

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Διδάσκων: Επίκουρος Καθηγητής Γεώργιος Καμπουράκης

Εργαστηριακοί Συνεργάτες: Δημήτρης Παπαμαρτζιβάνος (ΥΔ), Αλέξανδρος Φακής (ΥΔ)

Ασφάλεια Δικτύων Υπολογιστών
και Τεχνολογίες Προστασίας της Ιδιωτικότητας

1^η Εργαστηριακή Άσκηση

Μέλη Ομάδας:

Πλεξίδα Μαρία 321/2011138

Λέρτας Γιώργος 321/2011084

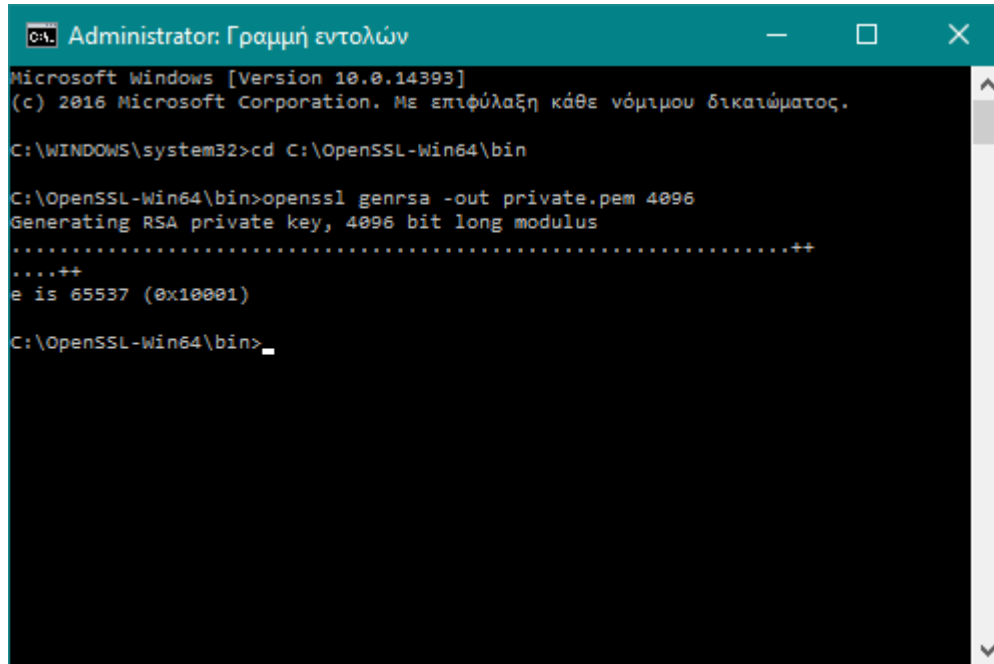
Κλιάρης Ευάγγελος 321/2011066

Δημιουργία Πιστοποιητικών

Όλες οι εντολές για τη δημιουργία του πιστοποιητικού γίνονται

Αρχικά παράγουμε το private key του server με την παρακάτω εντολή:

```
openssl genrsa -out private.pem 4096
```



```
Administrator: Γραμμή εντολών
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Με επιφύλαξη κάθε νόμιμου δικαιώματος.

C:\WINDOWS\system32>cd C:\OpenSSL-Win64\bin

C:\OpenSSL-Win64\bin>openssl genrsa -out private.pem 4096
Generating RSA private key, 4096 bit long modulus
.....++
.....++
e is 65537 (0x10001)

C:\OpenSSL-Win64\bin>
```

Παράγει ένα 4096-bit RSA κλειδί για τον server και το αποθηκεύει.

Με την παρακάτω εντολή δημιουργείται το πιστοποιητικό. (αρχείο πιστοποιητικό)

```
openssl req -new -x509 -days 730 -sha1 -key private.pem -out certificate.crt
```



Πληροφορίες για το πιστοποιητικό

Αυτό το πιστοποιητικό ρίζας της αρχής έκδοσης πιστοποιητικών δεν είναι αξιόπιστο. Για να είναι αξιόπιστο, εγκαταστήστε το στο χώρο αποθήκευσης των αξιόπιστων πιστοποιητικών ρίζας της Αρχής έκδοσης πιστοποιητικών.

Κάτοχος: icسد11138_icsд11084_icsд11066


Εκδόθηκε από: icسد11138_icsд11084_icsд11066

Έγκυρο από 15/10/2016 **έως** 15/10/2018

Εγκατάσταση πιστοποιητικού...

