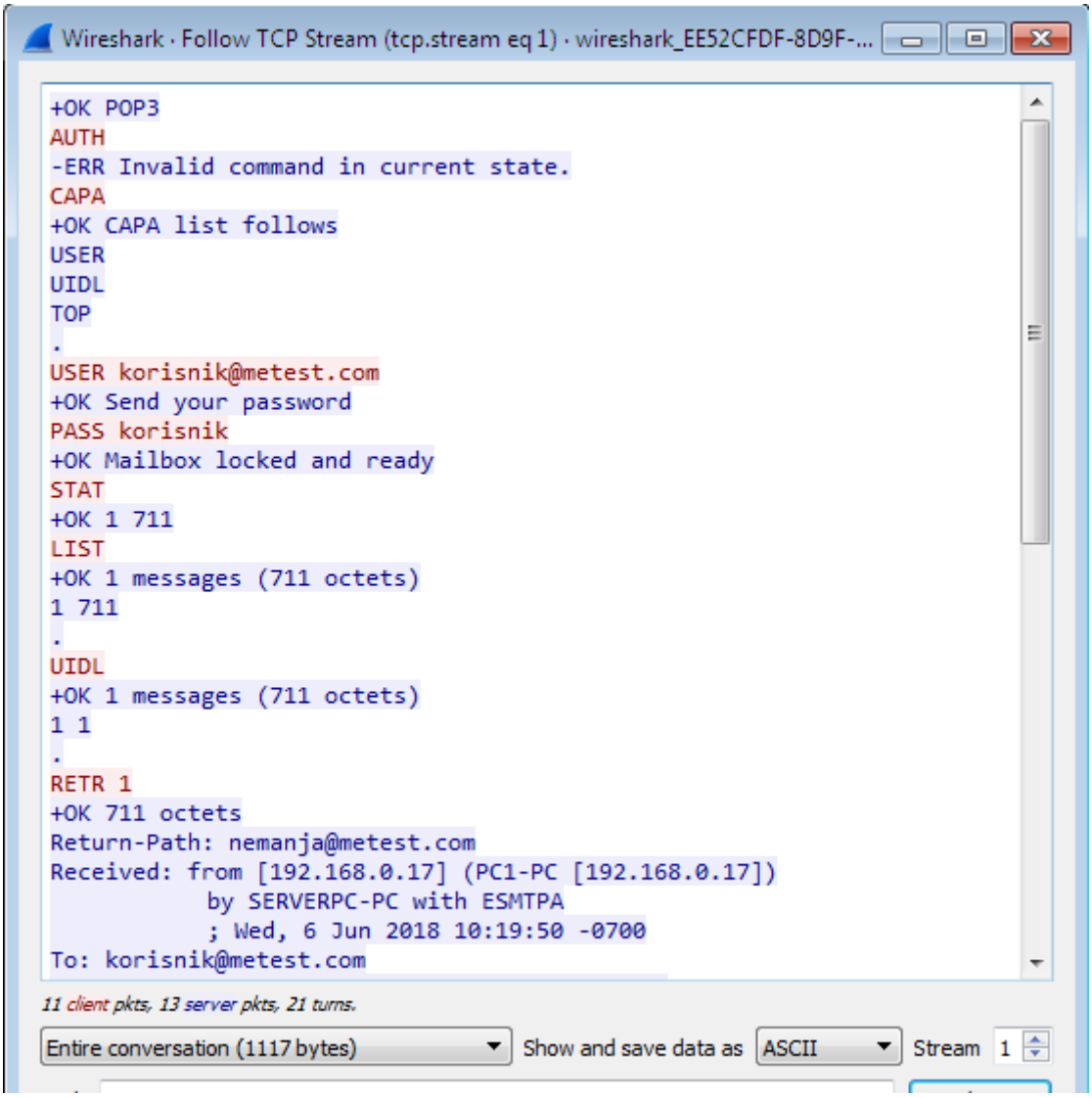


Slika 9 – Prva razmena poruka i analiza

Sa naloza nemanja@metest.com ćemo sastaviti i poslati prvu poruku ka korisnik@metest.com.

U klijentu korisnik-a pritiskamo dugme Get Messages, kako bi preuzeli poruke, što se izvršilo uspešno, i možemo pročitati poruku bez problema.

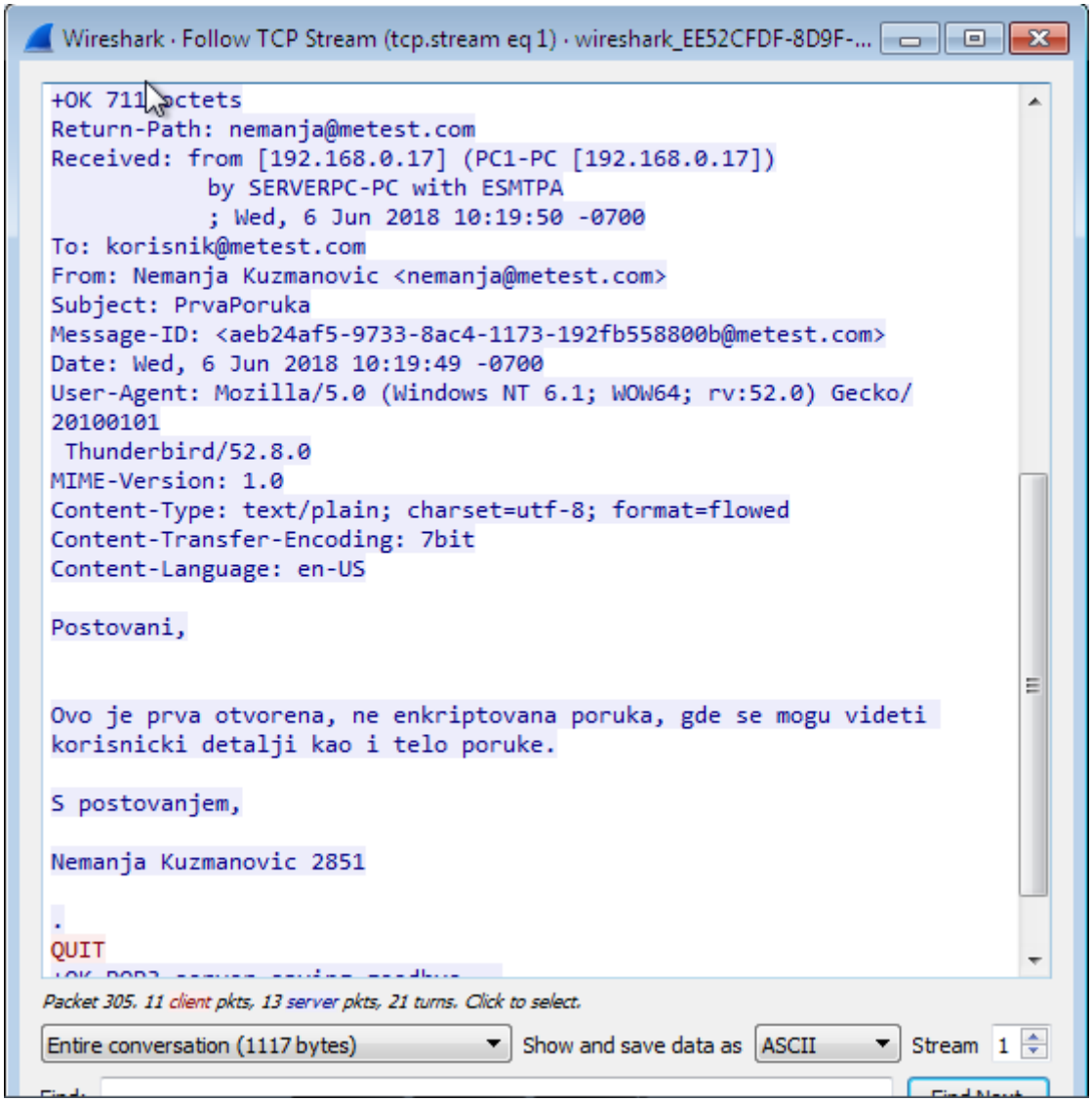
Na serveru, otvaramo Wireshark, postavljamo filter ‘’pop’’ i možemo videti standardne pop3 poruke, kao i ono kritično, a to je USER i PASS poruke, gde se jasno vidi koji korisnik, sa kojom lozinkom preuzima mail. Što je JAKO nebezbedno, ali ćemo u nastavku to popraviti!



Slika 10 – Praćenje TCP streama

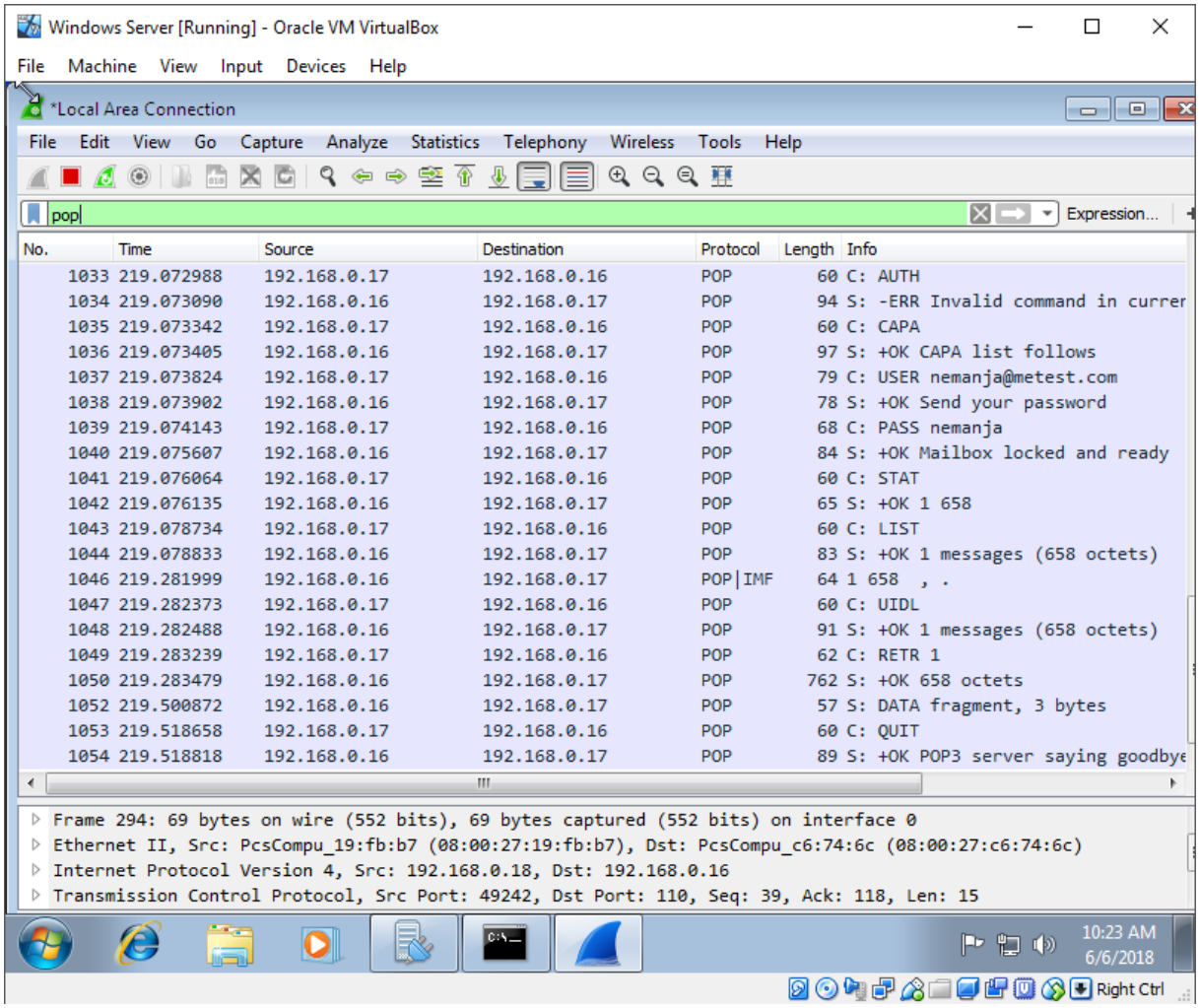
Praćenjem TCP streama, možemo detaljno videti ceo sadržaj maila. Konkretno na slici je prikazan prvi deo, ali će ubrzo biti prikazan i nastavak.

U ovom delu poruke možemo jasno videti sev korisničke detalje koji su kritični, a to su USER i PASS. Takođe, pri dnu se jasno vidi, ko kome šalje poruku, sa kog računara (naziv računara), IP adrese, kada, i kome se šalje poruka.



Slika 11 – Praćenje TCP streama - nastavak

U drugom delu TCP streama, možemo videti nastavak svih mogućih informacija koje se prenose putem nebezbedne mreže. A to podrazumeva, kome se šalje mail, od koga, naslov poruke, kada, sa kog klijenta, i najkritičnije možda, telo poruke, koje je potpuno otvoreno u otvorenom tekstu (Plain text).



Slika 12 – Analiza odgovora

Sada je korisnik@metest.com poslao poruku nazak ka nemanja@metest.com korisniku, a kada je Nemanja pritisnuo na Get Messages u Thunderbird klijentu, na serveru, kroz wireshark možemo jasno videti, da se i njegovi kredencijali prikazuju potpuno otvorenog tipa i neobezbeđeno, što predstavlja jako veliki rizik za same korisnike.

Kada se klijent poveže sa serverom sa kojim želi da komunicira u vidu komandi, tačnije, kada želi da dobije određene informacije oni direktno preko istih tih komandi i komuniciraju.

Na samom početku posle prve „3 way handshake“ konekcije, server vraca +OK poruku kao znak da je prepoznao određeni oređaj koji zahteva konekciju sa njim.

Posle uvodne konekcije javlja se komanda CAPA koja pita da li server ima (capability) sposobnost da porčita UIDL komandu,za preuzimanje numeričke liste svih poruka i njihove jedinstvene identifikacione brojeve, ili jedinsteveni ID za određenu poruku. Što u sledećem koraku server i potvrđuje sa +OK POP3 komandom, u suprotnom najverovatnije bi vratio – ERR kao naznaku na grešku. UIDL je bitan za samog korisnika kako bi sigurnije, efikasnije i bolje identifikovao pojedinačne poruke.

Zatim, klijent ponovo šalje serveru još nekoliko zahteva za proveru komandi, kako bi bio znao koje parametre kasnije prosleđuje.

Server potom odgovara, i potvrđuje da ima sve sposobnosti koje klijent zahteva, i šalje odjavnu pozdravnu poruku „Goodbye“.

Sledeći korak, kada se korisnik preko klienta uloguje, je da se proveri njegova autektikacija, da li se njegove informacije nalaze u bazi servera. Međutim u ovom primeru se desava nešto zanimljivo, u najmanju ruku, naime, server nema mogućnost autentikacije, iz razloga što nije konfigurisan za takvu radnju, postavljena je osnova radi analize, koja će se u nastavku više obezbediti, i uporediti razlike i mane ovako laički konfigurisanog servera.

Još jednom korisnik proverava sposobnosti servera za određene komande posle odbačaja autentikacije.

Napomena:

Server jeste registrovao korisnika i posle provere u bazi mu omogućio da se autentikuje kao autentični korisnik koji ima mogućnost da komunicira sa serverom, međutim ova autentikacija je ili slabo konfigurisana od samog mail servera, ili od samog klienta alata preko koga koristimo usluge servera. Trenutno nisu postavljeni nikakvi vidovi ni autentikacije ni enkripcije, što je vrlo verovatno da je klient očekivao.

Nakon ovoga, klient šalje korisnikov „USER“ na sta server odgovara da je našao odgovarajuće poklapanje, usled dolazi do slanja „PASS“ takođe od strane klienta koji pokušava da pristupi serveru. Prvenstveno dolazi do –ERR poruke, koja je u ovom slučaju vraćena iz razloga što se u postavkama samog klienta nije navelo puno korisničko ime nego samo deo, što ne predstavlja grešku nego server upozorava da se u bazi nalaze još neki podatci koji nisu dopunjeni, ali da su oni primarni +OK.

Sada nailazimo na veliki problem. Naime, USER od strane klijenta je poslat i ukoliko se desi da se na putu klienta i servera nađe maliciozan korisnik ili korisnik koji osluškuje pakete, on će mogu da pristupi samom sadržaju USER-a i to više nego očigledno. Još veći problem je što klient odmah posle šalje i otvorenu šifru bez ikakve enkripcije ili pokušaja da se sakrije, nego kao običan tekst, koji je opet, više nego očigledan. Ovako loše konfigurisan klient koji ne šalje enkriptovanu šifru kao i server koji ne zahteva enkriptovanu šifru, moze lako biti upotrebljen u maliciozne radnje, do kojih najčešće dolazi dobro poznatom metodom pribavljanja ovakvih informacija nazvanom „Man in the middle“. Gde se uređaj na putu od klienta do servera „ni kriv ni dužan“ nalazi i dobija jako senzitivne informacije.

Takođe, kada se osnovni koraci autentikacije, konekcije, i provere korisnika završe, oni se ponavljaju na nekom vremenskom intervalu, kako bi server znao da je klient još uvek tu i da treba da zadrži trenutnu sesiju, jer ukoliko se ne bi proveravalo i sesija se držala non stop, definitivno bi jako brzo došlo do preopterećenja samog servera pa mu niko ne bi mogao pristupiti dok se ne bi očistila radna memorija, što nikako nema smisla nikada raditi.

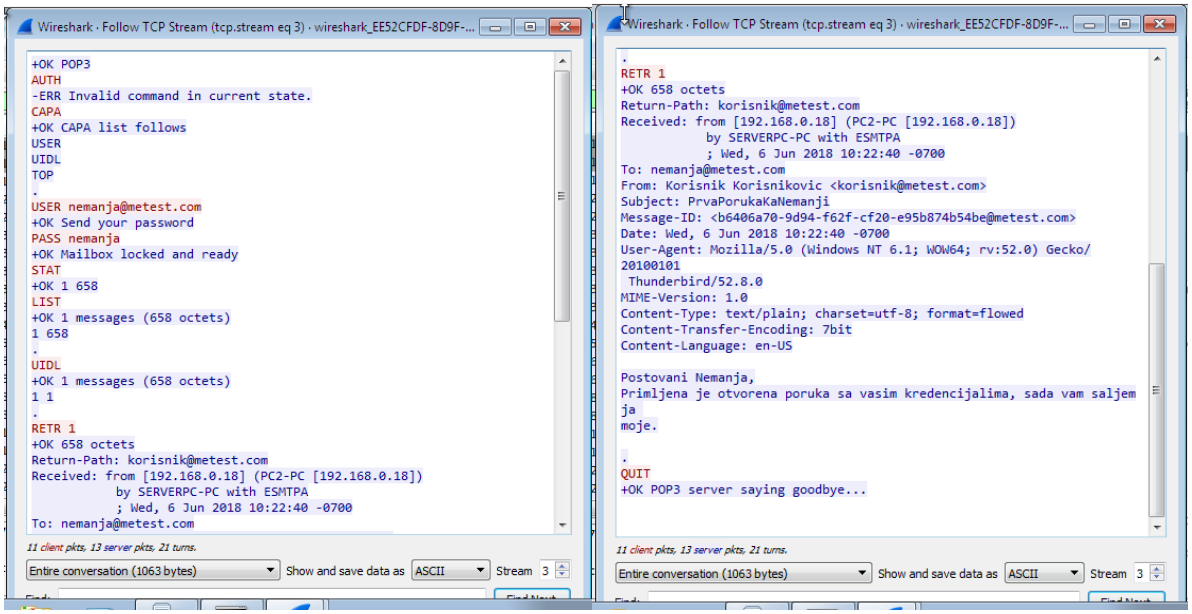
Klient šalje serveru komandu STAT u kojoj traži status svoga sandučeta, da li ima i koliki broj poruka na serveru.

