A,B) תהליך יצירת הKEY STOR עבור אלגוריתם RSA :

בעזרת הפקודות:

keytool -genkeypair -alias "barPair" -keyalg "RSA" -keypass "barfrimet" -keystore "bar.keystore" -storepass "barFrimet" -storetype "jks"

keytool -genkeypair -alias "alice" -keyalg "RSA" -keypass "alicebobo" -keystore "alice.keystore" -storepass "aliceBob" -storetype "jks"

```
C:\Users\Bar\java_pro>keytool -genkeypair -alias "bar" -keyalg "RSA" -keypass "barfrimet" -keystore "bar.keystore" -storepass "barFrimet" -storetype "jks"
What is your first and last name?
  [Unknown]:  bar
What is the name of your organizational unit?
  [Unknown]:  IDC
What is the name of your organization?
  [Unknown]:  IDC
What is the name of your City or Locality?
  [Unknown]:  Herzliya
What is the name of your State or Province?
  [Unknown]:  Israel
What is the two-letter country code for this unit?
  [Unknown]:  IL
Is CN=bar, OU=IDC, O=IDC, L=Herzliya, ST=Israel, C=IL correct?
  [no]:  yes


Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standard format using "keytool -importkeystore -srckeystore bar.keys
tore -destkeystore bar.keystore -deststoretype pkcs12".
```

```
C:\Users\Bar\java_pro>keytool -genkeypair -alias "alice" -keyalg "RSA" -keypass "alicebobo" -keystore "alice.keystore" -storepass "aliceBob" -storetype "jks"
What is your first and last name?
  [Unknown]:  alice
What is the name of your organizational unit?
  [Unknown]:  IDC
What is the name of your organization?
  [Unknown]:  somewhere
What is the name of your City or Locality?
  [Unknown]:  somewhere
What is the name of your State or Province?
  [Unknown]:  il
What is the two-letter country code for this unit?
  [Unknown]:  il
Is CN=alice, OU=IDC, O=someWhere, L=someWhere, ST=il, C=il correct?
  [no]:  yes


Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standard format using "keytool -importkeystore -srckeystore alice.ke
ystore -destkeystore alice.keystore -deststoretype pkcs12".
```

C) ראשית ניצור Self-Signed Certificate

בעזרת הפקודות:

keytool -exportcert -alias "bar" -file "bar.crt" -storepass "barFrimet" -keystore "bar.keystore"

keytool -exportcert -alias "alice" -file "alice.crt" -storepass "aliceBob" -keystore "alice.keystore"

```
C:\Users\Bar\java_pro>keytool -exportcert -alias "bar" -file "bar.crt" -storepass "barFrimet" -keystore "bar.keystore"
Certificate stored in file <bar.crt>

Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standard format using "keytool -importkeystore -srckeystore bar.keys
tore -destkeystore bar.keystore -deststoretype pkcs12".

C:\Users\Bar\java_pro>keytool -exportcert -alias "alice" -file "alice.crt" -storepass "aliceBob" -keystore "alice.keystore"
Certificate stored in file <alice.crt>

Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standard format using "keytool -importkeystore -srckeystore alice.ke
ystore -destkeystore alice.keystore -deststoretype pkcs12".

C:\Users\Bar\java_pro>
```

וכעת יש לנו תעודות של שני הצדדים נעביר אותו לצד השני על מנת לקבל certificate Trusted

בעזרת הפקודות:

keytool -importcert -alias "barCrt" -file "bar.crt" -storepass "aliceBob" -keystore "alice.keystore"

keytool -importcert -alias "aliceCrt" -file "alice.crt" -storepass "barFrimet" -keystore "bar.keystore"

```
C:\Users\Bar\java_pro>keytool -importcert -alias "barCrt" -file "bar.crt" -storepass "aliceBob" -keystore "alice.keystore"
Owner: CN=bar, OU=IDC, O=IDC, L=Herzliya, ST=Israel, C=IL
Issuer: CN=bar, OU=IDC, O=IDC, L=Herzliya, ST=Israel, C=IL
Serial number: 560ea08
Valid from: Sun Jan 05 03:15:48 IST 2020 until: Sat Apr 04 04:15:48 IDT 2020
Certificate fingerprints:
         SHA1: DF:56:05:D7:C8:58:F8:09:DF:7E:60:A6:4B:56:1F:92:41:32:40:F7
         SHA256: 45:84:C0:CC:49:39:4F:B0:17:CC:02:44:54:59:25:19:70:AC:15:BF:7F:44:C6:37:28:36:D0:68:28:2E:6B:25
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 64 DF 87 2F 3C 48 EB 0A   66 33 CB DD 7F F7 C2 48  d../<H..f3.....H
0010: 86 71 9D 7C                                        .q..
]
]

Trust this certificate? [no]:  yes
Certificate was added to keystore

Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standard format using "keytool -importkeystore -srckeystore alice.k
ystore -destkeystore alice.keystore -deststoretype pkcs12".
```

```
C:\Users\Bar\java_pro>keytool -importcert -alias "aliceCrt" -file "alice.crt" -storepass "barFrimet" -keystore "bar.keystore"
Owner: CN=alice, OU=IDC, O=someWhere, L=someWhere, ST=il, C=il
Issuer: CN=alice, OU=IDC, O=someWhere, L=someWhere, ST=il, C=il
Serial number: 56b60fc6
Valid from: Sun Jan 05 03:14:42 IST 2020 until: Sat Apr 04 04:14:42 IDT 2020
Certificate fingerprints:
         SHA1: 3A:F1:37:68:98:0C:19:C1:8D:C4:E6:D8:64:9B:75:60:F3:5C:62:C0
         SHA256: 72:1C:A0:A7:0B:48:3C:0A:C6:AF:B3:F6:62:34:1B:AC:5A:E3:20:96:B6:16:26:B2:ED:AE:1E:5E:96:02:30:59
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 79 FC 92 62 48 D1 F8 D0   DF 6E 5A 87 77 04 6F EA  y..bH....nZ.w.o.
0010: 4F 57 BF 75                                        OW.u
]
]

Trust this certificate? [no]:  yes
Certificate was added to keystore

Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standard format using "keytool -importkeystore -srckeystore bar.keys
tore -destkeystore bar.keystore -deststoretype pkcs12".
```