Δήλωση εκδότη

OK

 Πιστοποιητικό


×

Γενικά

Λεπτομέρειες

Διαδρομή πιστοποίησης

Διαδρομή πιστοποίησης

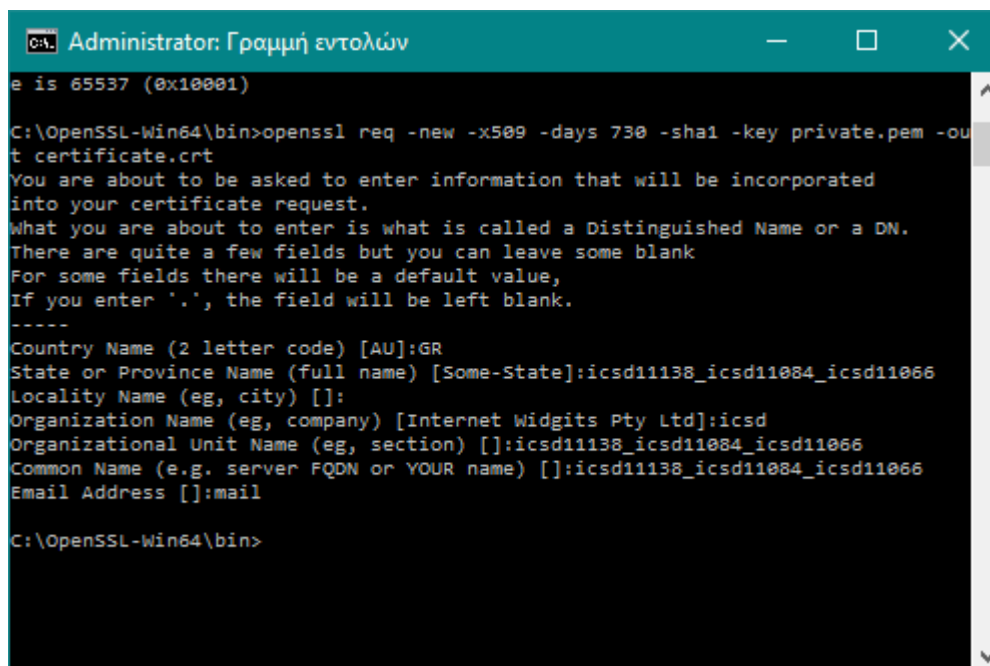
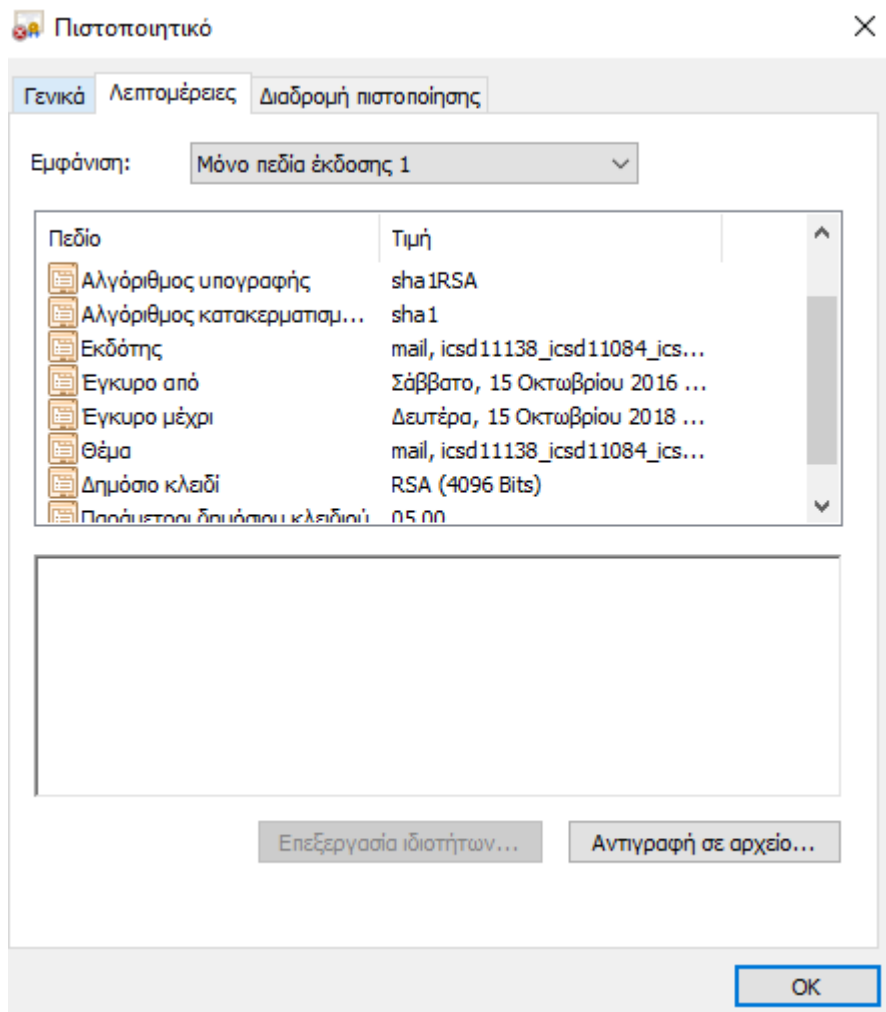
 icسد11138_icsد11084_icsد11066