Server dalje prihvata ovu komandu odgovara sa +OK i prosleđuje broj poruka koje čuva za određenog klijenta kao i veličinu tih poruka.

U ovom slučaju nalazi se jedna testna poruka, veličine 658 okteta.

Dakle, kada korisnik pritisne određeno „Get Messages“ dugme, klient šalje komandu stat u kome server vraća broj poruka, nakon čega klient zahteva UIDL, kako bi dobio jedinstvene indentifikatore svake poruke posebno, da bi posle komande klienta RETR server klientu poslao čitavo telo poruke, kao i informacije koje sadrži o njoj.

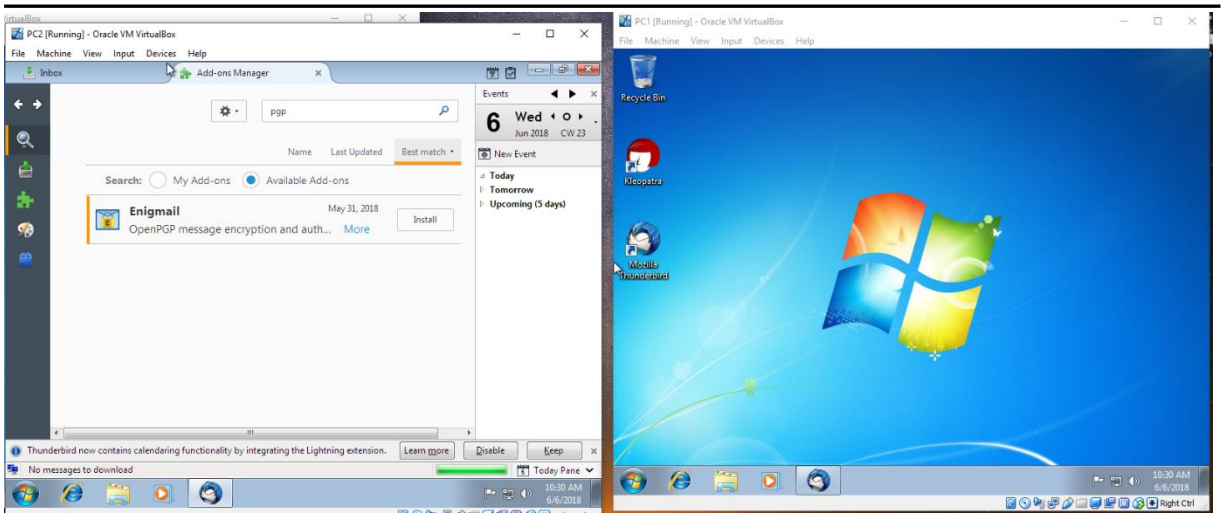
Na primeru se vide 1 RETR komande, iz razloga koji je naveden a to je da je jedna testna poruka poslata. Kada dođe do završetka primanja ponovo se šalje QUIT za prekid sesije i server se odjavljuje.



Slika 13 – Praćenje TCP stream odgovora

Takođe, kao i u prvom primeru, ako zapratimo TCP stream, možemo detaljno videti potpuno otvoren ceo sadržaj poruke koja se poslala sa svim detaljima, otvorenog tipa.

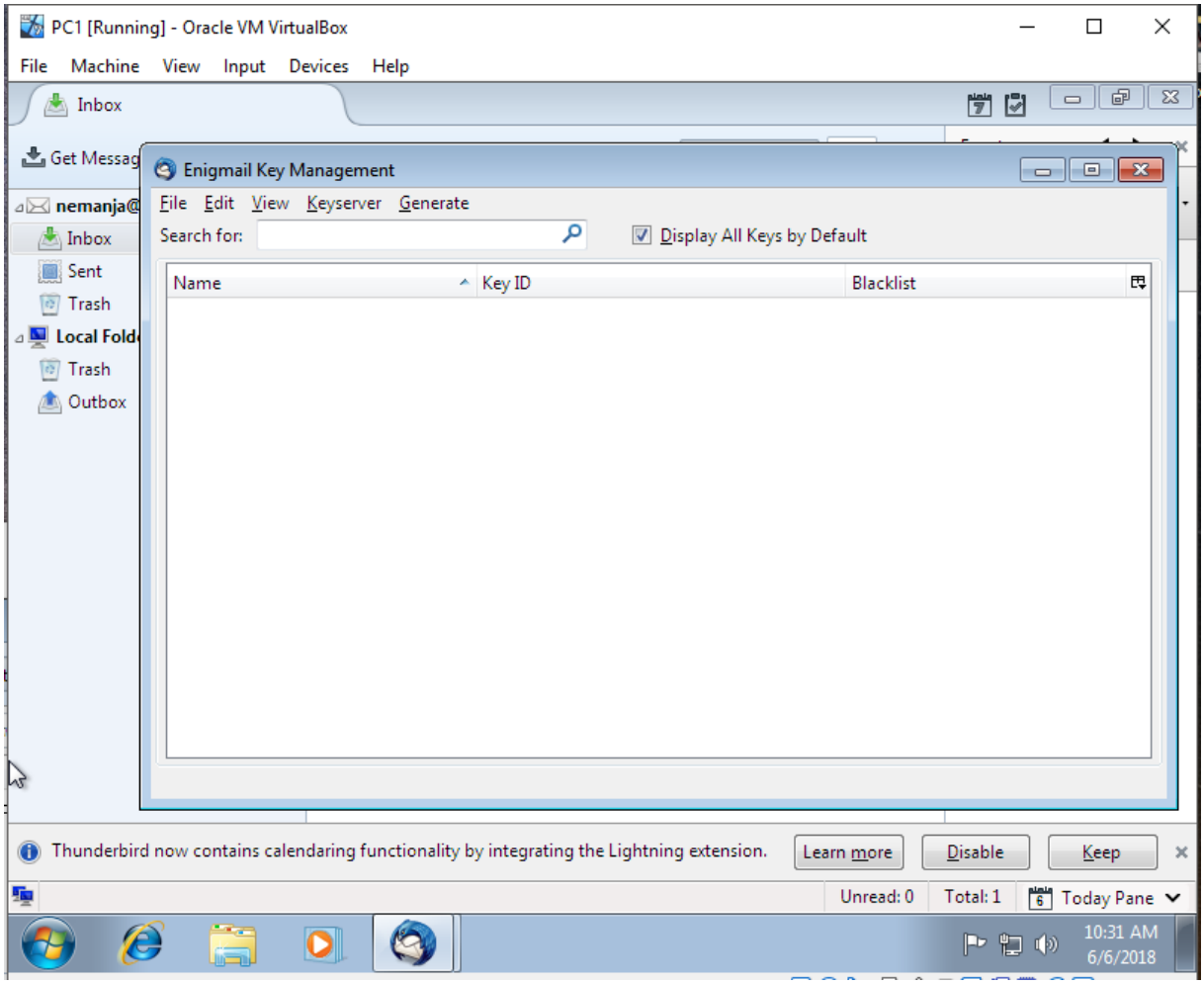
4.4 PGP (Pretty Good Privacy) Zaštita



Slika 14 – PGP

PGP predstavlja program za enkripciju kriptografskim algoritmina kojim će se obezbediti da telo poruke stigne enkriptovano na destinaciju. Instaliramo PGP4Win, koji ima svoj klijent Kleopatra, a potom instaliramo u samom klijentu Thunderbird, ADDON pod nazivom Enigmail, koji će nam omogućiti manipulaciju PGPom kroz sam email klijent. Generisanje para ključeva, kao i slanje javnog ključa korisniku s kim želimo enkriptovanu razmenu poruka.

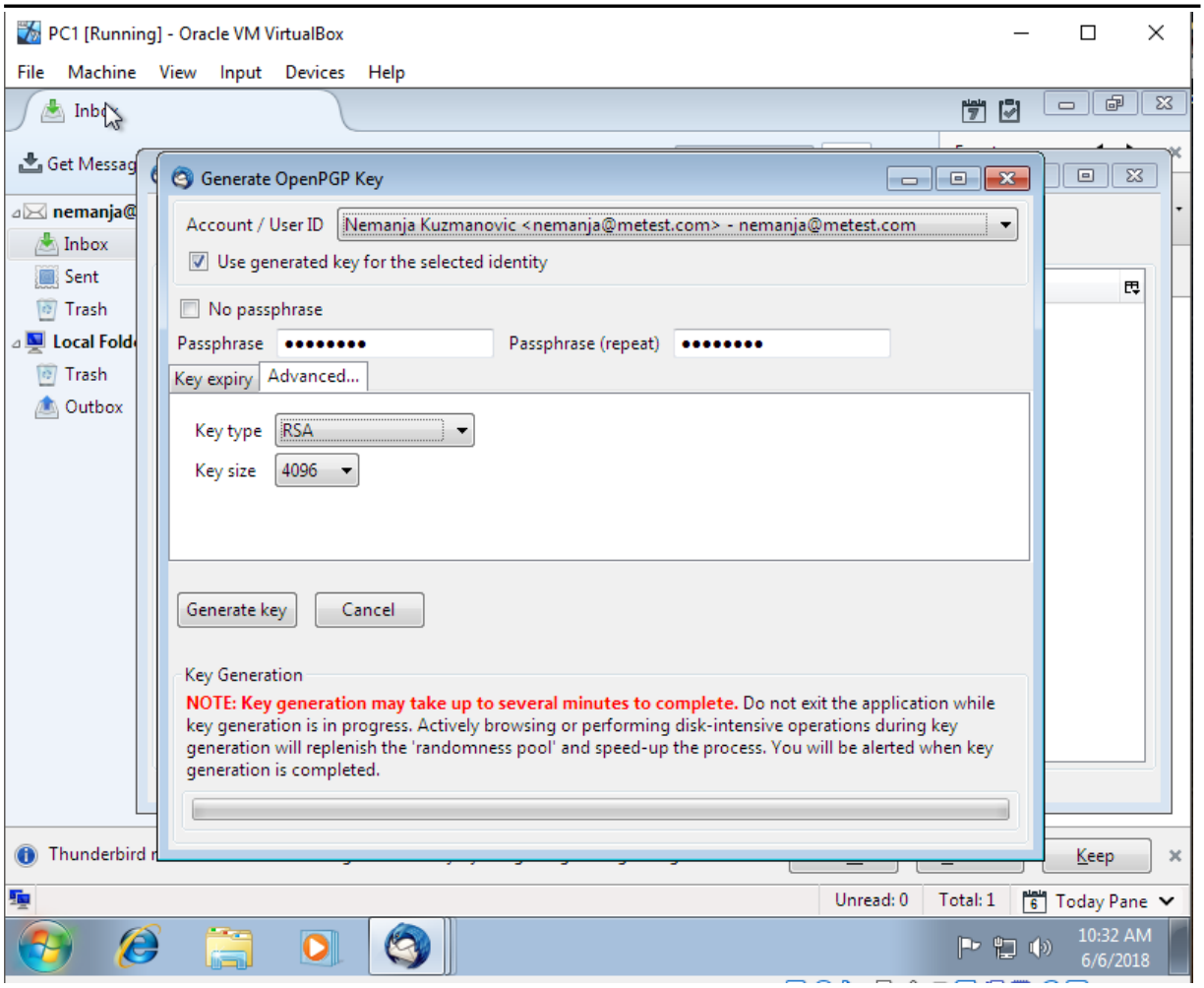
A ta razmena poruka se obavlja tako što svaki klijent generiše par privatnog i javnog ključa, gde javni pošalje onome s kim želi da komunicira, dok privatni ključ ostaje kod samog korisnika, koji ih je i generisao. Kada drugi korisnik dobije javni ključ, on piše poruku i enkriptuje samu poruku tim javnim ključem, te je šalje korisniku od koga je dobio javni ključ. Kada korisnik dobije poruku sa javnim ključem, koristi svoj privatni ključ kako bi dekriptovao poruku i video kompletan sadržaj iste. Komunikacija se tako odvija u oba smera.



Slika 15 – PGP Key Managment

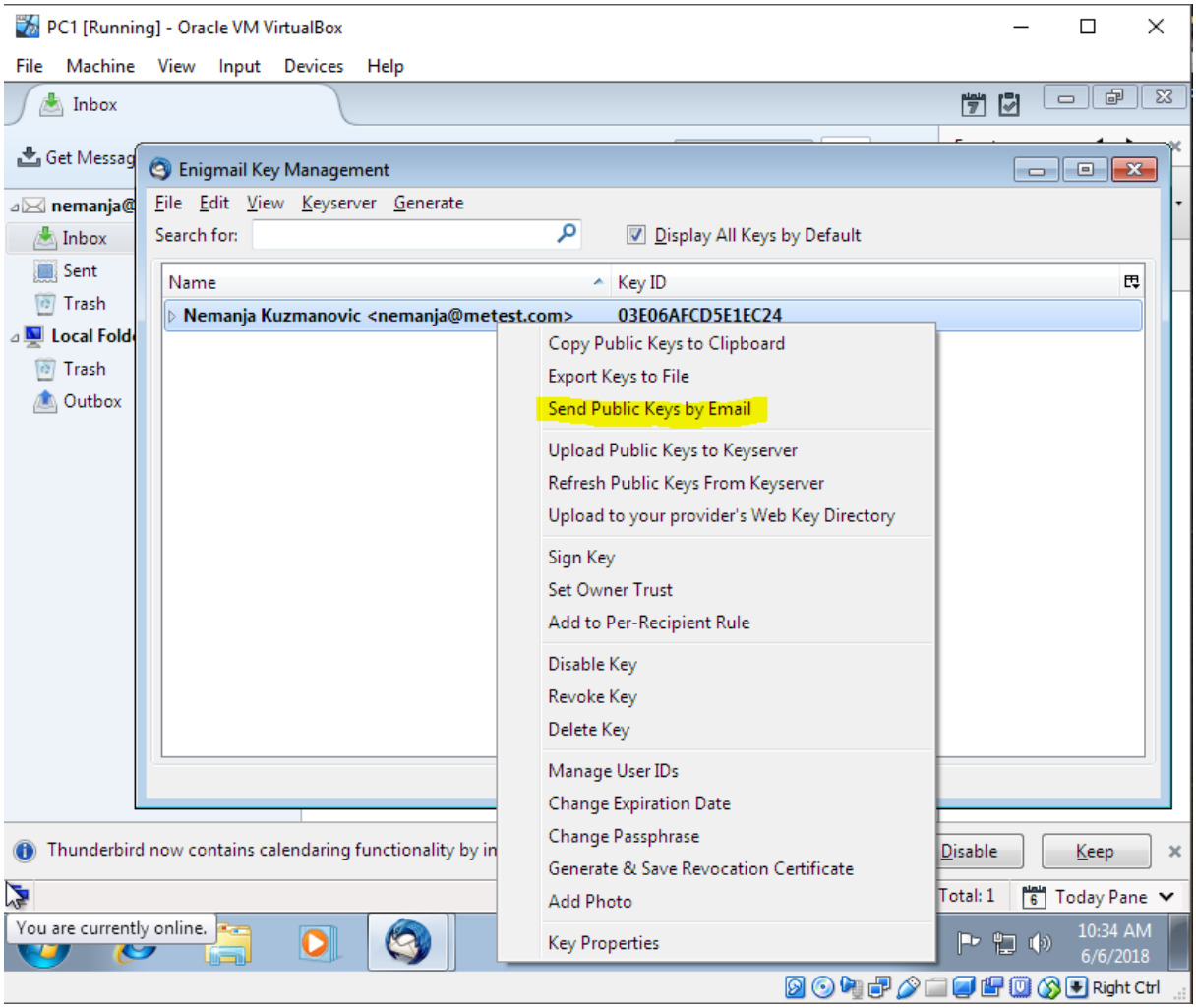
Kada odemo u klijentu na Addons, dobićemo novi nastavak instaliran koji se zove Enigmail, proširenjem njega, dobijamo za sada jedinu opciju koja se zove Enigmail Key Managment. Preko ovog alata ili prozora, možemo generisati par ključeva, poslati javni ključ, obrisati stari, i još gomila drugih opcija.

Za početak, svaki korisnik treba da generiše par ključeva, javni-tajni, i putem mejla, da pošalje drugom korisniku javni ključ, kako bi kasnije komunicirali tajno. Pošto javni ključ mogu da vide svi, i nije nikakva tajna, ovo se smatra bezbednim, da se pošalje kroz nebezbednu mrežu.



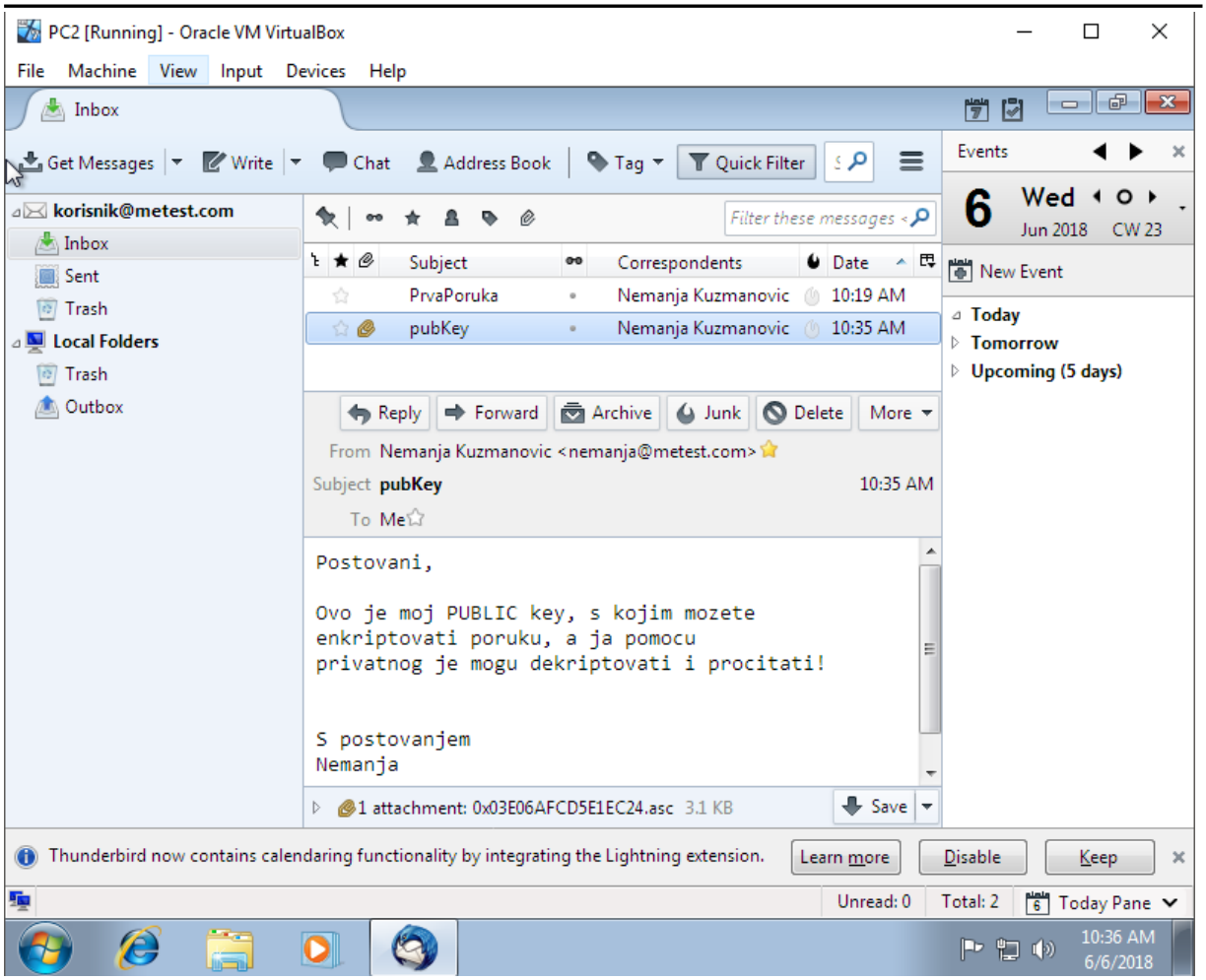
Slika 16 – PGP Generisanje para ključa

Iz padajućeg menija se bira Generate pa New Pair Key. Sada se otvorio prozor kao na slici. Biramo za koji nalog, koji je povezan sa mejl klijentom, želimo da generišemo par ključeva. U ovom slučaju biramo korsinika Nemanja Kuzmanovic. Kucamo šifru dva puta, koja se mora poklapati, i biti što komplikovanija, i zadovoljavati preporučenu šifru od minimalno 8 karaktera sa 2 slova 2 broja 2 znaka, velikim i malim slovom... Potom, biramo RSA i 4096 veličinu ključa, koja je podrazumevana vrednost. Konačno, idemo na Generate Key, i čekamo da se isti izgenerišu.



