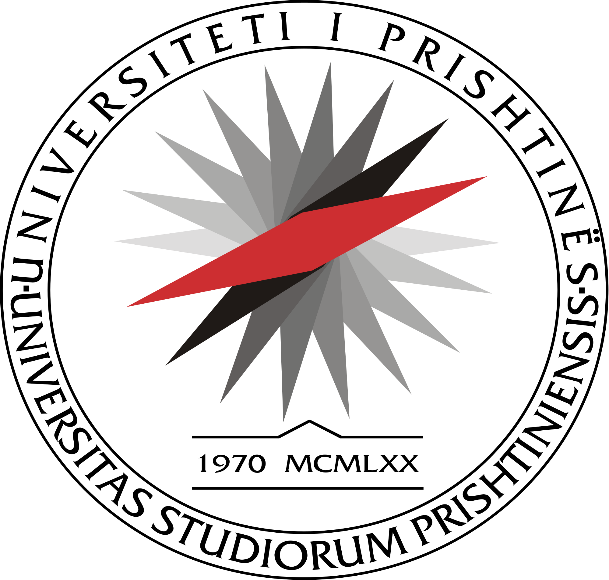
**UNIVERSITETI I PRISHTIN****ËS­**

Fakulteti i Inxhinierisë Elektrike dhe Kompjuterike

Departamenti i Inxhinierisë Kompjuterike

­

**SIGURIA E TË DHËNAVE**

Raporti i projektit të dytë

Studentë: **Natyra Vitija**  Profesori i lëndës: **Blerim Rexha**

**Toni Krasniqi** Asistenti i lëndës: **Arbnor Halili**

**Zahir Maliqi**

Prishtinë, Maj 2020

**Përmbajtja**

1. Hyrje..................................................................................................................................3

2. Komanda create-user……………..........................................................................................4

3. Komanda delete-user….....................................................................................................4

4.Komanda export-key..........................................................................................................5

5.Komanda import-key.........................................................................................................6

6.Komanda write-message...................................................................................................7

7.Komanda read-message....................................................................................................8

8. Udhëzimet rreth ekzekutimit te komandave………………………………………………………………9

# **Abstrakti**

Ky raport ka të bëjë me fazen e dytë të projektit në lënden Siguria e të Dhënave ku si qëllim e kemi vazhdimin dhe zgjerimin e fazës së parë. Komandat të cilat bazohen në teknika modern të enkriptimit simetrik dhe asimetrik.

Këto komanda janë: Komanda create-user

Komanda delete-user

Komanda export-key

Komanda import-key

Komanda read-message

Komanda write-message

# **Hyrja**

Para se të tregojmë komandat dhe funksioni i tyre, së pari duhet të kemi njohuri mbi disa koncepte të tjera në mënyrë që ta kemi të qartë se për çfarë bëhet fjalë.

Enkriptimi simetrik, i njohur ndryshe edhe si kriptimi me “çelës secret”,përdorë një madhësi sekrete me të cilën kriptohen të gjitha mesazhet.Kriptimi simetrik i ka këto karakteristika:   
Enkriptimi simetrik është matematikisht më i thjeshtë prandaj është edhe më i shpejtë sesa kriptimi asimetrik,prandaj edhe për këtë arsye është më i përdorshëm në rastet kur dëshirojmë të kriptojmë fajllat që janë shumë të mëdha.

Enkriptimi asimetrik është një formë e enkriptimit ku enkriptimi dhe dekriptimi bëhen duke përdorur qelësa të ndryshëm,një qelës publik dhe një qelës privat. Enkritpimi asimetrik më I perdorur është RSA që përkrahet edhe nga .NET. Skema e RSA është një bllok-shifer ku teksti i pa kriptuar dhe teksti I kriptuar jane numra të plotë në mes 0 dhe n-1 për n-numra. Zakonisht madhësia e n është 1024 bita.

# **Komanda create-user**

Kjo komande ka per detyre qe te krijon një çift publik/privat të RSA .