Προβολή πιστοποιητικού

Κατάσταση πιστοποιητικού:

Αυτό το πιστοποιητικό ρίζας της αρχής έκδοσης πιστοποιητικών δεν είναι αξιόπιστο διότι δεν βρίσκεται στο χώρο αποθήκευσης των αξιόπιστων πιστοποιητικών ρίζας της αρχής έκδοσης πιστοποιητικών.

OK

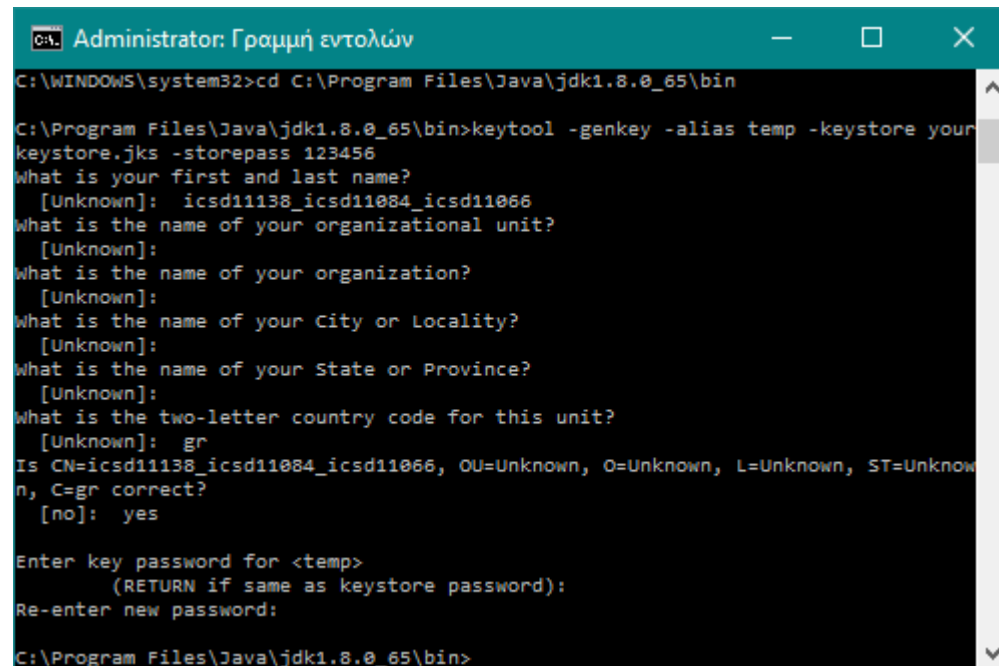


Επειδή η java δεν μπορεί να διαβάσει τα αρχεία .crt τα μετατρέπουμε μέσω των παρακάτω keytool εντολών σε .jks.