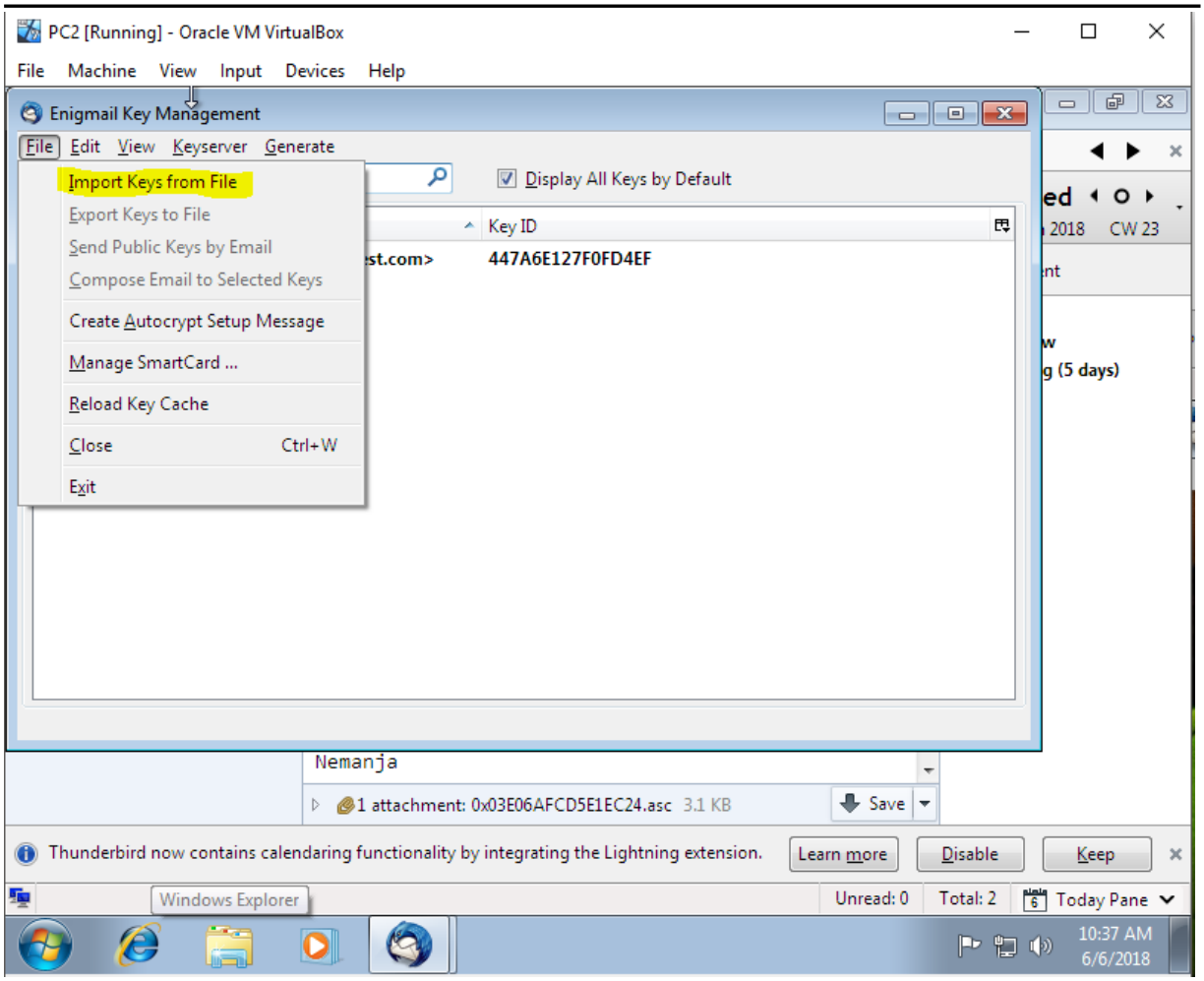
Slika 17 – PGP Slanje javnog ključa

Sada se sa korisničkog naloga nemanja@metest.com šalje Public (javni) ključ, korisničkom nalogu korisnik@metest.com .



Slika 18 – PGP Primanje javnog ključa

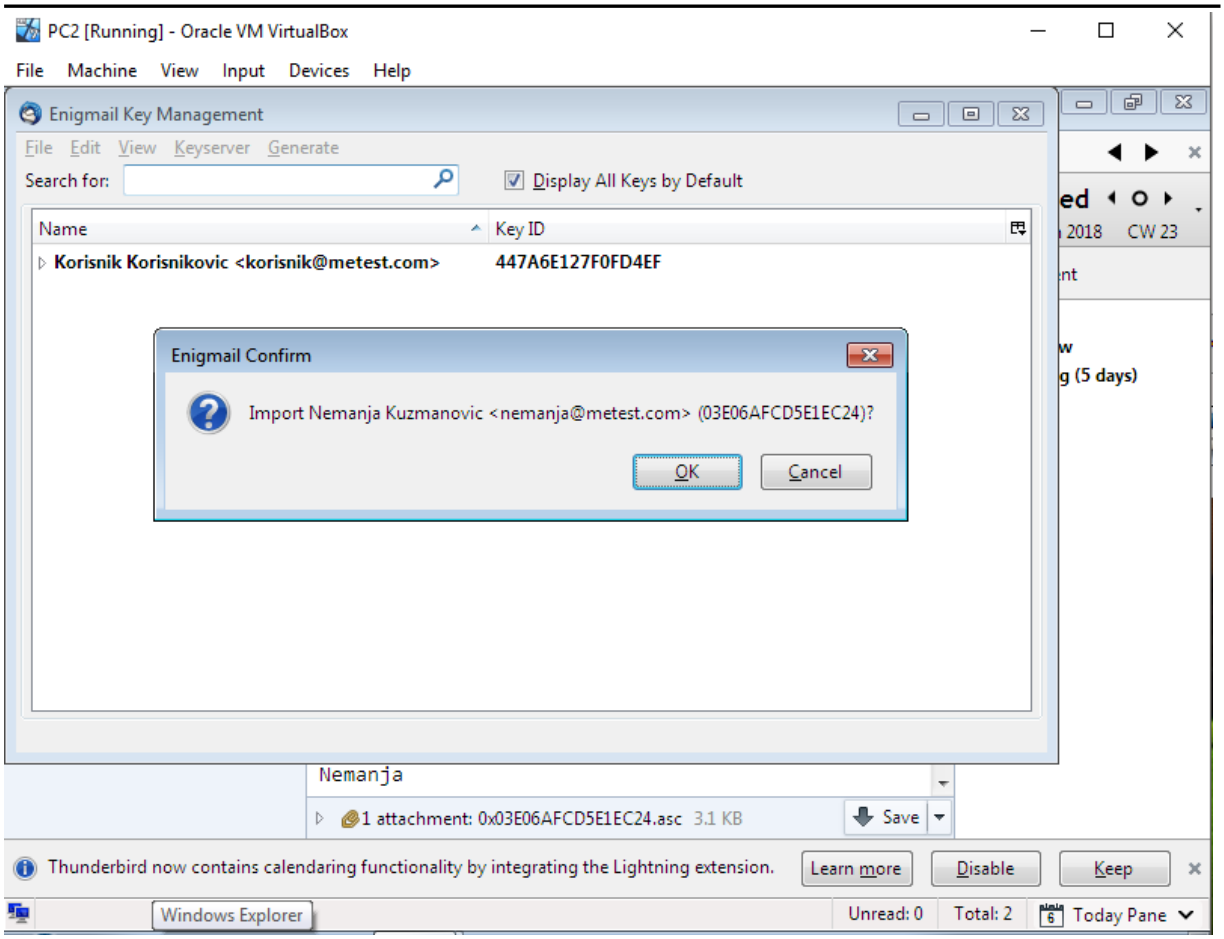
Kada Korisnik Korisnikovic primi poruku, dobiće u prilogu (attachmentu) fajl sa ekstenzijom asc, u kome se nalazi javni ključ osobe sa kojom želi da razmeni enkriptovanu poruku. Potrebno je da korisnik importuje taj ključ preko Key Managment-a, kako bi enkripcija radila.



Slika 19 – PGP Uvezivanje javnog ključa

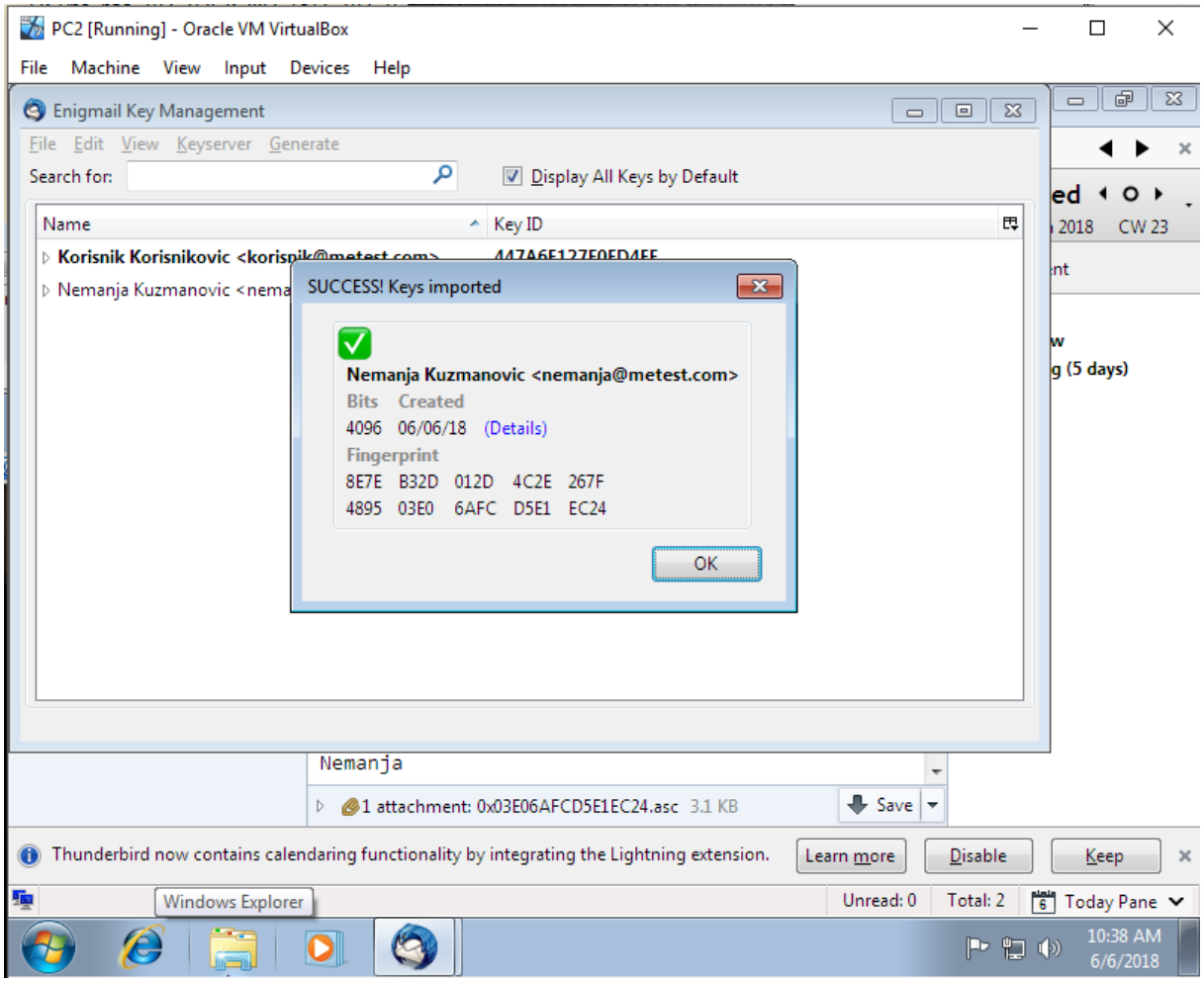
Javni ključ smo iz mejla sačuvali na računar. Otvaramo Key Managment, i idemo File Import Keys from file.

Biramo javni ključ sa destinacije, i idemo na open.



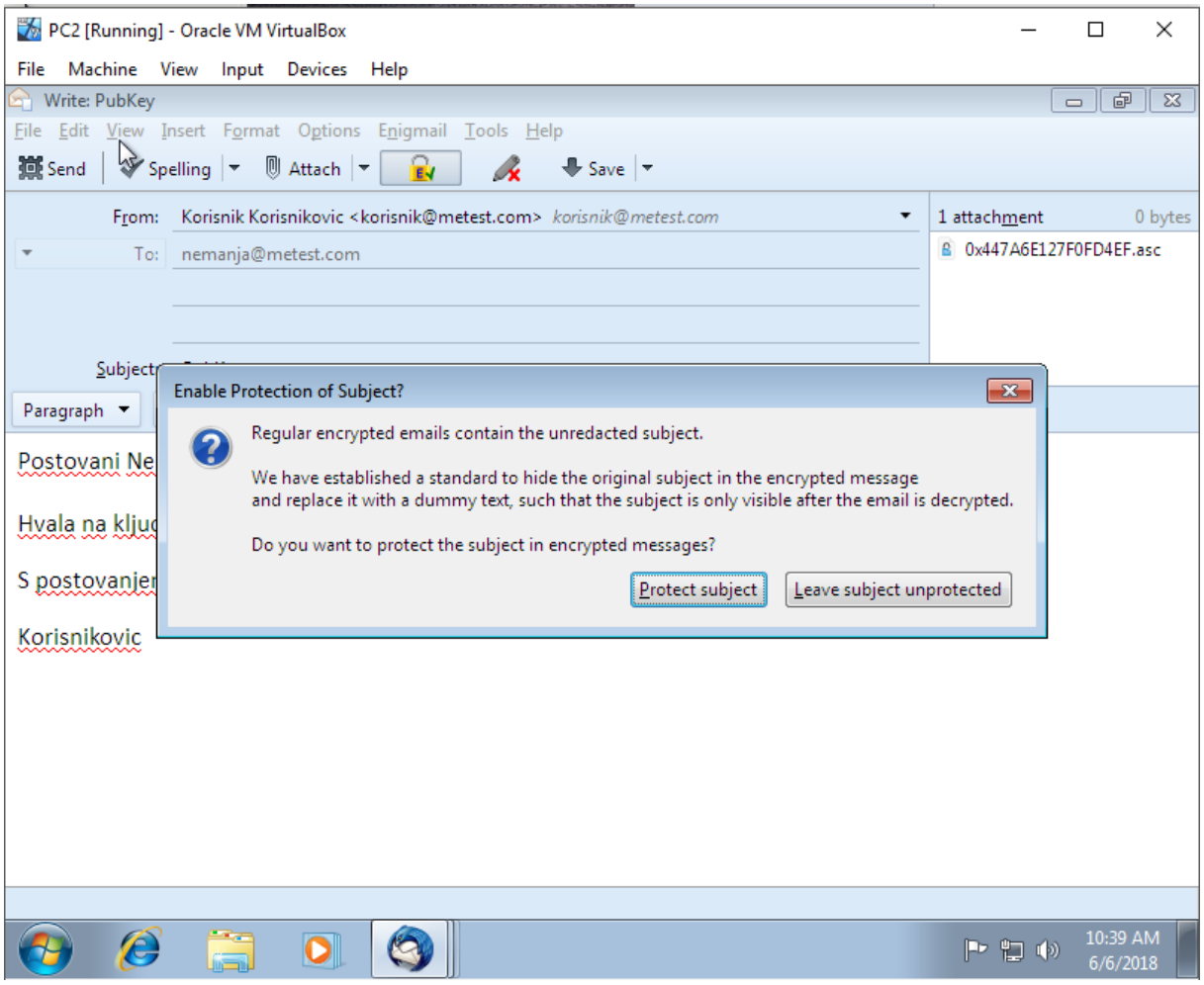
Slika 20 – PGP Uvezivanje javnog ključa - nastavak

Korisnik sada dobija poruku, da li želi da importuje ključ od korisnika Nemanja Kuzmanovic, što ćemo naravno potvrditi.