Ekzekutimi I komandes:

create-user <name>

Per shembull:

create-user zahir



create-user toni



Nuk krijohen celesa te cilet ekzistojne paraprakisht:



Emri <name > duhet te permbaje shkronja,numra dhe ”\_” por jo hapesira “ ”:



# **Komanda delete-user**

I largon të gjithë çelësat ekzistues të shfrytëzuesit.

Ekzekutimi I komandes:

delete-user <name>

Per shembull:

delete-user zahir



Ne rastin kur egziston vetem celesi publik ose ai private i shfrytëzuesit dhe deshirojme ta largojme:

Nese dojme me largu (me fshie ) celesat e nje shfrytëzuesit qe fare nuk egziston:

# **Komanda export-key**

Eksporton çelësin publik ose privat të shfrytëzuesit nga direktoriumi i çelësave.

Ekzekutimi I komandes:

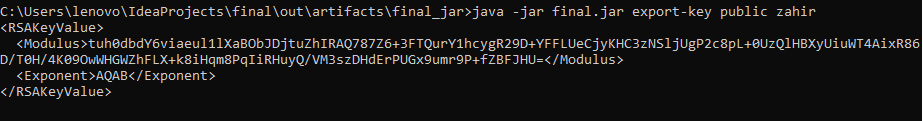
export-key <public|private> <name>

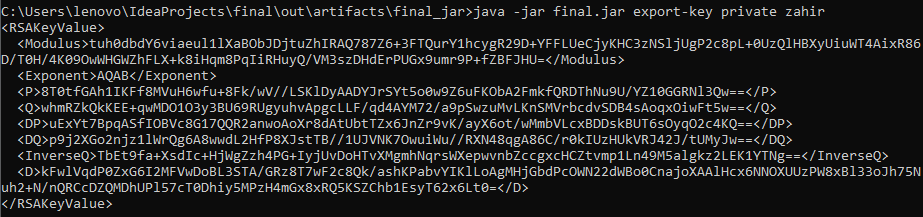
export-key <public|private> <name> [file]

Per shembull:

Ne rastin kur mungon argumenti I 3 atehere celesi publik/private i shfrytëzuesit do te shfaqet ne console :

export-key public zahir





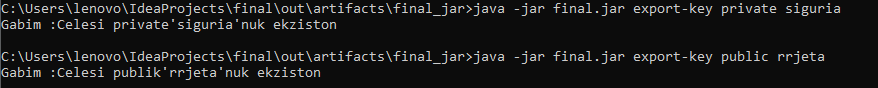
export-key private zahir

Kur japim argumentin e 3 ne te cilin ne caktojme shtegun qe ruhet celesi I export-uar:





Nese dojme qe te export-ojme nje qelese qofte private/ public qe fare nuk egziston :



# **Komanda import-key**

Importon çelësin publik ose privat të shfrytëzuesit nga shtegu i dhënë dhe e vendos në direktoriumin

e çelësave.

Ekzekutimi i komandes:

import-key <name> <path>

Per shembull:

Çelësi i importuar mund të jetë publik ose privat. Importimi I celesit publik:



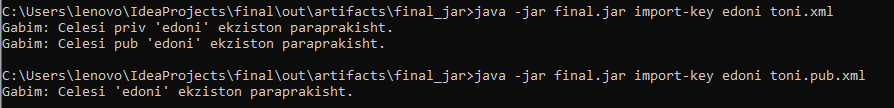


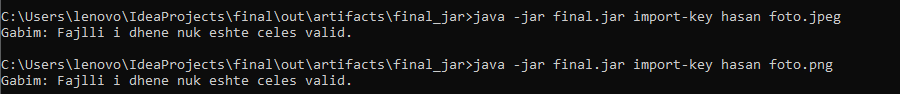
Nëse çelësi që po importohet është privat, atëherë automatikisht do ta gjeneroni edhe pjesën publike që t’i ruani të dyjat në direktoriumin e çelësave.