-yourkeystore: όνομα αρχείου

-123456: κωδικός συστήματος

keytool -genkey -alias temp -keystore yourkeystore.jks -storepass 123456



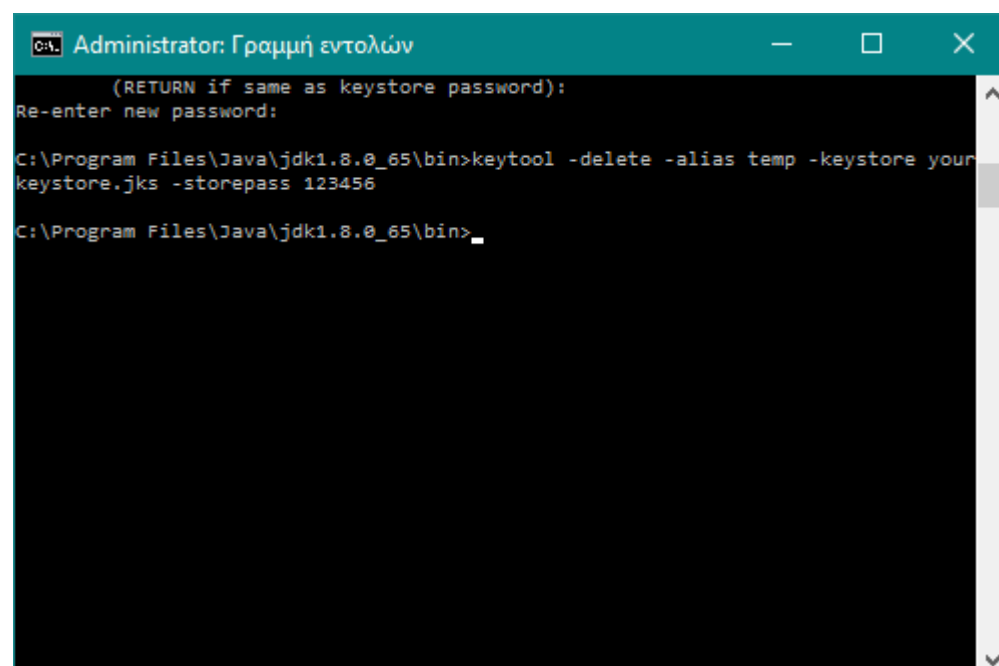
```
C:\WINDOWS\system32>cd C:\Program Files\Java\jdk1.8.0_65\bin

C:\Program Files\Java\jdk1.8.0_65\bin>keytool -genkey -alias temp -keystore your
keystore.jks -storepass 123456
What is your first and last name?
[Unknown]: icsd11138_icsd11084_icsd11066
What is the name of your organizational unit?
[Unknown]:
What is the name of your organization?
[Unknown]:
What is the name of your City or Locality?
[Unknown]:
What is the name of your State or Province?
[Unknown]:
What is the two-letter country code for this unit?
[Unknown]: gr
Is CN=icsd11138_icsd11084_icsd11066, OU=Unknown, O=Unknown, L=Unknown, ST=Unknow
n, C=gr correct?
[no]: yes

Enter key password for <temp>
(RETURN if same as keystore password):
Re-enter new password:

C:\Program Files\Java\jdk1.8.0_65\bin>
```

keytool -delete -alias temp -keystore yourkeystore.jks -storepass 123456

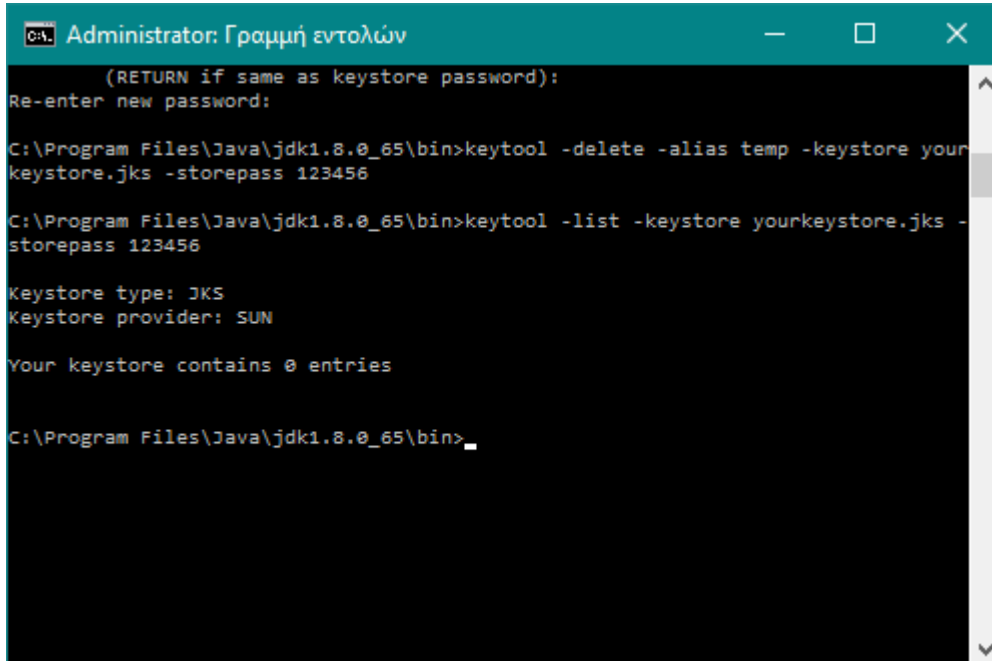


```
(RETURN if same as keystore password):
Re-enter new password:

C:\Program Files\Java\jdk1.8.0_65\bin>keytool -delete -alias temp -keystore your
keystore.jks -storepass 123456

C:\Program Files\Java\jdk1.8.0_65\bin>
```