אופן הרצת התוכנית: כל הקבצי ה- JAVA מצורפים בקובץ ה- JAR אשר ממנה נריץ את התוכנית כאשר הפרמטרים להרצת התוכנית הם:

1) ה- KeyStore שממו נרצה להצפין את המידע.
2) הסיסמה של ה- KeyStore.
3) ה- Alias של המפתח שב- KeyStore .
4) הסיסמה ל- Alias.
5) ה- certificate.
6) המצב של פועלה אשר אנו נרצה: e = עבור הצפנת הקובץ ו- d = עבור הפיענוח.

הסבר על הקוד:

ראשית בחרתי באלגוריתם AES אשר התבקשנו להשתמש בו שהוא מספק לנו אלגוריתם הצפנה סימטרי , כאשר חסרונו של האלגוריתם טמון בכך שראשית על 2 הצדדים מראש להעביר בין אחד לשני מפתח שאינו מאובטח דיו ולכן אנו נצפין מפתח זה בעזרת אלגורית RSA אשר הוא אלגורית א-סימטרי שמטרתו לספק תעבורה מאובטחת של מידע ובכך אני מיישם את השיטה של ההצפנה היברידית שהמפתח הסימטרי יוצפן בעזרת אלרגוריתם א-סימטרי ובכך אנו מקשים על צד שלישי לפענח את המפתח הסימטרי וגם משתמשים ביכולת של המפתח הסימטרי אשר ההצפנה איתו יעילה יותר וגם נעזר בו עם המפתח הא-סימטרי אשר הוא פחות יעיל אך מאובטח יותר.
בנוסף השתמשתי במוד CBC אשר בעזרת ה-IV האקראי (אשר משתמשים בו רק בתחילת ההצפנה) המידע ישמר באפן מאובטח מבחינה קריפטוגרפית. אומנם מוד זה אינו מיתן להצפנה מקבילית אך מאפשר פיענוח מקבילי בעזרת מספר מעבדים .

מצב ההצפנה(e):

ראשית הגרלנו IV פסדו אקראי בעזרת SecureRandom .
בעזרת KeyGenerator יוגרל מפתח נוסף אשר הוא יהיה מפתח סימטרי.

כעת נעזר ב- MessageDigest שבעזרתו שחשב את טבלת ה- hash ובעזרת המפתח הפרטי של המצפין נחתום על הטבלה של ה- hash בעזרת Singnature.
כעת על מנת להצפין את הקובץ הרצוי לencrypted.txt נעזר ב- IV הקיים ואבנה את האובייקט Cipher אשר בעזרתו עובדת האלגורית AES שבחרנו כעת נצפינו בעזרת CipherOutputStream .
כעת לאחר ההצפנה נעזר ב- Singnature שבעזרתו נקח את המפתח הפומבי מהצד השני נצפין אותו אשר כבר הוגדר מראש בתעודה ונצפין אותו באלגוריתם א-סימטרי על מנת לאבטח את המפתח הפומבי.
כעת נכניס את החתימה הדיגיטלית, המפתח המוצפן ואת ה- IV בקובץ conf.properties אשר הוא הקובץ הקונפיגורציה שלנו.

מצב ההצפנה(d):
ראשית נפתח את קובץ הקונפיגורציה שלנו(conf.properties) אשר בעזרתו אשמור את הנתונים הנחוצים .
כעת נעזר ב- Cipher אשר הוא יפענח עבורנו את המפתח ה- RSA שעל המפתח הסימטרי המוצפן של הצד שמנסה לפענח את הקובץ.
עכשיו אנו נעזר ב- Cipher חדש וב- IV אשר מוכר עבור הצד המפענח שבעזרתו ובעזרת CipherInputStream נפענח את הקובץ המוצפן ונזין את המידע המפוענח לקובץ המפוענח(decrypted.txt).
כעת עלינו לבדוק שהתחימה הדיגטלית אכן נכונה ובכך נווה שהקובץ אינו שונה במהלך הדרך לצד המפענח זאת אעשה בעזרת hash ובעזרת ה- Signature (בזהה לאופן החתימה על הקובץ המוצפן) זאת בעזרץ המפתח הפומבי של הצד המצפין אשר כתוב בתות התעודה הדיגיטלית של הצד המפענח .