Nese file qe po dojme me importua ne te celsa te RSA , egziston paraprakisht:



Ne rastin ku si file importe nuk e kemi ni xml file po psh ni foto jpeg,png,gif atehere:

Nëse <path> fillon me http:// ose https://, atëherë dërgohet një GET request në URL <path> dhe do të merret trupi i përgjigjes si vlera e çelësit.



# **Komanda write-message**

E shkruan një mesazh të enkriptuar të dedikuar për një shfrytëzues.

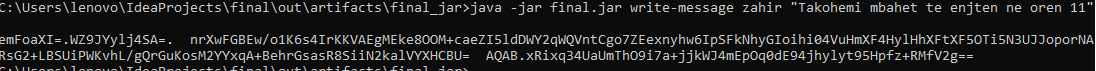
Ekzekutimi I komandes:

write-message <name> <message>

write-message <name> <message> [file]

Per shembull:

Ne rastin kur mungon argumenti I 3 atehere mesazhi I enkriptuar I nje shfrytëzuesit do te shfaqet ne console :



name = emri i marresit te mesazhit

pubkey = celesi publik i <name>

iv = random 8 bytes

key = random 8 bytes

encryptedKey = rsa(key, me pubkey)

encryptedMessage = des(message, me key)

part1 = base64.encode( utf8.encode(name) )

part2 = base64.encode(iv)

part3 = base64.encode(encryptedKey)

part4 = base64.encode(encryptedMessage)

return part1 + "." + part2 + "." + part3 + "." + part4

Ne argument[3] vendosim se ku do te ruhet mesazhi i kaluar neper metodat e me siperme:





Dime qe per enkriptim te RSA na duhet celesi publik I shfrytëzuesit.Ne rast qe nese nuk e kemi:



# **Komanda read-message**

E dekripton dhe e shfaq në console mesazhin e enkriptuar.

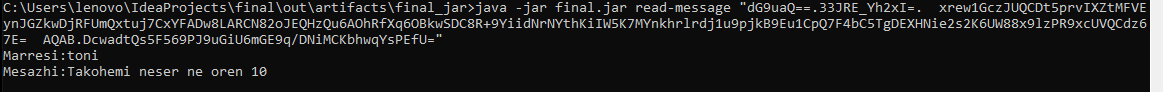
Ekzekutimi I komandat:

read-message <encrypted-message>

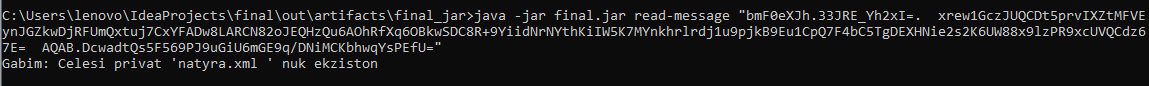
read-message [file]

Per shembull:

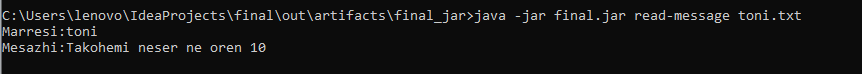
Nese si argument te pare ja japim mesazhin e marrur ne file qe e kemi ruajtur ne baze te formules se komandes “write-message”:



Kuptohet që për ta dekriptuar mesazhin nevojitet çelësi privat i shfrytëzuesit. Nëse mungon ky çelës do të shfaqet një mesazh gabimi:



Mund te provohet të lexohet argumenti si shteg i fajllit në të cilin gjendet mesazhi:



# **Udhezimet rreth ekzektumit te komandave**

# **Komanda create-user:** **create-user <name>**

# **Komanda delete-user: delete-user <name>**

# **Komanda export-key : export-key <public|private> <name>**

**export-key <public|private> <name> [file]**

# **Komanda import-key: import-key <name> <path>**

# **Komanda write-message: write-message <name> <message>**

**write-message <name> <message> [file]**

# **Komanda read-message: read-message <encrypted-message>**

**read-message [file]**