keytool -list -keystore yourkeystore.jks -storepass 123456



```
Administrator: Γραμμή εντολών

(RETURN if same as keystore password):
Re-enter new password:

C:\Program Files\Java\jdk1.8.0_65\bin>keytool -delete -alias temp -keystore yourkeystore.jks -storepass 123456

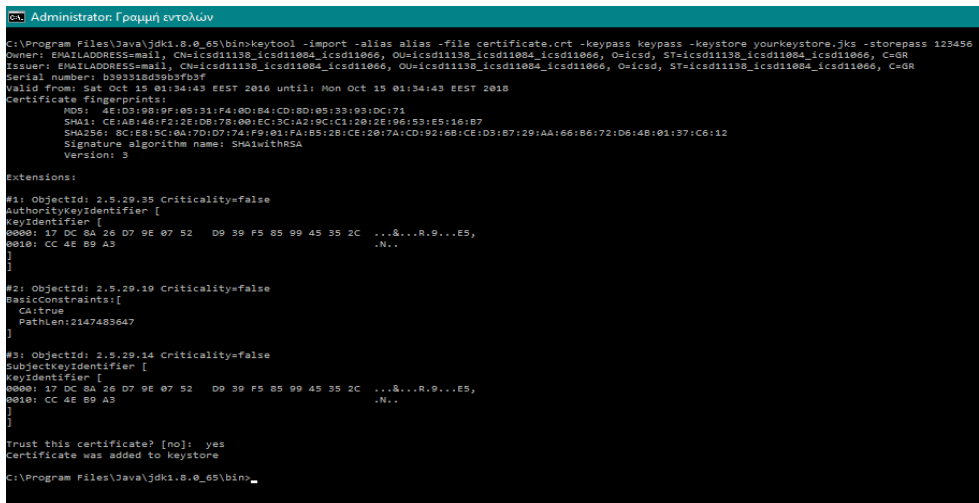
C:\Program Files\Java\jdk1.8.0_65\bin>keytool -list -keystore yourkeystore.jks -storepass 123456

Keystore type: JKS
Keystore provider: SUN

Your keystore contains 0 entries

C:\Program Files\Java\jdk1.8.0_65\bin>
```