Slika 21 – PGP Uvezivanje javnog ključa uspešno

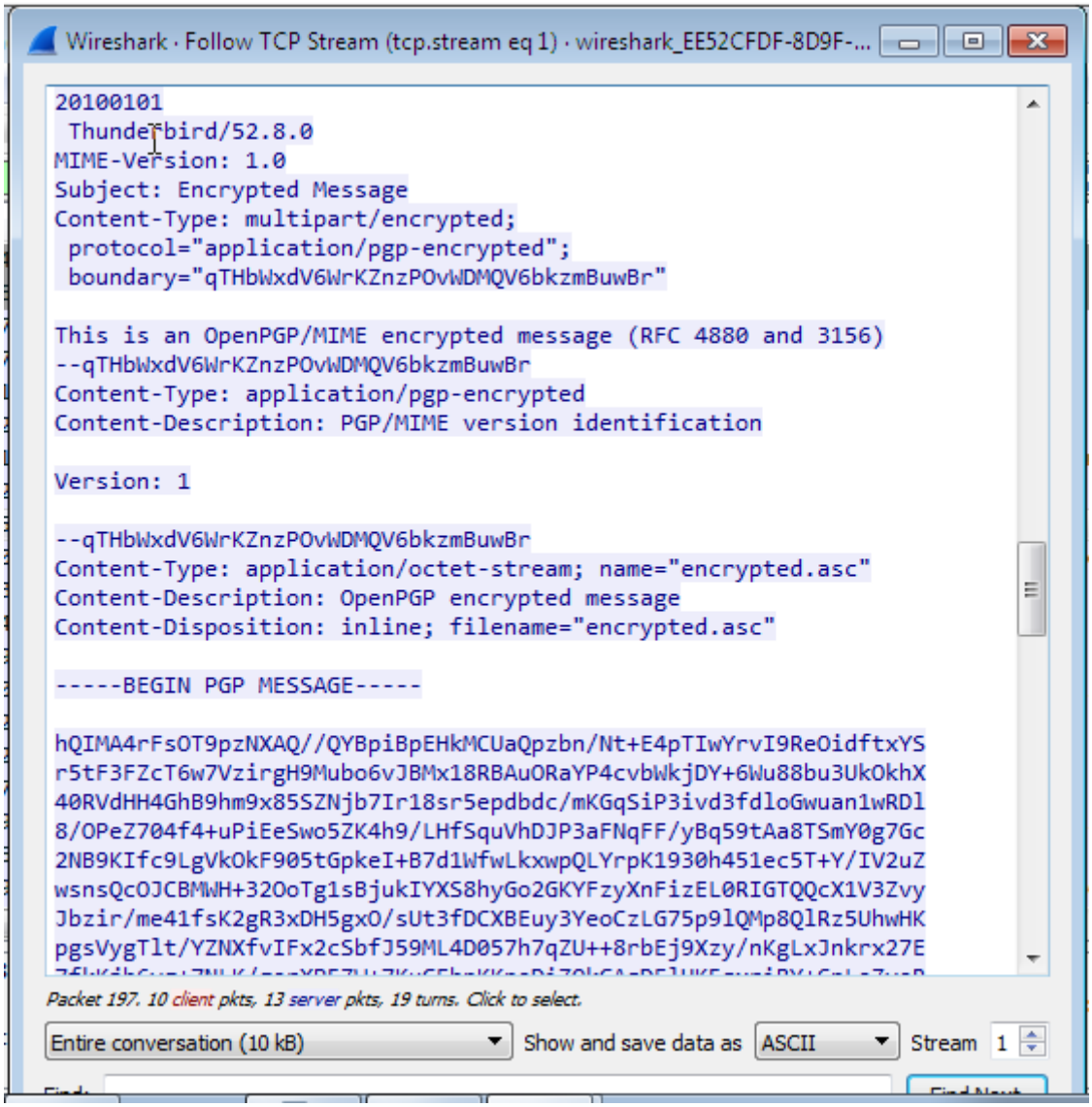
Kada se javni ključ uveze uspešno, dobijamo informacije o njemu. Neke od tih informacija su veličina javnog ključa (4096 bitova), kada je kreiran, kao i otisak tj. Fingerprint tog ključa. Na dugme details, možemo videti šire informacije o ključu i vlasniku ključa. Takođe, naravno, vidimo i vlasnika ključa, čiji smo ključ i importovali.



Slika 22 – PGP Slanje prvog enkriptovanog maila

Korisnik će sada pristupiti slanju prvog enkriptovanog maila. Čim unesemo korisnika čiji smo ključ uvezli, napišemo poruku i idemo na Send, dobićemo dijalog u kome piše da smo uspostavili standard za enkripciju tj. Skrivanje poruke, nečitljivim tekstom, i da li želimo da nastavimo, i pošaljemo enkriptovanu poruku, ili ipak da je i dalje ostavimo nezaštićenom.

Naravno, biramo da pošaljemo zaštićenu poruku!



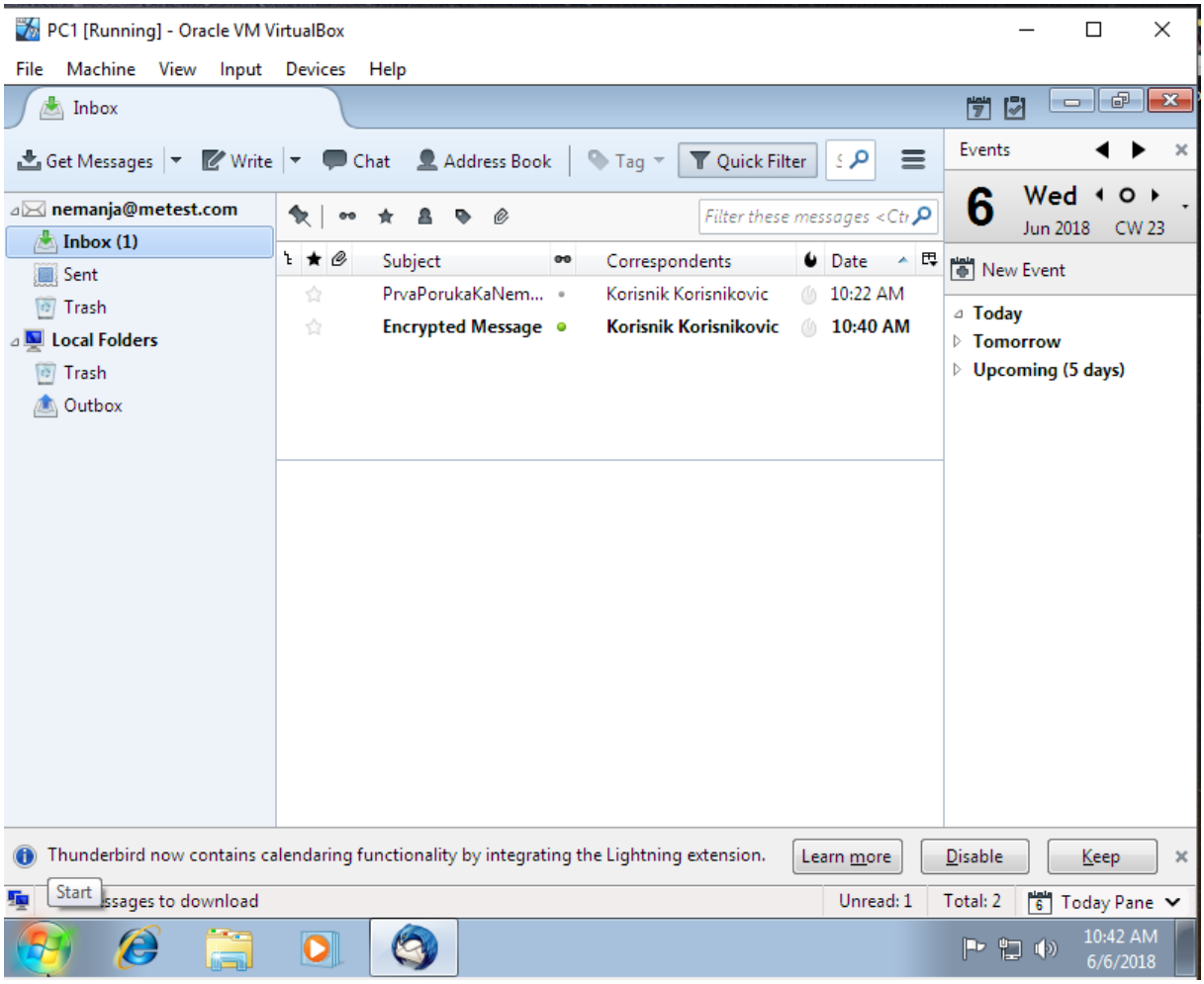
Slika 23 – PGP Analiza prvog enkriptovanog maila

Sada, kada smo poslali prvi mail zaštićen PGP enkripcijom, preko Wiresharka, koji sve vreme radi u pozadini, možemo analizirati saobraćaj filterom POP. Vidimo da korisnik nemanja@metest.com skida poruku sa servera, pa je i wireshark kupi, nakon čega pratimo TCP stream poruke. Kao i do sada, možemo videti korisnika i njegovu lozinku bez problema. Međutim, sadržaj poruke je sada enkriptovan i umesto plain teksta, dobijamo početak poruke u vidu
BEGIN PGP MESSAGE

Pa gomilu nerazumnog koda (enkripcije)

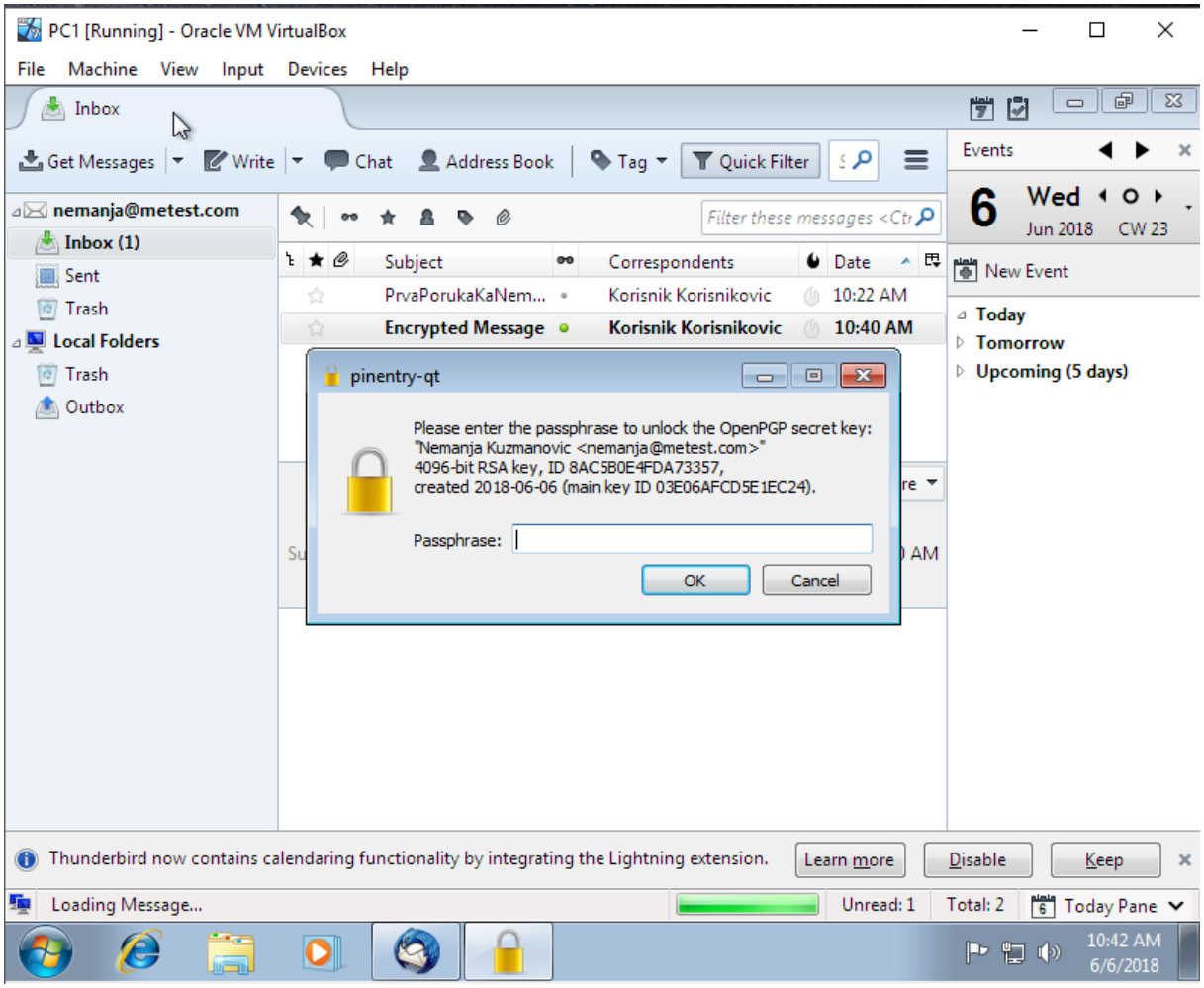
END PGP MESSAGE

Ovu poruku ne može niko pročitati, sem onoga ko ima par tajnog ključa, za javni ključ kojim je ova poruka enkriptovana.



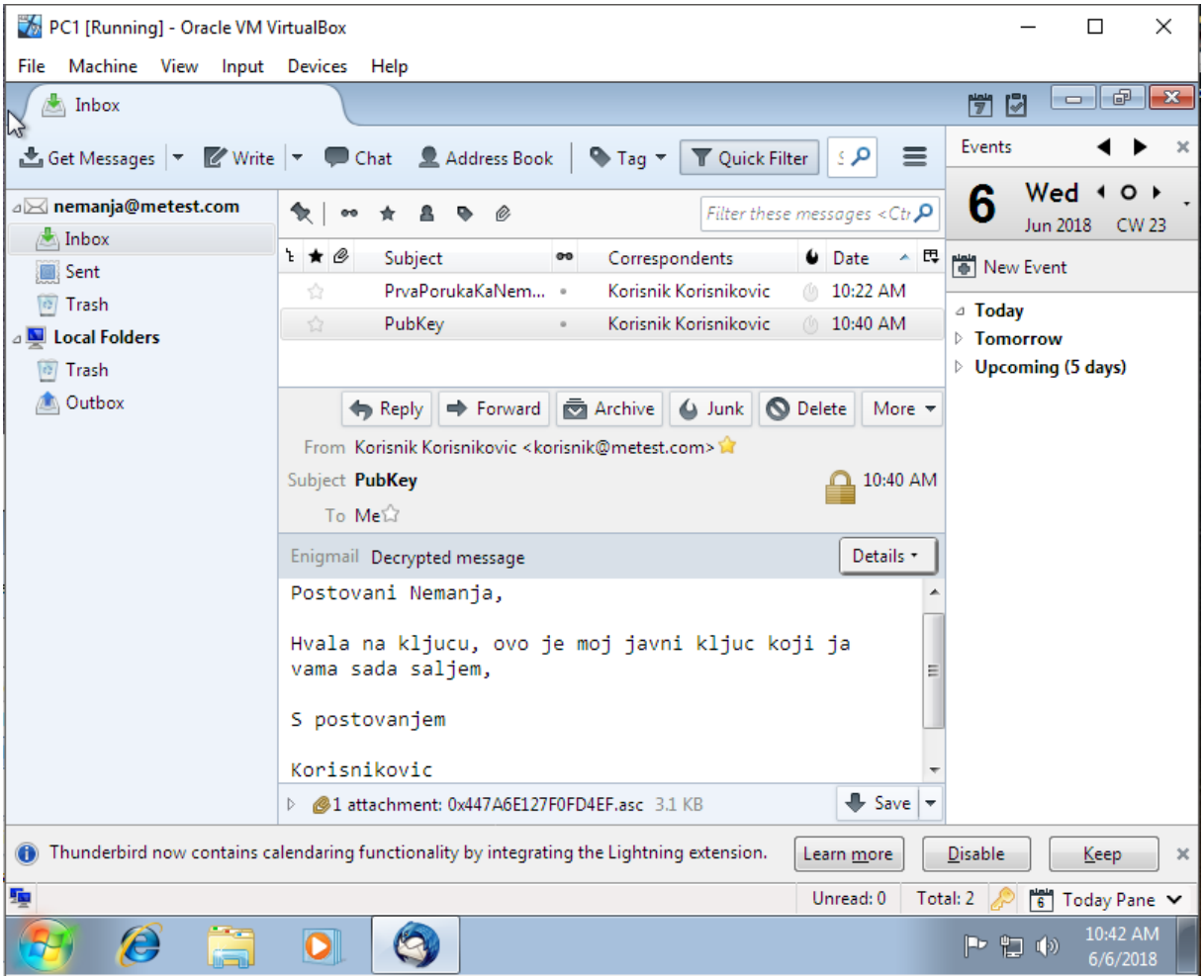
Slika 24 – PGP Pristigli enkritpovan mail

Korisnik koji je primio ovaj mail, i dalje ne može videti sadržaj. Naime, njemu stoji da je poruka enkriptovana, i dok ne unese lozinku za svoj ključ koji je generisao, ne može pročitati sadržaj poruke.



Slika 25 – PGP Unošenje lozinke za tajni ključ

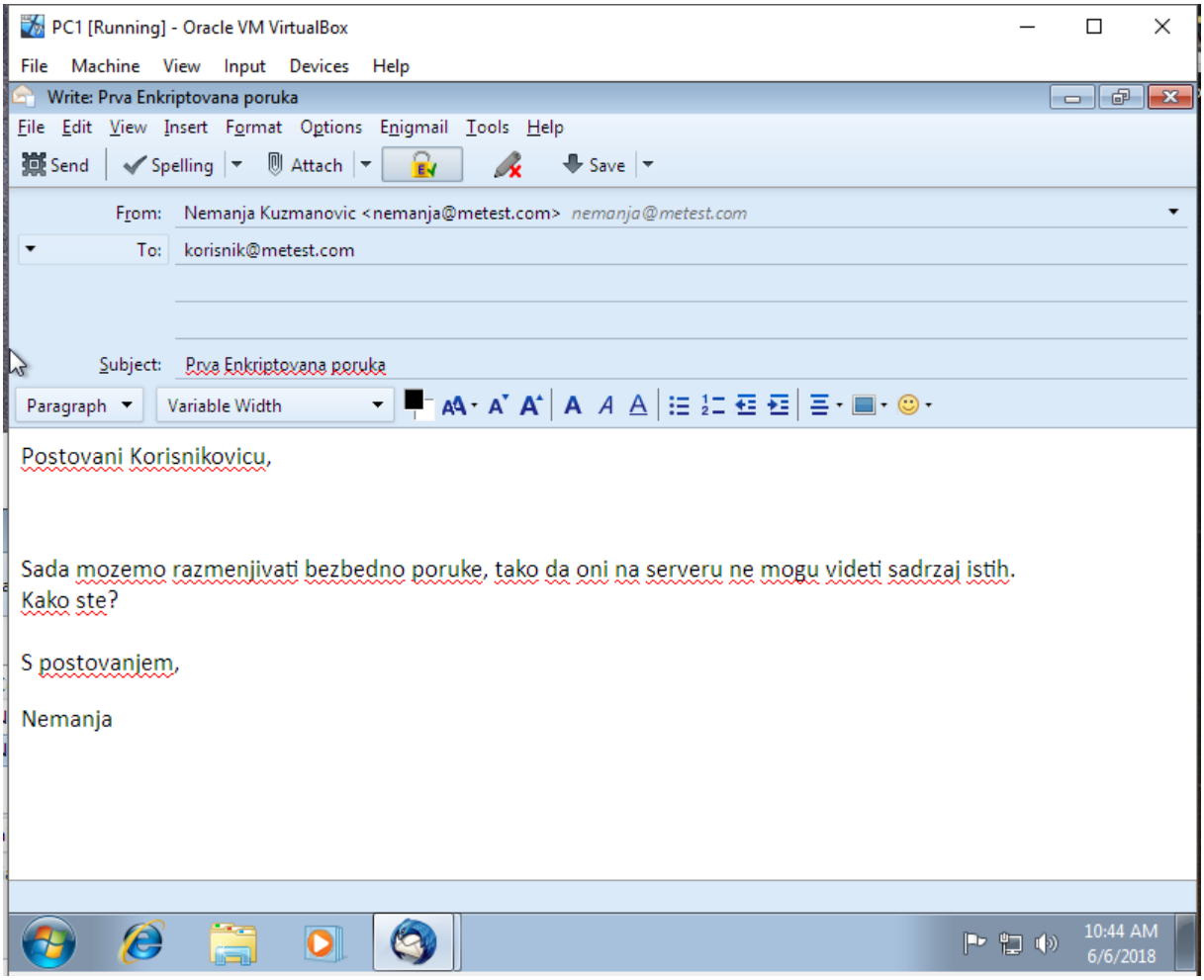
Dakle, korisnik u sledećem dijalogu dobija upozorenje, da unese lozinku za dati privatni ključ kako bi uspeo da pročita sadržaj poruke.



Slika 26 – PGP Čitanje sadržaja enkriptovane poruke

Na početku samog tela poruke, možemo videti deo: Enigmail Decrypted Message, što znači da smo uneli ispravnu lozinku i uspeli jedino mi da dekriptujemo mail i pročitamo sadržaj istog. U ovoj poruku, sada je Korisnik Korisnikovic poslao Nemanji svoj javni ključ kako bi ovaj mogao da ga importuje i uzvrati Korisniku Korisnikovicu enkriptovanu poruku, koju bi ovaj dekriptovao i pročitao. Ovim se obezbeđuje potpuna dvostrana enkripcija i sigurna razmena tela poruke.