keytool -import -alias alias -file certificate.crt -keypass keypass -keystore yourkeystore.jks -storepass 123456



```
Administrator: Γραμμή εντολών

C:\Program Files\Java\jdk1.8.0_65\bin>keytool -import -alias alias -file certificate.crt -keypass keypass -keystore yourkeystore.jks -storepass 123456
Owner: EMAILADDRESS@mail, CN=icsd11138_icsd11084_icsd11066, OU=icsd11138_icsd11084_icsd11066, O=icsd, ST=icsd11138_icsd11084_icsd11066, C=GR
Issuer: EMAILADDRESS@mail, CN=icsd11138_icsd11084_icsd11066, OU=icsd11138_icsd11084_icsd11066, O=icsd, ST=icsd11138_icsd11084_icsd11066, C=GR
Serial number: b39318d9b59b3f
Valid from: Sat Oct 15 01:34:43 EEST 2016 until: Mon Oct 15 01:34:43 EEST 2018
Certificate fingerprints:
    MD5: 4E:D3:08:0F:05:31:F4:0D:84:CD:8D:05:33:93:DC:71
    SHA1: CE:A8:46:F2:2E:0B:7B:00:EC:3C:A2:9C:C1:20:2E:96:53:E5:16:07
    SHA256: 8C:E8:5C:0A:70:D7:74:F9:01:FA:85:2B:CE:20:7A:CD:92:6B:CE:D3:87:29:AA:66:B6:72:D6:48:01:37:C6:12
    Signature algorithm name: SHA1withRSA
    Version: 3
Extensions:
    #1: ObjectID: 2.5.29.35 Criticality=false
    AuthorityKeyIdentifier [
        KeyIdentifier [
            0000: 17 DC BA 26 D7 9E 07 52 D9 39 F5 85 99 45 35 2C ...&...R.0...E5,
            0010: CC 4E B9 A3
        ]
    ]
    #2: ObjectID: 2.5.29.19 Criticality=false
    BasicConstraints [
        CA:true
        PathLen:2147483647
    ]
    #3: ObjectID: 2.5.29.14 Criticality=false
    SubjectKeyIdentifier [
        KeyIdentifier [
            0000: 17 DC BA 26 D7 9E 07 52 D9 39 F5 85 99 45 35 2C ...&...R.0...E5,
            0010: CC 4E B9 A3
        ]
    ]
Trust this certificate? [no]: yes
Certificate was added to keystore

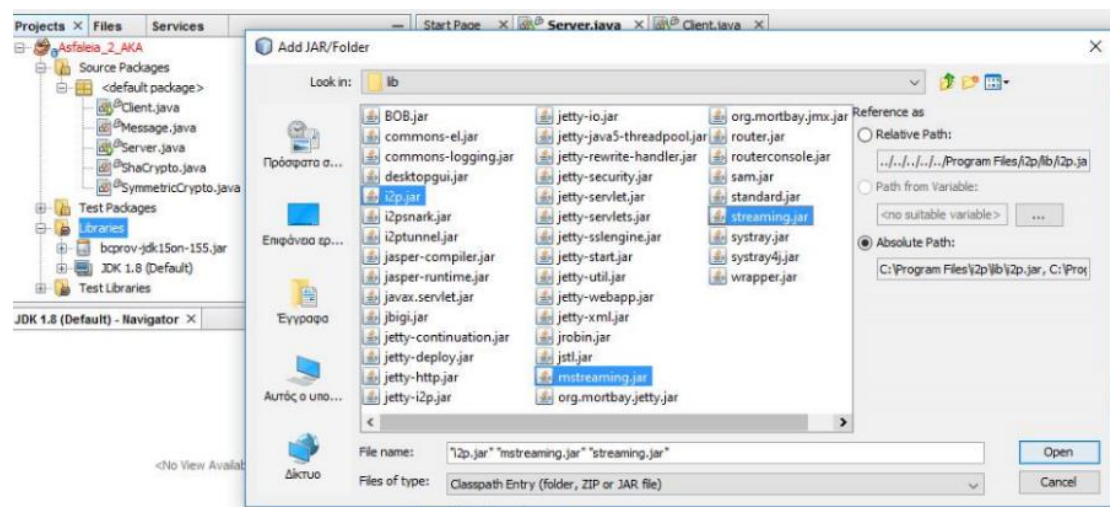
C:\Program Files\Java\jdk1.8.0_65\bin>
```

Πηγή: <http://stackoverflow.com/questions/4022604/java-how-to-obtain-keystore-file-for-a-certification-crt-file>

Για το I2P:

Έπρεπε να κατεβάσουμε τις παρακάτω βιβλιοθήκες.

Πηγή: <https://geti2p.net/tr/get-involved/develop/applications>



Θεωρητικές

1. Το AKA πρωτόκολλο εξασφαλίζει από επιθέσεις τύπου spoofing και security downgrade; Αν ναι, με ποιόν τρόπο; Αν όχι, πως θα μπορούσαν να αποφευχθούν τέτοιου είδους επιθέσεις; Αναπτύξτε με συντομία.

Αρχικά, spoofing είναι η τεχνική όπου αποκτούμε πρόσβαση σε υπολογιστές με την δημιουργία πακέτων TCP/IP, χρησιμοποιώντας τη διεύθυνση και τα στοιχεία κάποιου άλλου αξιόπιστου. Τα routers χρησιμοποιούν την διεύθυνση της IP προορισμού ώστε να διαδώσουν τα πακέτα μέσω διαδικτύου αλλάζοντας εικονικά την διεύθυνση της IP πηγής. Αυτή η διεύθυνση χρησιμοποιείται μόνο από το μηχάνημα προορισμού όταν απαντά πίσω στη πηγή.

Το πρωτοκόλλο AKA εξασφαλίζει από επιθέσεις spoofing διότι ο server στέλνει ένα τυχαίο αλφαριθμητικό ως cookie στον client και περιμένει να το πάρει πίσω στο επόμενο μήνυμα του client.

Downgrade: Σε αυτή την κατηγορία επιθέσεων (επιθέσεις υποβίβασης) ο επιτιθέμενος προσπαθεί να αναγκάσει τα δύο μέρη να επιλέξουν αδύναμες κρυπτογραφικές σουίτες.

Στην περιπτωσή μας αν η Alice επιλέξει π.χ. 2 αλγόριθμους και ο ένας από τους δύο είναι αδύναμος τότε ο επιτιθέμενος μπορεί να "κόψει" το δυνατό κ να σταλθεί έτσι ο αδύναμος στο server.

Συνεπώς, δεν εξασφαλίζεται πλήρως από το security downgrade, παρόλο που όπως βλέπουμε στο σχήμα της εκφώνησης (βήμα 5) ότι υπάρχει εν μέρη security downgrade.

2. Το πρωτόκολλο εξασφαλίζει την επικαιρότητα (freshness) της συνόδου; Αν ναι, με ποιόν τρόπο; Αν όχι, πως μπορεί να εξασφαλιστεί αυτή η ιδιότητα;

Στο πρωτόκολλο AKA υπάρχει η επικαιρότητα (freshness) διότι χρησιμοποιεί συνεχώς διαφορετικό κλειδί και όχι το ίδιο

3. Μπορείτε να βελτιώσετε τη λειτουργία του πρωτοκόλλου; Ποια θεωρείτε ότι είναι τα αδύναμα χαρακτηριστικά του και τι θα προτείνατε για τη βελτίωση του;

Για τη βελτίωση του πρωτοκόλλου θα μπορούσαμε να αλλάξουμε τον τρόπο όπου στέλνουμε τις σουίτες. Αν οι σουίτες στέλνονταν κρυπτογραφημένες δεν θα υπήρχε ο κίνδυνος του downgrade.

4. Μελετήστε τη λειτουργία του δικτύου ανωνυμίας I2P και του δικτύου ανωνυμίας TOR. Ποιες διαφορές παρουσιάζουν; Ποιο είναι ασφαλέστερο και γιατί;

- Στον Tor όταν θέλουμε να επισκεφτούμε το ίντερνετ μπορούμε ελεύθερα ενώ σε αντίθεση στον I2P χρειαζόμαστε έναν outbound proxy.
- Στον I2P σπάει η διαδρομή σε 2 κομμάτια ενώ στο tor γίνεται με το ίδιο κανάλι.
- Ο Tor έχει σχεδιαστεί με μεγάλους κόμβους εξόδου σε αντίθεση με τον I2P.
- Ο Tor υποστηρίζει C ενώ ο I2P Java.

- Στο I2P τα δεδομένα στέλνονται σε πακέτα ενώ οι clients του Tor καθορίζουν τυχαία το μονοπάτι της σύνδεσης.
- Το Tor κάνει πιο αποτελεσματική χρήση μνήμης.
- Οι κόμβοι του Tor έχουν χαμηλό εύρος ζώνης.
- Ο κεντρικό έλεγχος στο Tor μειώνει την πολυπλοκότητα σε κάθε κόμβο και μπορεί να αντιμετωπίσει αποτελεσματικά τις επιθέσεις Sybil
- Στο Tor ένας πυρήνας υψηλής χωρητικότητας κόμβου παρέχει υψηλότερη απόδοση και χαμηλότερη αφάνεια.
- Το I2P έχει σχεδιαστεί για κρυφές υπηρεσίες οι οποίες είναι πιο γρήγορες έναντι του Tor.
- Στο I2P υπάρχει πλήρης κατανομής και αυτο οργάνωση.
- Οι διαδρομές (tunnels) στο I2P έχουν μικρή διάρκεια ζωής σε αντίθεση με του Tor.
- Το I2P έχει ενσωματωμένο μηχανισμό αυτόματης ενημέρωσης ενώ το Tor όχι.

Η πιο βασική διαφορά του Tor από τον I2P είναι πως στο Tor το πακέτο κρυπτογραφείται επανειλημμένα σε κάθε κόμβο που διέρχεται μέχρι να φτάσει στο τελικό κόμβο όπου αποκρυπτογραφείται και προωθείται στον τελικό χρήστη. Αυτό αποτρέπει τους ενδιαμέσους κόμβους να μάθουν την προέλευση, τον προορισμό και το περιεχόμενο του μηνύματος. Συνεπώς αυτό το κάνει ασφαλέστερο έναντι του I2P.