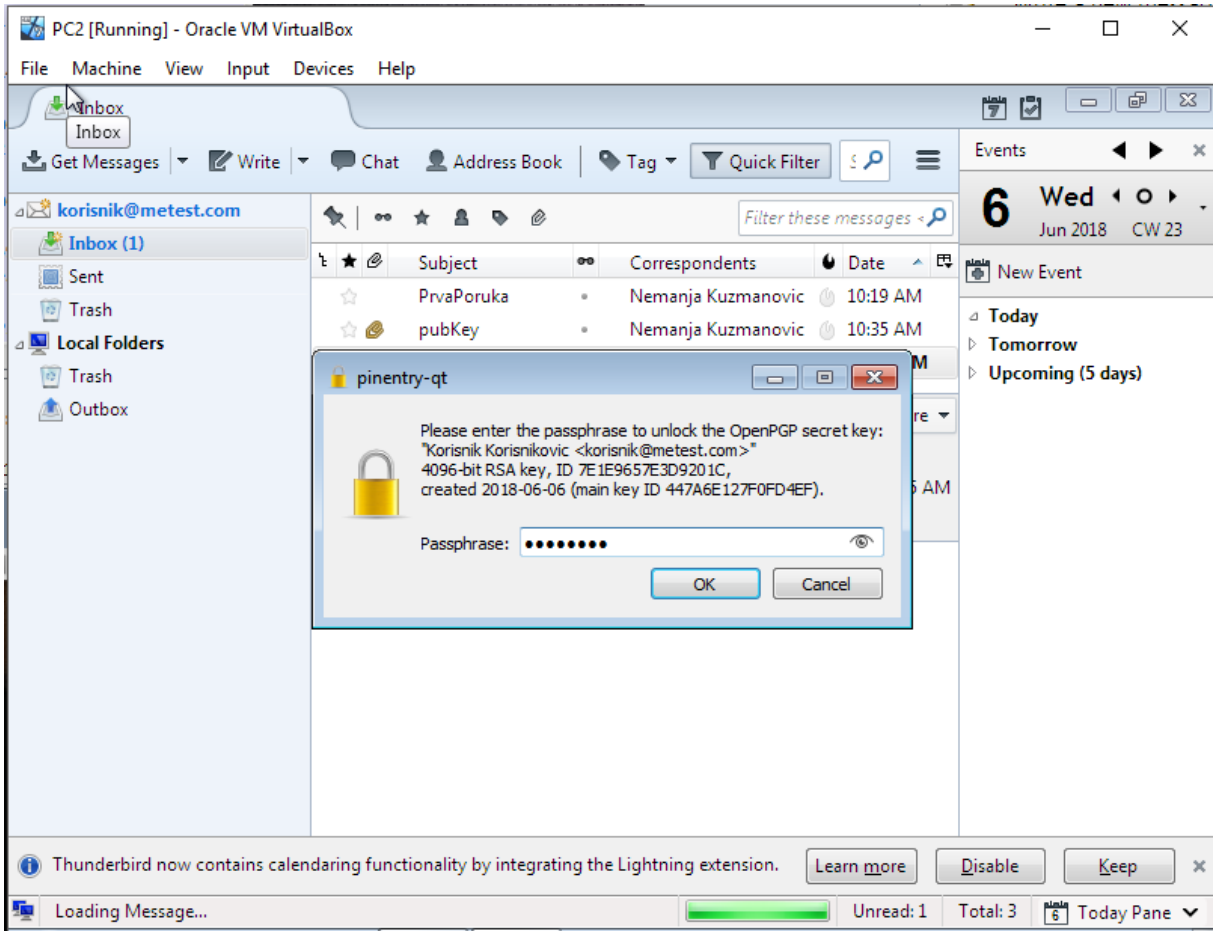
NAPOMENA:
I dalje onaj ko presretne saobraćaj može videti USER i PASS korisnika, potpuno otvoreno, i jednostavno. Naravno, u teoriji, može ući na mail sa tim kredencijalima, ali ukoliko nema lozinku tajnog ključa korisnika, neće moći da pročita sadržaj. Jer se ona unosi svaki put kada korisnik ponovo uđe u klijent i želi da isčita raniju poruku!



Slika 27 – PGP Obostrana enkripcija

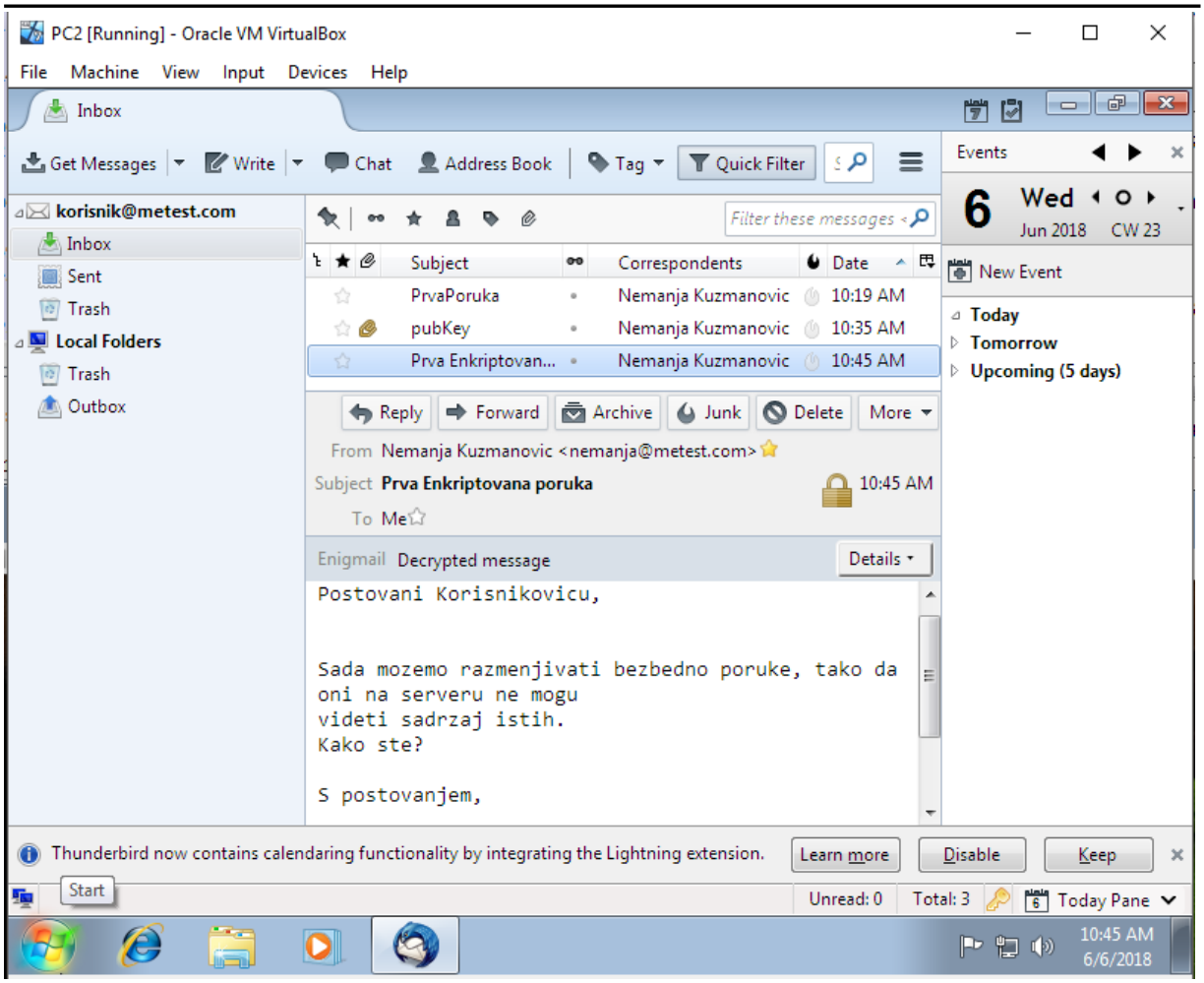
Sada ćemo korisniku odgovoriti sa naloga nemanja, kako bi demonstrirali potpuno zaštićenu i enkriptovanu razmenu poruka.

Takođe u gornjem delu ekrana na sredini vidimo žuti E ključ (Encryption) koji je štikliran istog trenutka kada unesemo email korisnika čiji ključ imamo importovan. Što znači da nam je mail koji šaljemo enkriptovan i samo ga taj korisnik može pročitati unošenjem lozinke za svoj privatni ključ.



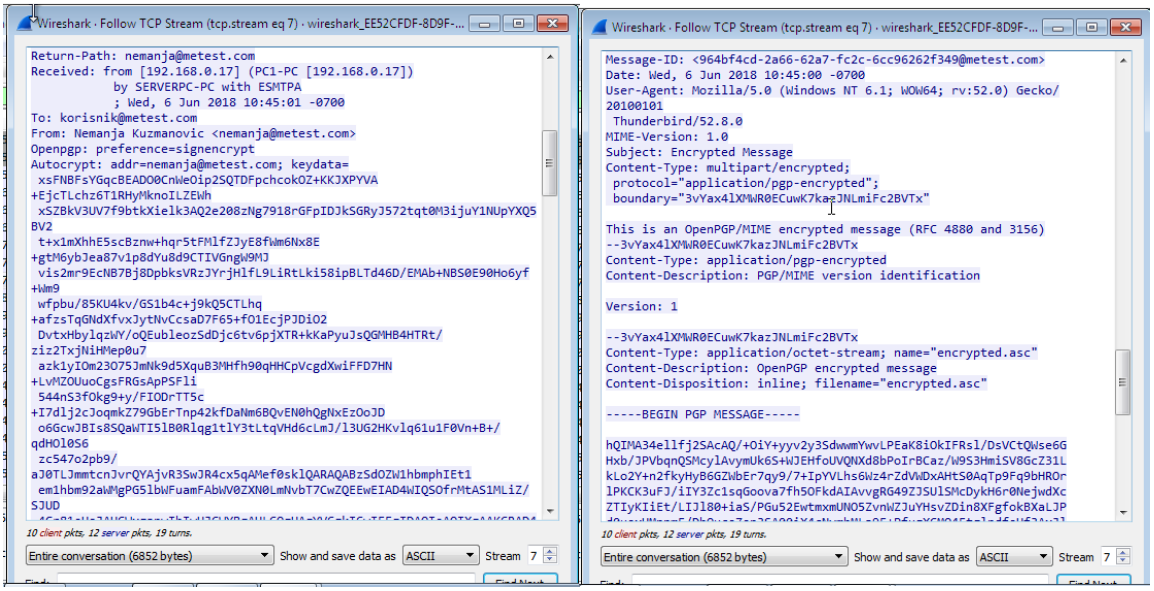
Slika 28 – PGP Čitanje enkriptovane poruke

Kada korisnik primi poruku, kako bi je pročitao, mora uneti svoju lozinku za tajni ključ.



Slika 29 – PGP Čitanje dekriptovane poruke

Poruka dekriptovana, i može se pročitati.



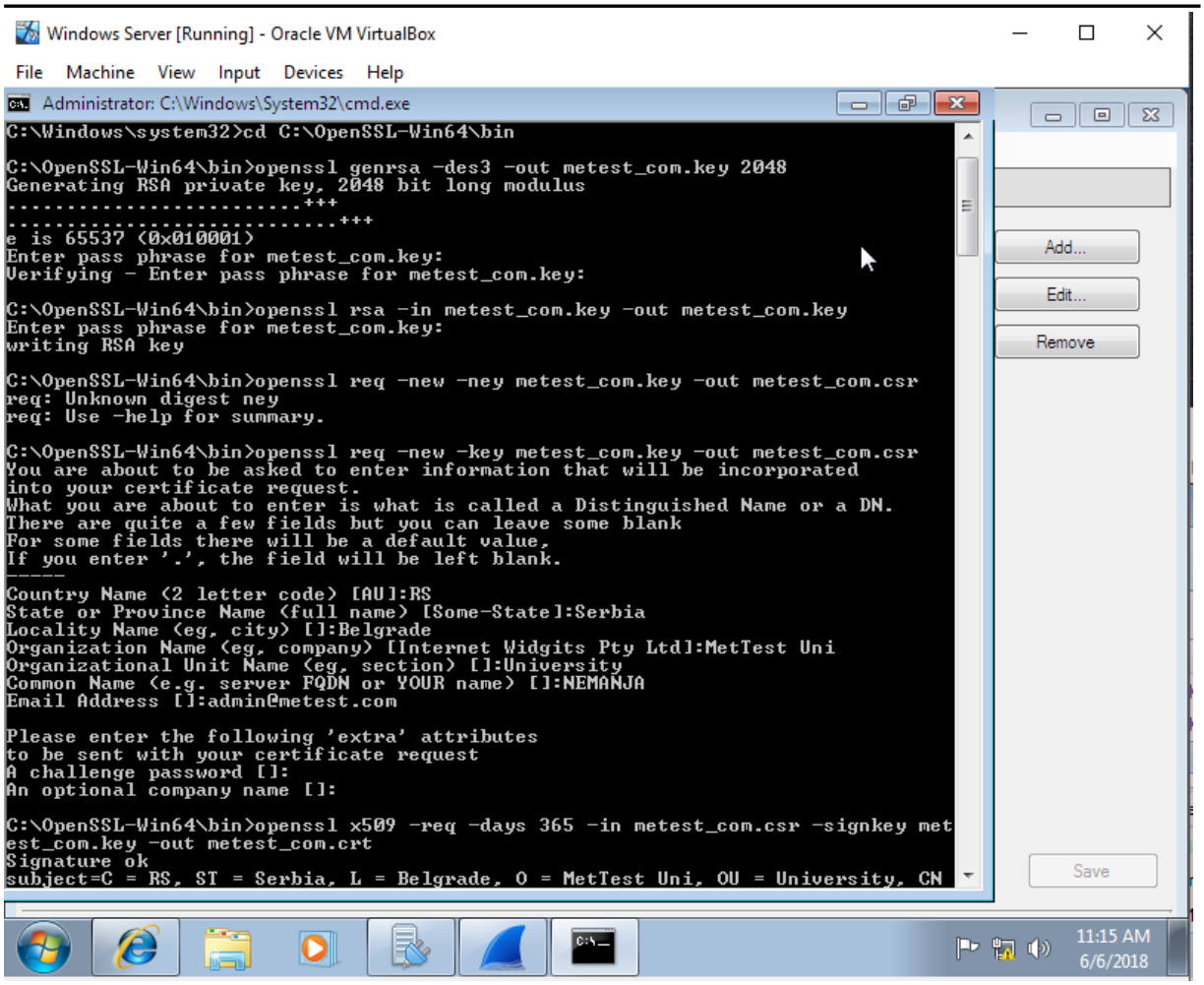
Slika 30 – TCP stream enkritpvane poruke

Kao što je već bilo reči i primera, kada otvorimo TCP stream, videćemo potpuno enkriptovanu razmenu poruka u oba smera, i umesto tela poruke kao otvoren tekst, videćemo znakove enkripcije koji su nečitljivi, i tako se štitimo od ljudi koji bi nam osluškivali pakete maila preko nebezbedne mreže.

4.5 OpenSSL Potpisivanje sertifikata

Kako bi onemogućili da nam neko osluškivanjem saobraćaja pročita USER I PASS koje POP3 otvoreno traži kako bi skinuo mail sa servera i prikazao korisniku, moramo na samom serveru imati SSL (Secure Socket Layer), koji će umesto POP3 i SMTP koristiti TLS v1.2 kako bi u potpunosti enkriptovao samo komunikaciju između klijenta i servera, a PGP već imamo za telo poruke, tako da se kompletan mail potpuno bezbedno na ovaj način šalje putem nebezbedne mreže (interneta).

Za početak skidamo OpenSSL za windows (OpenSSL-Win64.exe).



Slika 31 – OpenSSL potpisivanje sertifikata

Sertifikat ćemo sami potpisati, jer nemamo mogućnosti plaćanja sertifikacionom telu kako bi ga javno odobrio. Na ovo će takođe većina aplikacija, mail klijenata, pretraživača... upozoriti korisnika kada bude dodavao sertifikat, da je potpisan od strane neautorizovanog lica (self signed), i da nije najbezbedniji način komunikacije, jer ga sertifikaciono telo nije javno potpisalo.

Kada se instalira OpenSSL, konzola (CMD) se otvara pomoću Adrministratorskih prava, ide se u folder /bin instaliranog OpenSSLa koji se na ovoj mašini nalazi na putanji C:\OpenSSL-Win64\bin

Komandom

```
openssl genrsa -des3 -out metest_com.key 2048
```

dobijamo privatni ključ. On je veličina 1024, 2048, 4096 itd. Mi smo ovde odabrali 2048. Kada nas pita za lozinku, unosimo lozinku koja je u skladu sa ‘’bezbednom’’ lozinkom.

Da bi uvezli kasnije ključ (importovali) u hMailServer, .key fajl ne sme imati lozinku, pa nju skidamo sledećom komandom

```
openssl rsa -in metest_com.key -out metest_com.key
```

Da bi dobili CSR za ovaj sertifikat, kucamo sledeću komandu

```
openssl req -new -key metest_com.key -out metest_com.csr
```

Onda će nas pitati za sledeće informacije:

```
Country Name (2 letter code) [GB]:<country code example: NL>
State or Province Name (full name) [Berkshire]:<your state or province name>
Locality Name (eg, city) [Newbury]:<your city>
Organization Name (eg, company) [My Company Ltd]:<your organization name>
Organizational Unit Name (eg, section) []:<your department from the origination>
Common Name (eg, your name or your server's host-name) []:<your_domain_com> (this is the
name that will be requested for the authority. Should this need to change you need a new
certificate)
Email Address []:<your mail address>
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []: <DO NOT FILL IN, LEAVE THIS EMPTY!>
An optional company name []: <DO NOT FILL IN, LEAVE THIS EMPTY!>
```

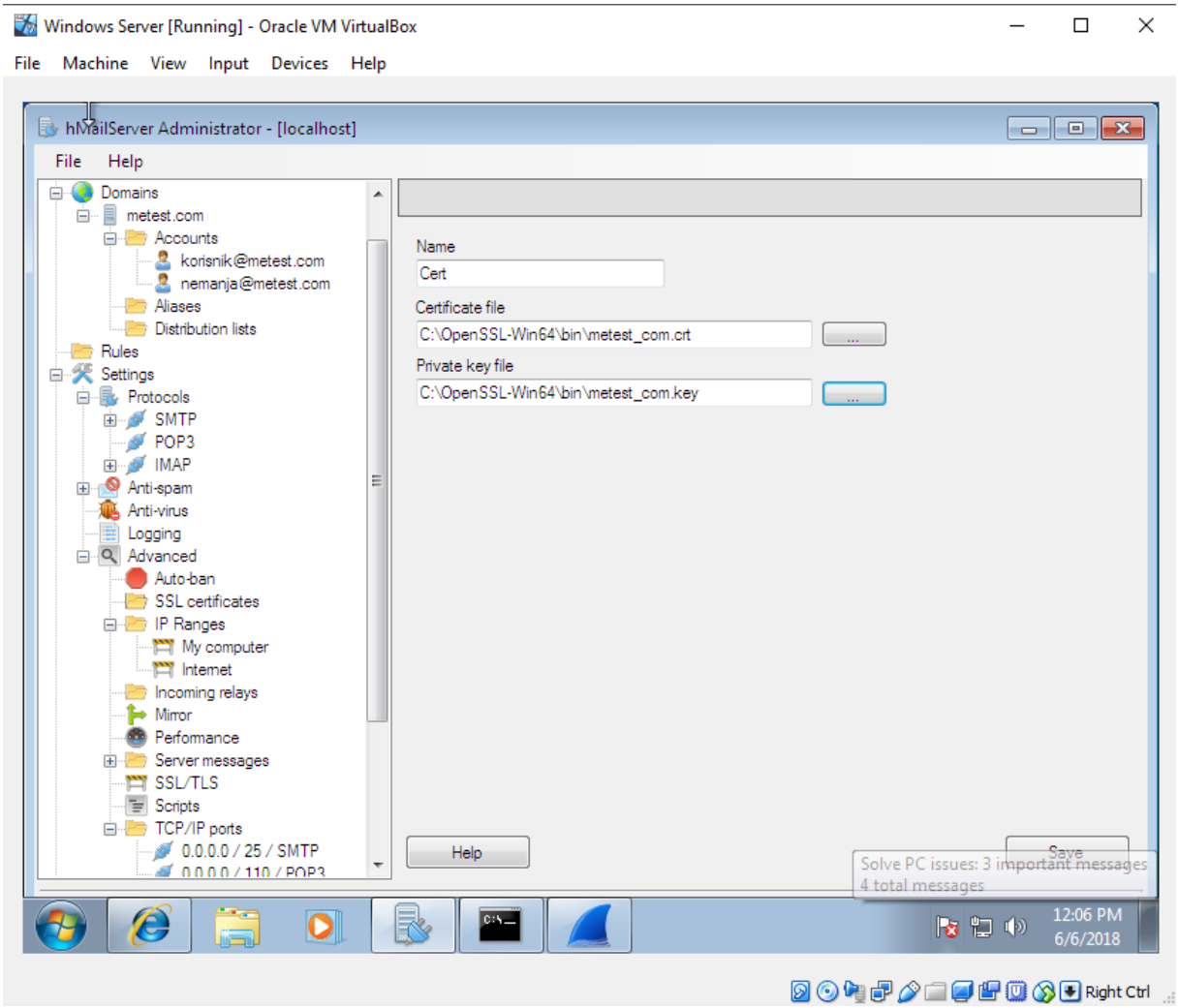
Koje popunjavamo po uptstvu iz dokumentacije hmailserver-a, priloženog iznad.

Sledećom komandom generišemo sertifikat:

```
openssl x509 -req -days 365 -in metest_com.csr -signkey metest_com.key -out
metest_com.crt
```

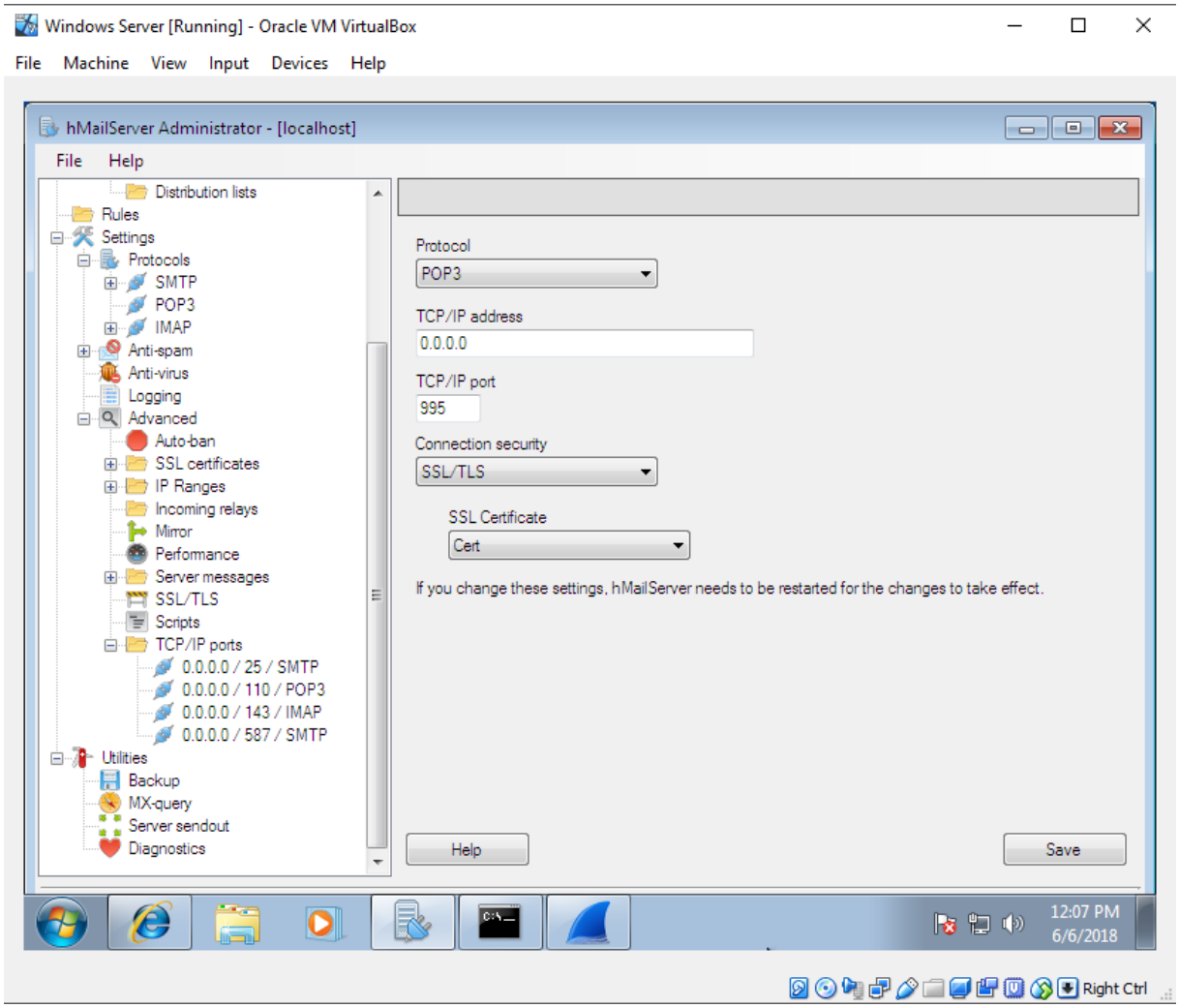
.Key fajl moramo čuvati na bezbednom mestu!!!

Ukoliko bi neko došao u posed ovog ključa mogao bi da kompromituje ceo mail server, i presretne ceo saobraćaj i dekriptuje ga, što predstavlja ozbiljnu bezbednosnu opasnost i propust!



Slika 32 – OpenSSL uvoženje sertifikata

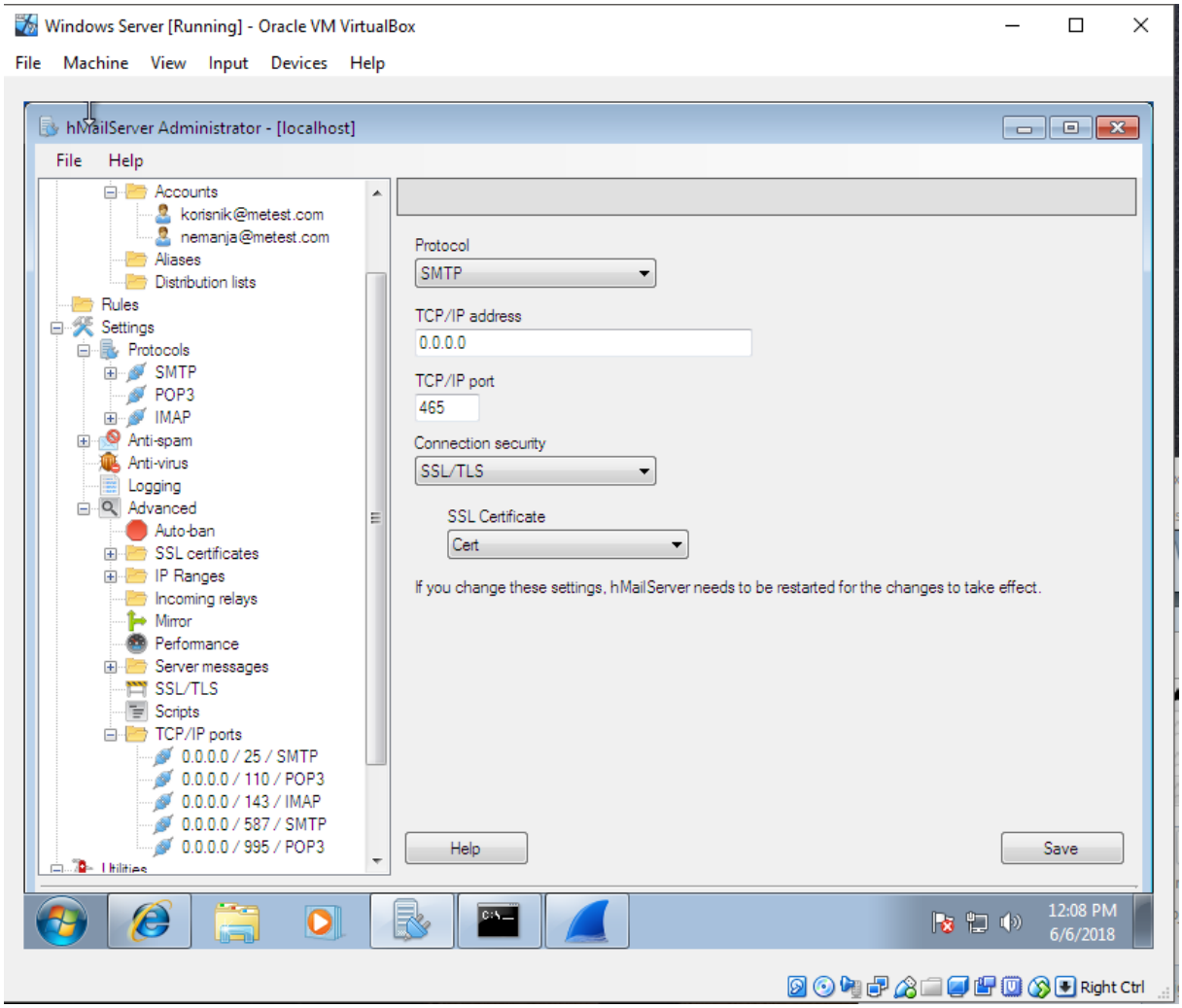
Kako bi uvezli (importovali) sertifikat, idemo na SSL certificates koji se nalazi u Advaned delu aplikacije, unosimo željeno ime sertifikata, a potom uvozimo dva fajla. Prvi je .crt a drugi .key privatni ključ.



Slika 33 – OpenSSL konfigurisanje servera

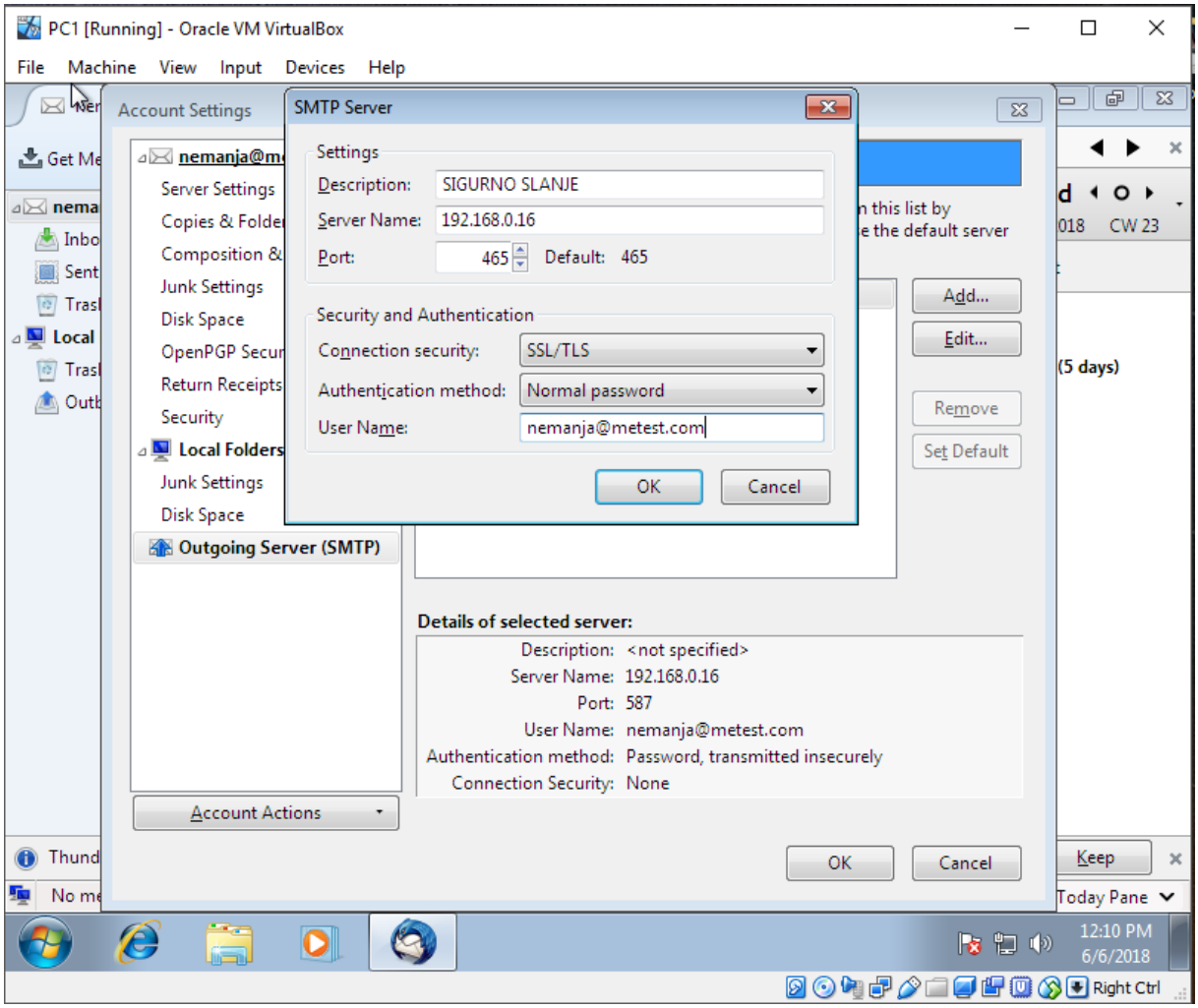
Nije dovoljno samo generisati i potpisati sertifikat. Moramo podesiti i portove sa protokolima, preko kojih radi sam SSL tačnije TLS.

Za preuzimanje poruka koristićemo POP3 protokol, ali na portu 995. Sigurnost konekcije stavljamo na SSL/TLS, i biramo iz padajuće liste naš Cert sertifikat. Čuvanjem promena, mora se restartovati (ponovo pokrenuti) server, kako bi se promene prihvatile, i promenile na podrazumevane.



Slika 34 – OpenSSL konfigurisanje servera - nastavak

Analogno prvom delu, ovoga puta za slanje poruka protokolom SMTP, biramo port 465 (Ovaj port je izabran kao jedan od mogućnosti, nije uzeto u obzir da je on deprecated već, mogli smo odabrati bilo koji port, realno) i ponovo SSL/TLS sa odgovarajućim SSL sertifikatom iz padajuće liste.



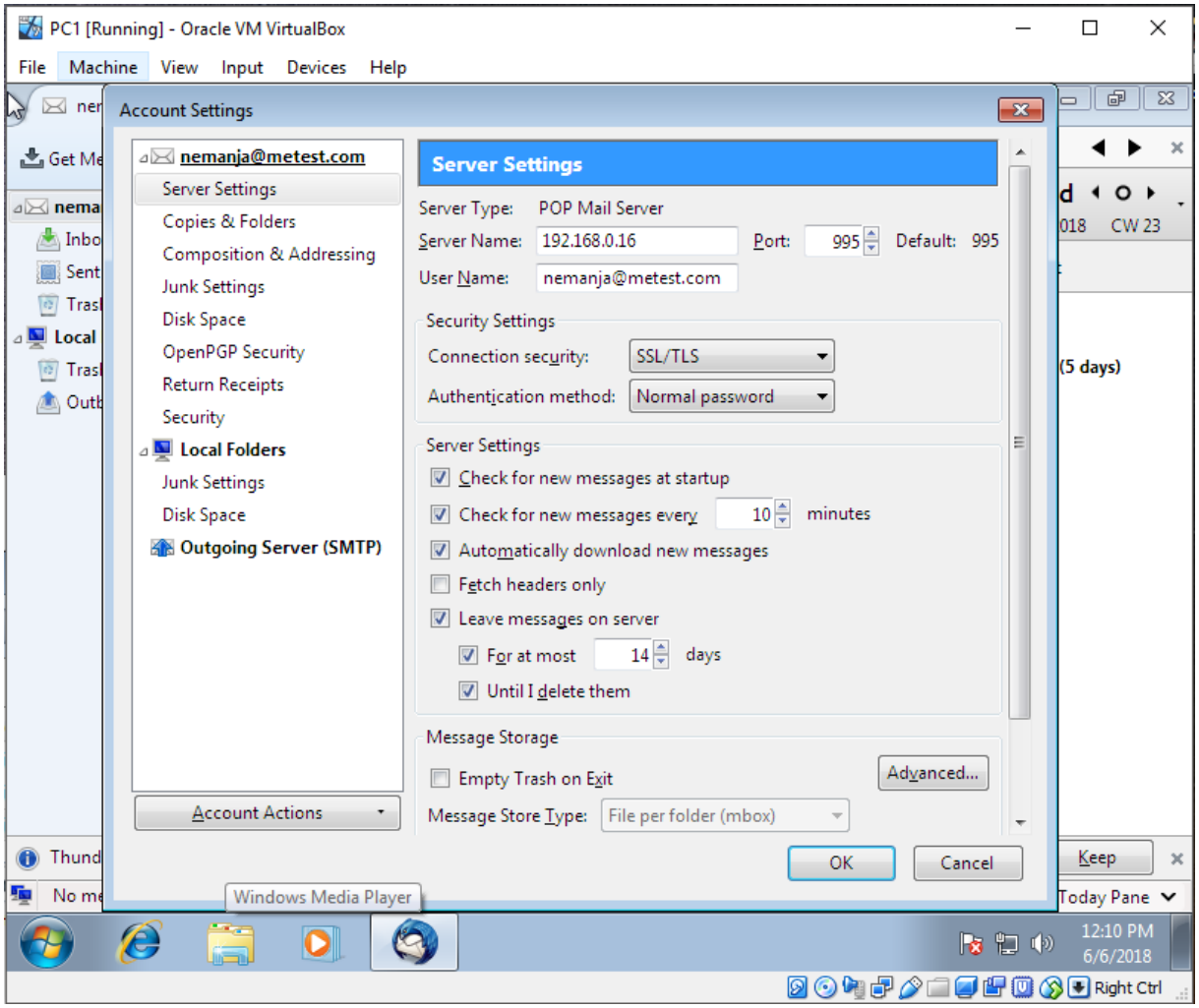
Slika 35 – OpenSSL konfigurisanje klijenta

Kada smo konfigurisali server, moramo konfigurisati i klijent kako bi slao i primao poruke preko odgovarajućih portova, tj. Protokola.

Konkretno, prvo u podešavanjima za slanje (Outgoing server SMTP), pravimo novu opciju, koja ima opis SIGURNO SLANJE, unosime naziv servera (IP ADRESU), biramo port 465, kao i Sigurnosnu konekciju SSL/TLS.