Printscreen Εκτέλεσης Προγράμματος:

```

I2P Service
jvm 1 | Copied C:\Program Files\i2p\i2ptunnel.config
jvm 1 | Copied C:\Program Files\i2p\systay.config
jvm 1 | Copied C:\Program Files\i2p\eeppsite\jetty.xml with modifications
jvm 1 | Copied C:\Program Files\i2p\eeppsite\jetty-ssl.xml with modifications
jvm 1 | Copied C:\Program Files\i2p\eeppsite\contexts\base-context.xml with modifications
jvm 1 | Copied C:\Program Files\i2p\eeppsite\contexts\cgi-context.xml with modifications
jvm 1 | Copied C:\Program Files\i2p\clients.config with modifications
jvm 1 | Successfully copied data files to new user directory C:\Users\Mairi\AppData\Roaming\I2P
jvm 1 | INFO: Native CPUID library jcpuid-x86-windows.dll loaded from resource
jvm 1 | trying to load resource: jbigi-windows-coreisbr_64.dll
jvm 1 | Loaded library: jar:file:/C:/Program%20Files/i2p/lib/jbigi.jar!/jbigi-windows-coreisbr_64.dll
jvm 1 | INFO: Native BigInteger library jbigi-windows-coreisbr_64.dll loaded from resource
jvm 1 | Reseed start
jvm 1 | Reseeding from https://reseed.i2p.vzaws.com:8443/i2pseeds.su3
jvm 1 | INFO: Addressbook directory addressbook created
jvm 1 | INFO: 75 files extracted to C:\Users\Mairi\AppData\Local\Temp\i2p-0oSWnBuT.tmp\reseeds-859866299
jvm 1 | Reseed got 75 router infos from https://reseed.i2p.vzaws.com:8443/i2pseeds.su3 with 0 errors
jvm 1 | Reseeding from https://uk.reseed.i2p2.no:444/i2pseeds.su3
jvm 1 | INFO: 60 files extracted to C:\Users\Mairi\AppData\Local\Temp\i2p-0oSWnBuT.tmp\reseeds-286034631
jvm 1 | Reseed got 60 router infos from https://uk.reseed.i2p2.no:444/i2pseeds.su3 with 0 errors
jvm 1 | Reseed complete, 135 received
jvm 1 | Warning: Unable to reach any of the NTP servers [0.gr.pool.ntp.org, 1.gr.pool.ntp.org, 2.gr.pool.ntp.org, 0.europe.pool.ntp.org, 1.europe.pool.ntp.org, 2.europe.pool.ntp.org, 0.pool.ntp.org, 1.pool.ntp.org, 2.pool.ntp.org] - network disconnected? Or set time.sntpServerList=myserver1.com,myserver2.com in advanced configuration.

```

```
Output X
Asfaleia_2_AKA (run-single) X Asfaleia_2_AKA (run-single) #2 X

ant -f C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9 -Djavac.includes=Server.java -Dnb.internal.action.name=run.sin
init:
Deleting: C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9\build\build-jar.properties
Compiling 1 source file to C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9\build\classes
compile-single:
run-single:
vS01g5Y-yA5R-EnaIhz5uNielXN668IZNWRIm0ZnLSzFuniDiv0S3THxOWm8ZOxrGKM6wx4ZAQ6oD6qJNh6hv8hMmysYTwEWy-9Cck4Qsp7wV-MP0Qtumuyg6aM-iJFqvcKpr2HozW
Hello
16
16
3c2b2d61f77e8e585a69222ec61f169d24066d4e3590ed2ec91238e8032a42a4
[B@2f6c85dc
[B@56fbca51
```

```
Output X
Asfaleia_2_AKA (run-single) X Asfaleia_2_AKA (run-single) #2 X

ant -f C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9 -Djavac.includes=Client.java -Dnb.internal.action.name=run.sin
init:
Deleting: C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9\build\build-jar.properties
Compiling 1 source file to C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9\build\classes
Note: C:\Users\Mairi\Documents\NetBeansProjects\NetSec_2016Under_Team9\src\Client.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
compile-single:
run-single:
Please enter a Destination:
vS01g5Y-yA5R-EnaIhz5uNielXN668IZNWRIm0ZnLSzFuniDiv0S3THxOWm8ZOxrGKM6wx4ZAQ6oD6qJNh6hv8hMmysYTwEWy-9Cck4Qsp7wV-MP0Qtumuyg6aM-iJFqvcKpr2HozW

BUILD SUCCESSFUL (total time: 1 minute 11 seconds)
```