Naravno, moramo uneti i korisnički nalog.

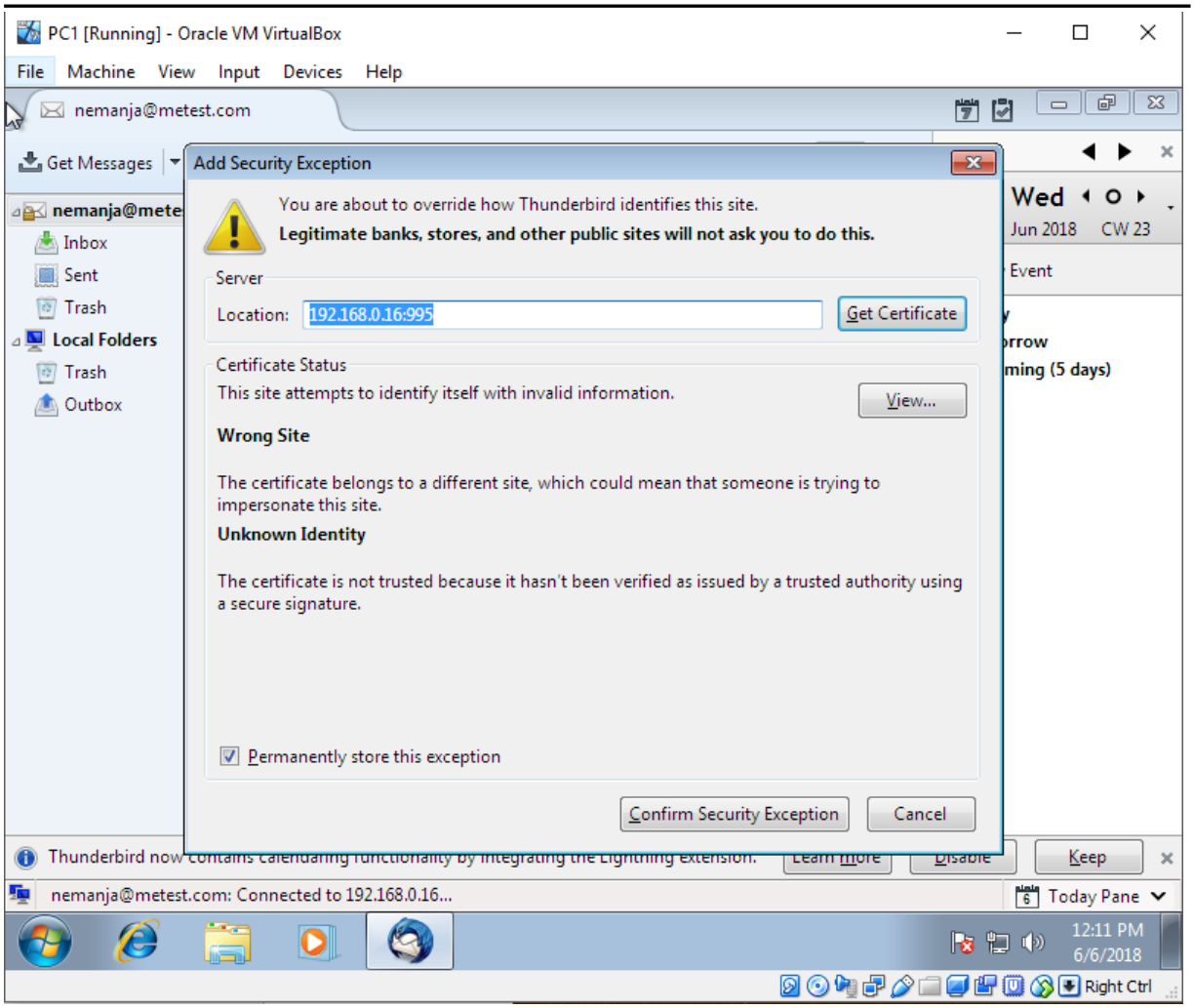
Isto radimo za oba korisnika!



Slika 36 – OpenSSL konfigurisanje klijenta - nastavak

Pod nalogom u Server Settings, podešavamo POP port za primanje maila, a to je 995, kao što je prethodno podešeno na serveru, i ponovo biramo SSL/TLS sigurnu vezu (konekciju).

Dakle, isto radimo za oba korisnika!



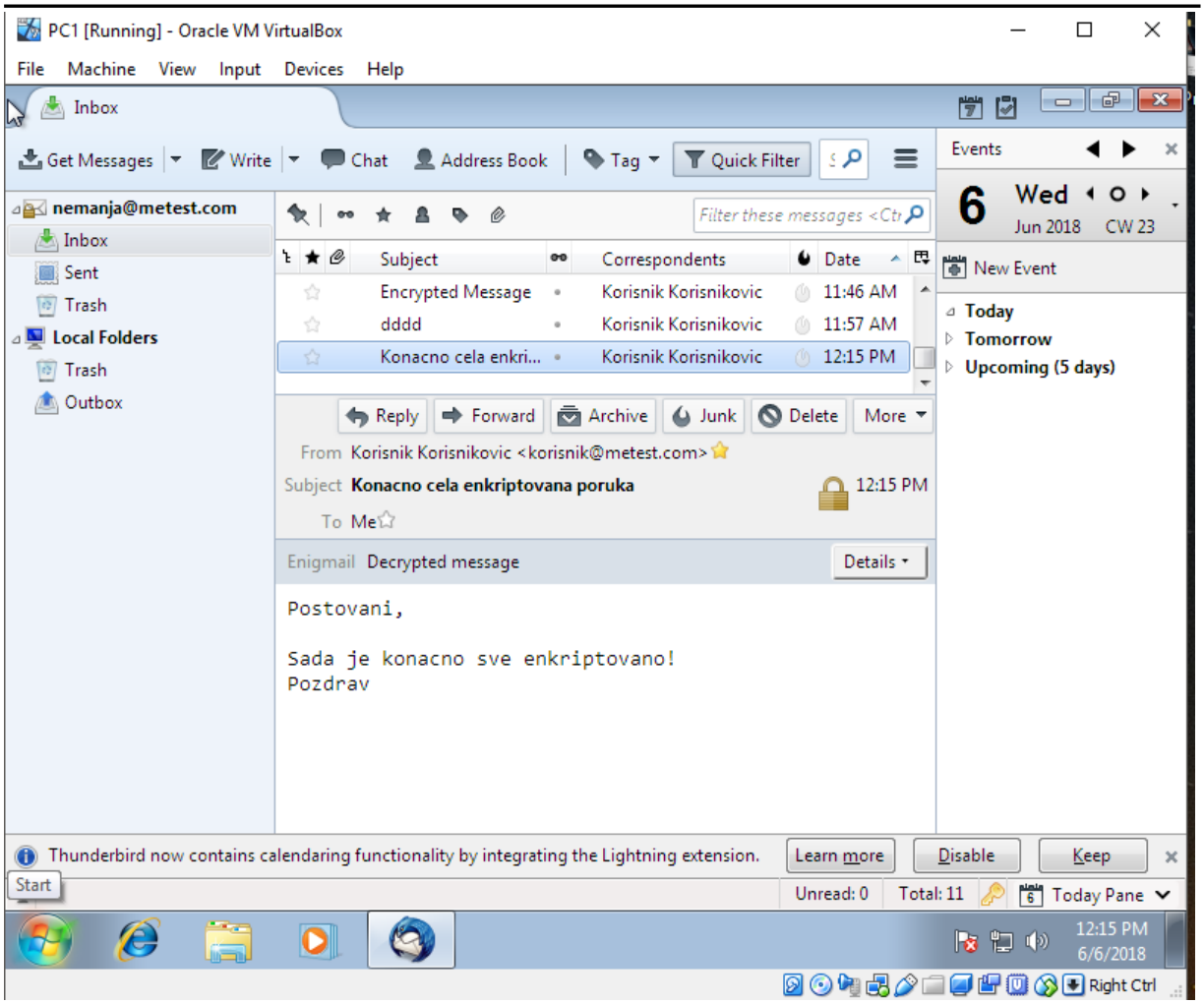
Slika 37 – OpenSSL dodavanje sertifikata

Kada smo to uradili i postavili promene, tj. Konfiguraciju, dobijamo upozorenje da sertifikat nije javno potpisan, da mu se ne zna identitet, i da nije siguran. Takođe, da ga banke, prodavnice i slične institucije nikada neće slati u ovoj formi kao dodavanje bezbednosnog izuzetka.

Pošto smo sami ovaj sertifikat potpisali, ovo moramo prihvatiti kako bi nastavili razmenu poruka sigurnim putem.

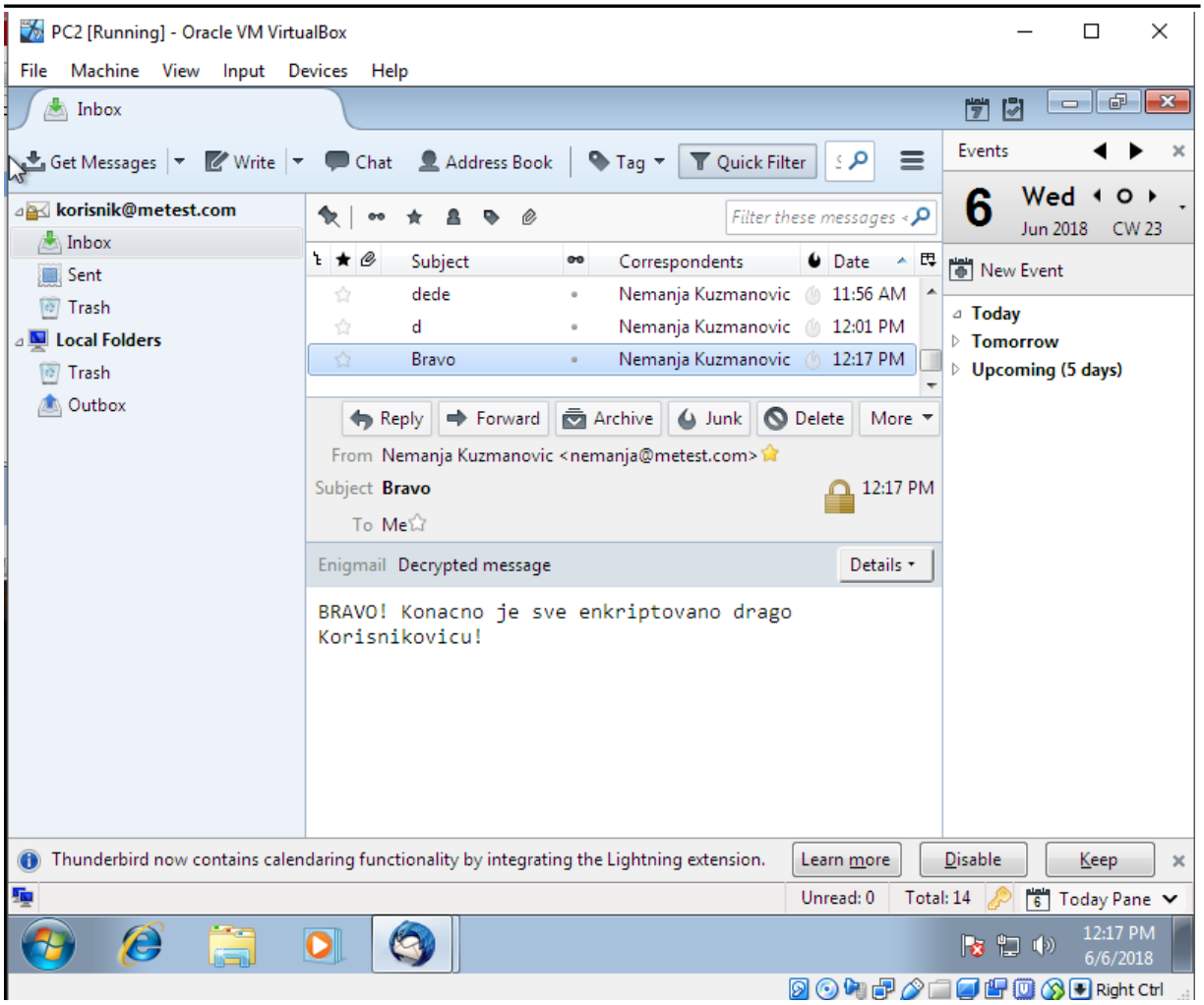
Pritiskom dugmeta Confirm Security Exception, dodajemo izuzetak u klijent, što je poslednji korak pre potpuno enkriptovane i sigurne razmene mailova.

Takođe, ovo isto radimo i kod drugog korisnika!



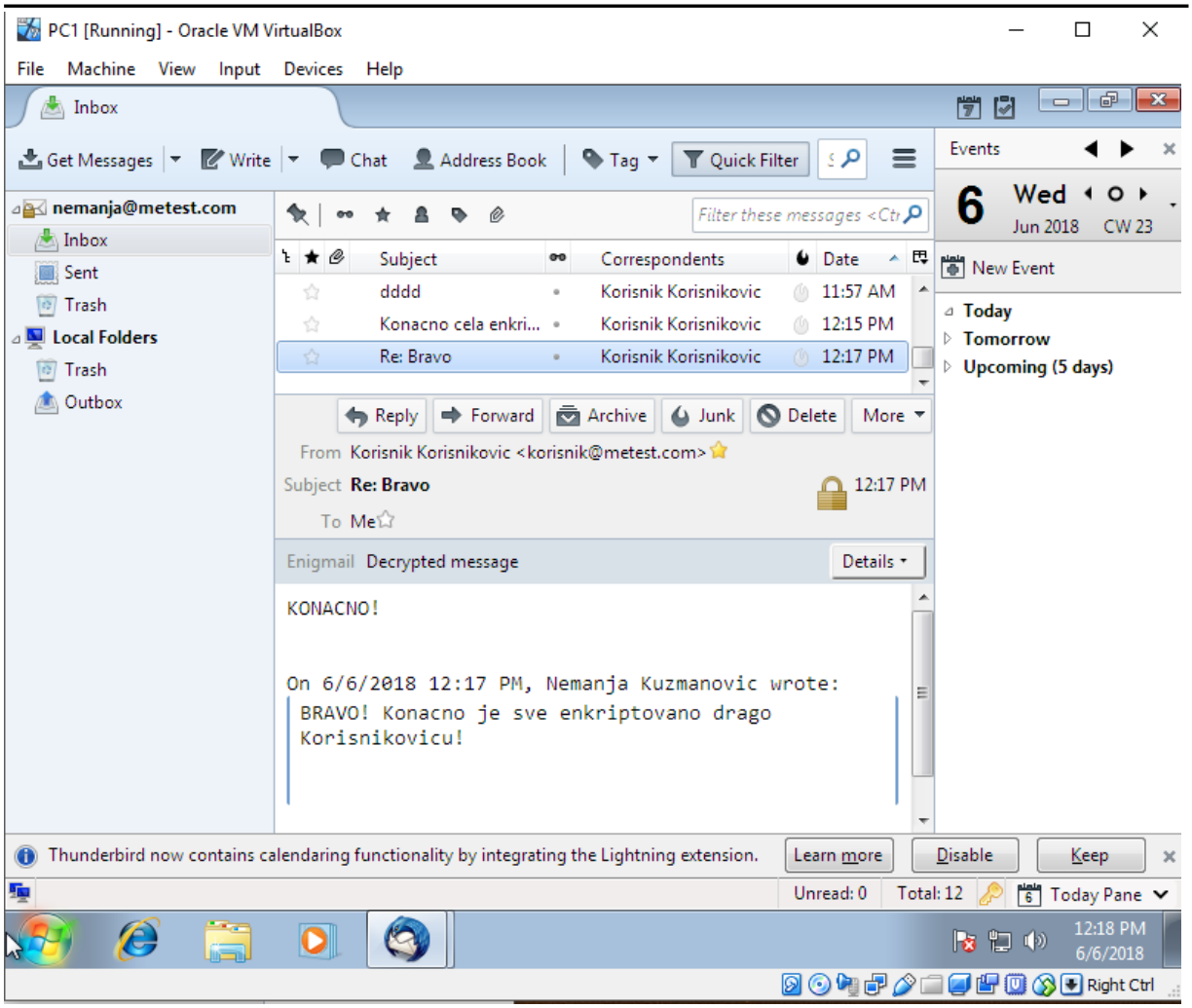
Slika 38 – OpenSSL Slanje prve enkriptovane poruke

Konačno, poslaćemo prvu, potpuno enkriptovanu poruku drugom korisniku!



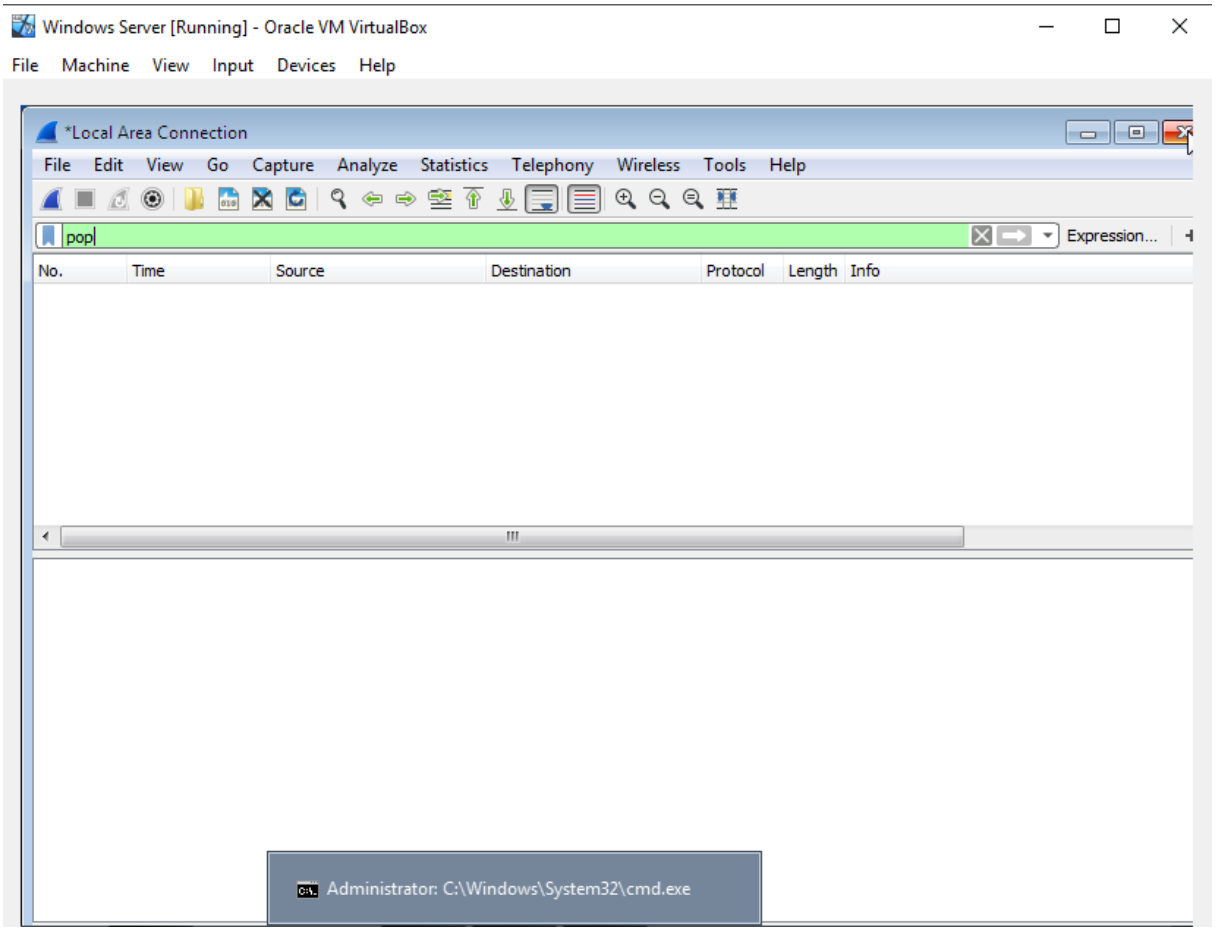
Slika 39 – OpenSSL Primanje prve enkriptovane poruke

Kada šaljemo poruku, korisnik mora pre slanja uneti svoju lozinku kako bi potvrdio slanje. Takođe, kada korisniku stigne poruka, da bi je otvorio, mora uneti svoju šifru za tajni ključ kako bi iščitao sadržaj iste, što je ovde i urađeno.



Slika 40 – OpenSSL Odgovor potpuno enkriptovanim mailom

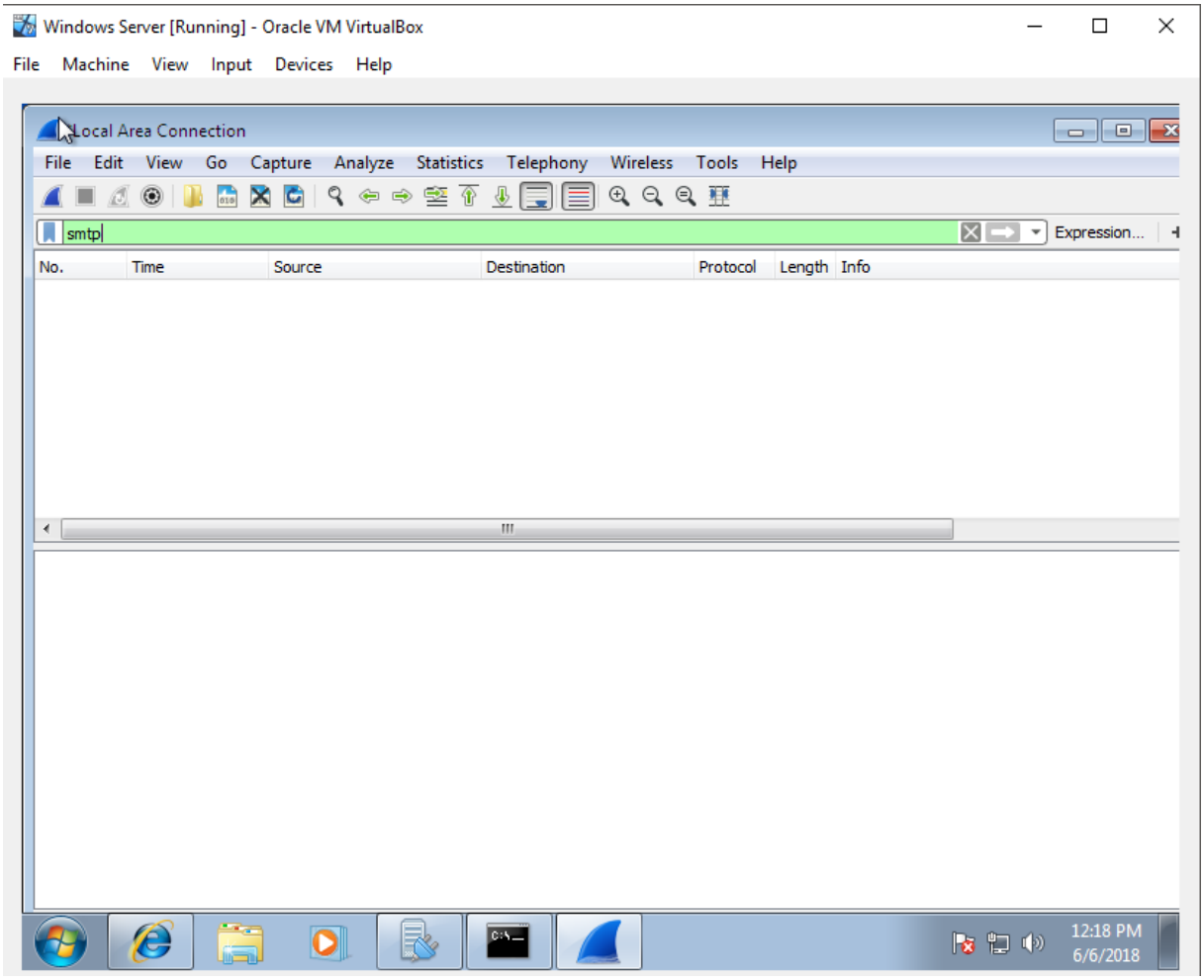
Kao poslednji korak, odgovaramo putem Replaya korisniku, čisto da pokupimo više paketa kroz Wireshark, putem slanja više poruka, kako bi imali bolju analizu saobraćaja.



Slika 41 – OpenSSL Analiza POP saobraćaja

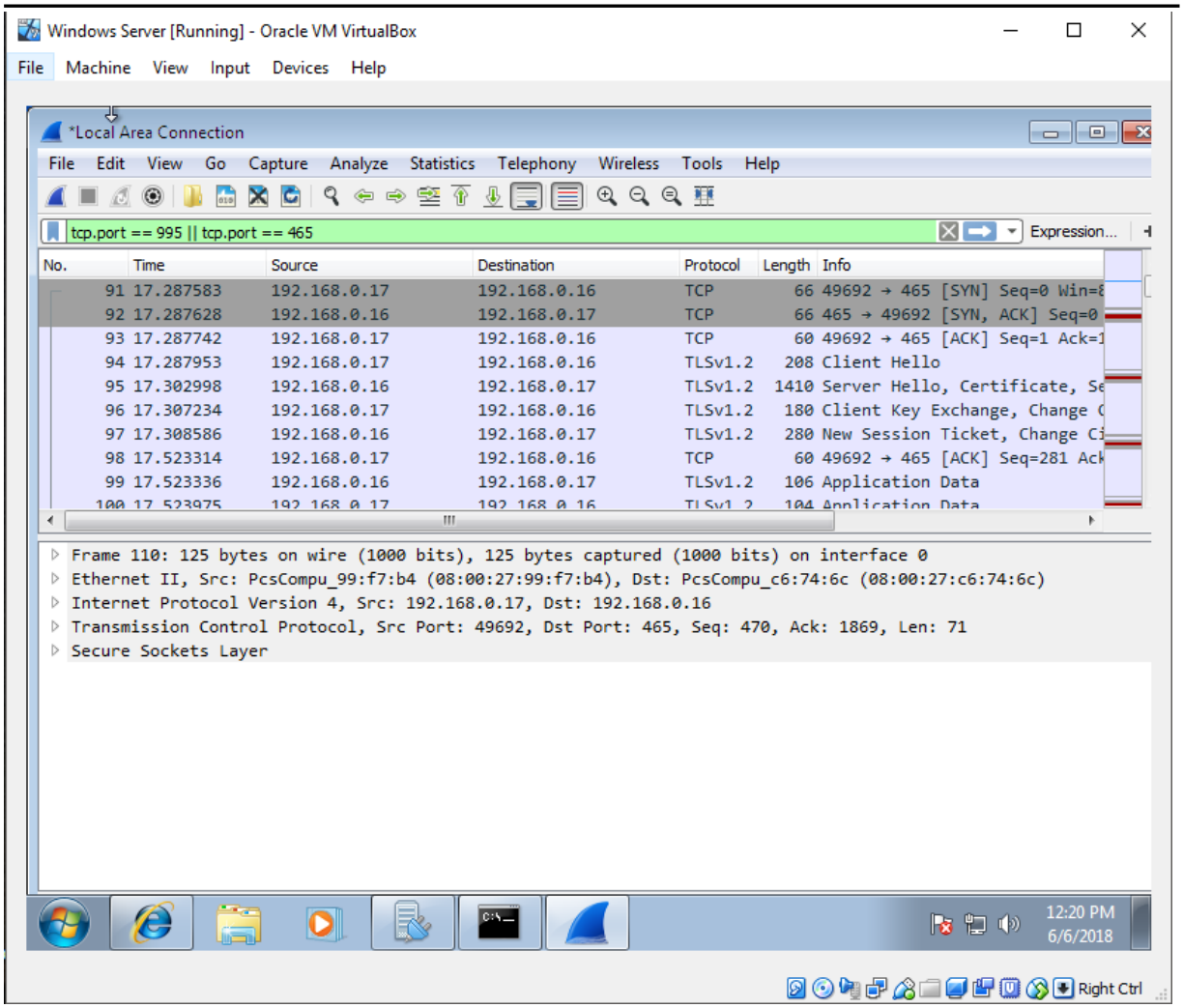
Kao što znamo, do sada, kada ukucamo POP filter, izlazio nam je saobraćaj, i USER I PASS poruka, koja je potpuno otvoreno otkrivala o kom korisniku se radi kao i njegovu lozinku što je bilo potpuno nebezbedno!

Sada, POP se uopšte ne koristi, i filtriranjem paketa ne možemo ništa videti, jer ne koristimo standardan otvoreni POP3 na portu 110.



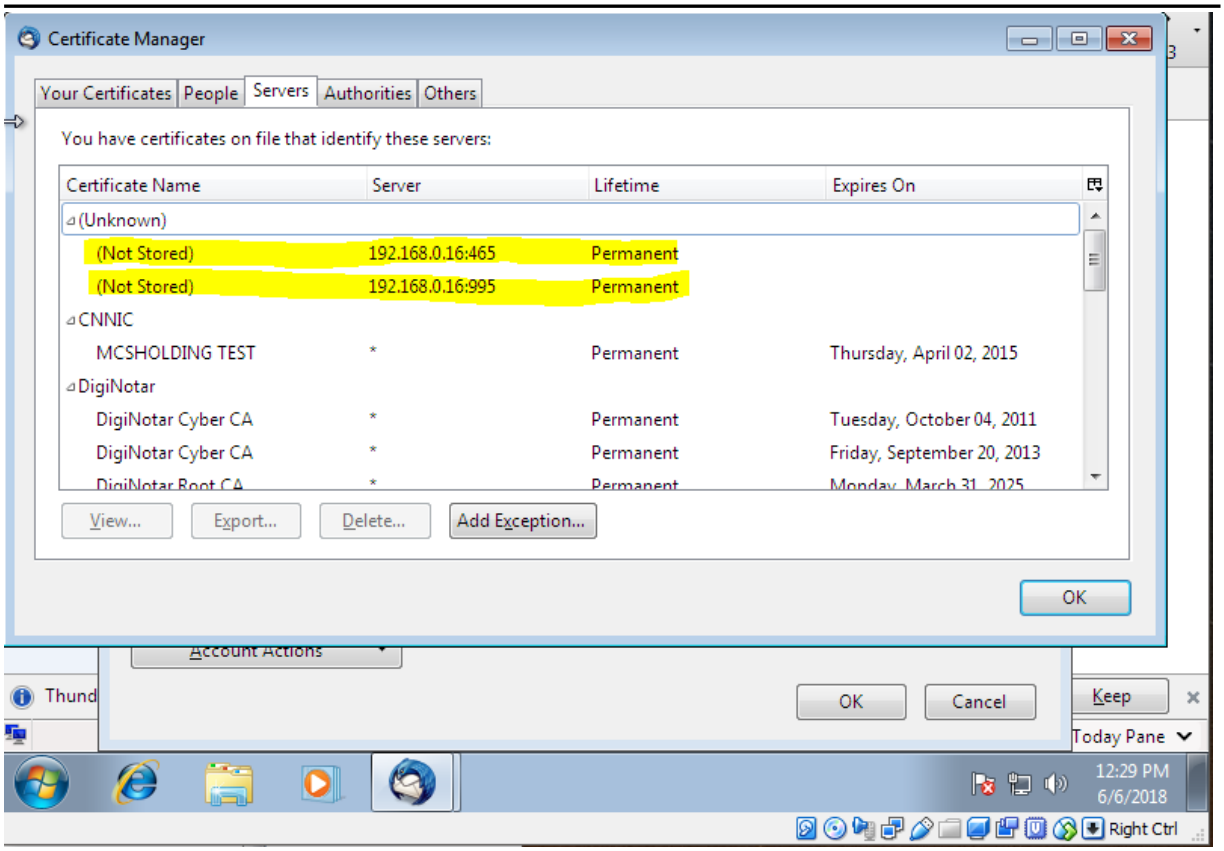
Slika 42 – OpenSSL Analiza SMTP saobraćaja

Isto je i za SMTP saobraćaj. Ne koristimo standardan 25 port za slanje poruka, pa ne možemo videti deo autentifikacije korisnika, kao ni telo poslate poruke.



Slika 43 – OpenSSL Analiza TLSv1.2 saobraćaja

Međutim, kada u filter ukucamo tcp.port == 995 || tcp.port == 465 možemo videti veliki broj saobraćaja, a kada malo bolje pogledamo, vidimo koji IP s kojim razmenjuje i šta. Dakle, vidi se da klijen i server razmenjuju i uspostavljaju sigurnu vezu, putem SSL tačnije TLSv1.2 protokola, i ni jedan bit koji se prenosi nije ne enkriptovan i presecanjem i osluškivanjem saobraćaja ne možemo dobiti apsolutno ništa. Što daje korisnicima potpuno bezbednu obostranu razmenu poruka, kao i razmenu veze klijenta i servera, bez curenja bilo kakvih informacija!



Slika 44 – OpenSSL Sertifikat

Možemo u Certificate Manageru i videti da su instalirani sertifikati za već pominjane portove 465 i 995, sa IP adrese 192.168.0.16 koja odgovara serveru. Dakle, ovde bi stajali potpisani sertifikati od ovlašćenog sertifikacionog tela, kao što već postoje neki, u donjem delu slike.

5. Bezbednost

5.1 Nedostatci

Kao što smo videli u navedenim eksperimentalnim primerima, ukoliko u lokalnoj mreži postavimo mail server i klient bez dodatnog obezbeđenja, doći će do ogromnih posledica. Svi naši korisnici će biti izloženi napadima od ostalih korisnika na mreži, a to je milionski broj. Sam pop3 nema bezbednosne mehanizme jer je protokol za komunikaciju između klienta i samog servera tačnije nam služi za povlačenje poruka sa servera, pa kao što je već navođeno lako se može desiti da neko ko osluškuje pakete negde između servera i klienta (man in the middle) može doći do naših senzitivnih informacija, što nije dobro. Pored ovih nedostataka ne iskustvo korisnika koji konfiguriše mail server može takođe dovesti do ogromnih nedostataka u sistemu, što je nedopustivo.

Poruke poslate iz klienta i preuzete preko pop3 protokola se kao što je već prikazano mogu lako rekonstruisati, ukoliko nije implementirana zaštita i oslanja se na zaštitu samog protokola, tačnije na to da nema zaštite.

5.2 Zaštita I bezbednost

Dakle, nije dovoljno samo konfigurisati klient i mail server i osloniti se na bezbednost protokola, jer sam protokol ne sadrži nikakve bezbednosne mehanizme.

Prva i osnovna i najveća zaštita bi uvek trebalo da bude sertifikat (SSL/TLS). On će nam omogućiti da se bezbedno povežemo sa serverom kroz nebezbednu mrežu bez mogućnosti da nam bilo ko oslušne vezu od klijenta do servera, I tako preuzme naše senzitivne informacije tj. Kredentijale. Tako na primer najpouzdaniji i najupotrebljivaniji SSL sertifikat štiti i nas i korisnika, enkriptovajući svu vezu između nas i servera. Postoji i ona najjeftinija a najnebezbednija varijanta, a to je, ručno pisani sertifikat. Za razliku od zvaničnih sertifikata, javno izdatih od sertifikacionog tela, koji su možemo reći sa sigurnošću sigurni, za ručno pisane se ne može znati postoje li propusti, pa takođe svaki put kada se koristi ručno pisani sertifikat, pretraživač ili klient će izbaciti upozoravajuću poruku da se takav sertifikat koristi i da se pažljivo šalju senzitivne informacije, baš zbog toga što zvanično nije utvrđena validnost istog. Sertifikati se prvenstveno podešavaju na samom mail serveru, a potom i u klientu koji koristimo za pristup serveru, tako što mu naglasimo koji bezbednosni port se koristi kao i koja vrsta sertifikata/enkripcije.

Takođe, za enkripciju same poruke, može se koristiti PGP (Pretty Good Privacy) koji na bazi privatni-javni ključ, enkriptuje telo poruke, I samo oni koji imaju razmenjene ključeve mogu komunicirati na bezbedan način, tj. Dekriptovati I otključati poruku.

Svakako, najbezbednije je kombinacija ova dva, dakle, sertifikat izdat od strane sertifikacionog tela, plus neki bezbednosni mehanizam za enkripciju samog maila tačnije tela poruka.

Naravno, tu postoji ogroman spektar rešenja, što opensource što Enterprice, I u zavisnosti od klijentskih potreba, a I potreba onoga ko naručuje takve usluge (server – klijent), treba razmatrati različita rešenja.

Adekvatnih zaštita postoji, samo treba biti svestan prednosti, mana, propusta, I mogućnosti korisnika da nasedne na zlonamerne napade, pa u skladu sa tim se adekvatno I zaštititi.

5. Zaključak

U projektu se osvrnulo delom na funkcionisanje I funkcionalnosti pop3 protokola, kao I analizu saobraćaja koju traži I propušta pop3. U laboratorijskom (improvizovanim) uslovima je odrađena analiza paketa kao I simulacija razmena poruka kroz lokalnu (u realnom svetu internet) nebezbednu mrežu.

Analizirao se otvoren saobraćaj između dva klijenta preko servera, putem SMTP I POP3 protokola.

To je prošireno, pa se uvela bezbednost korišćenjem PGP-a, izvršila su se testiranja razmena poruka, kao I konfigurisanje I podešavanje servera I klijenata, te se izvršila analiza tako zaštićenih poruka.

Napisan je bezbednosni sertifikat I samostalno potpisan, integrisan u sam server, te je server konfigurisan tako da radi na sigurnom nivou SSL/TLS, te su I klijenti podešeni da koriste istu opciju.

Izvršena je ponovo analiza I zabeležene ralike, prednosti, mane, nedostatci…

Na kraju samog projekta, pored osvrta da protokole, naučeno je bezbedno konfigurisanje klijenta pomoću PGP-a, konfigurisanje servera putem samostalno potpisanih sertifikata, te se osvrnulo na obnavljanje znanje podizanja mašina, razmene podataka između njih, kao I uvid I bolje razumevanje propusta u samoj bezbednosti, bez korišćenja adekvatnih preporučenih zaštita.

Takođe se osvrnulo na korišćenje Wiresharka, gde sve kada se uzme u obzir šta je korišćeno, može se reći da je pokrilo dobar deo stvari koje se rade na samim vezbama iz ovog predmeta.

Svakako da će ovo iskustvo I znanje dobro doći u nastavku karijere, pogotovo što će svima pri odbrani projekta podići svest koliko razmena senzitivnih podataka može biti nebezbedna, I ugroziti samog korisnika, a što je još gore, sutra možda I neku veliku kompaniju.

6. Reference

1. http://lams.metropolitan.ac.rs:8080/lams/ Predavanja Vežbe
2. https://wiki.mozilla.org/Thunderbird/Docs
3. https://www.openpgp.org/
4. https://www.hmailserver.com/
5. Stackoverflow.com