INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

TEMAS SELECTOS DE CRIPTOGRAFÍA

“CURVA ELIPTICA CRIPTOGRAFICA Y OPENSSL”

GRUPO:

7CM2

PRESENTAN:

CORDOVA PICHARDO FRANCISCO UZIEL

GARCIA JIMENEZ JUAN CARLOS

MONTES LOZADA CRISTOFER

PROFESOR:

DIAZ SANTIAGO SANDRA

FECHA DE REALIZACION: 02 DE MAYO DEL 2023

FECHA DE ENTREGA: 02 DE MAYO DEL 2023

Cristofer Montes Lozada.

1. List the elliptic curves available in OpenSSL. Identify if there are curves from the standard SP800-186 in Openssl.

```
thegentleman@PC: ~
thegentleman@PC:~$ openssl ecparam -list_curves
  secp112r1 : SECG/WTLS curve over a 112 bit prime field
  secp112r2 : SECG curve over a 112 bit prime field
  secp128r1 : SECG curve over a 128 bit prime field
  secp128r2 : SECG curve over a 128 bit prime field
  secp160k1 : SECG curve over a 160 bit prime field
  secp160r1 : SECG curve over a 160 bit prime field
  secp160r2 : SECG/WTLS curve over a 160 bit prime field
  secp192k1 : SECG curve over a 192 bit prime field
  secp224k1 : SECG curve over a 224 bit prime field
  secp224r1 : NIST/SECG curve over a 224 bit prime field
  secp256k1 : SECG curve over a 256 bit prime field
  secp384r1 : NIST/SECG curve over a 384 bit prime field
  secp521r1 : NIST/SECG curve over a 521 bit prime field
  prime192v1: NIST/X9.62/SECG curve over a 192 bit prime field
  prime192v2: X9.62 curve over a 192 bit prime field
  prime192v3: X9.62 curve over a 192 bit prime field
  prime239v1: X9.62 curve over a 239 bit prime field
  prime239v2: X9.62 curve over a 239 bit prime field
  prime239v3: X9.62 curve over a 239 bit prime field
  prime256v1: X9.62/SECG curve over a 256 bit prime field
  sect113r1 : SECG curve over a 113 bit binary field
  sect113r2 : SECG curve over a 113 bit binary field
  sect131r1 : SECG/WTLS curve over a 131 bit binary field
  sect131r2 : SECG curve over a 131 bit binary field
  sect163k1 : NIST/SECG/WTLS curve over a 163 bit binary field
  sect163r1 : SECG curve over a 163 bit binary field
  sect163r2 : NIST/SECG curve over a 163 bit binary field
  sect193r1 : SECG curve over a 193 bit binary field
  sect193r2 : SECG curve over a 193 bit binary field
  sect233k1 : NIST/SECG/WTLS curve over a 233 bit binary field
  sect233r1 : NIST/SECG/WTLS curve over a 233 bit binary field
  sect239k1 : SECG curve over a 239 bit binary field
  sect283k1 : NIST/SECG curve over a 283 bit binary field
  sect283r1 : NIST/SECG curve over a 283 bit binary field
  sect409k1 : NIST/SECG curve over a 409 bit binary field
  sect409r1 : NIST/SECG curve over a 409 bit binary field
  sect571k1 : NIST/SECG curve over a 571 bit binary field
```

```
thegentleman@PC: ~
  sect571k1 : NIST/SECG curve over a 571 bit binary field
  sect571r1 : NIST/SECG curve over a 571 bit binary field
  c2pnb163v1: X9.62 curve over a 163 bit binary field
  c2pnb163v2: X9.62 curve over a 163 bit binary field
  c2pnb163v3: X9.62 curve over a 163 bit binary field
  c2pnb176v1: X9.62 curve over a 176 bit binary field
  c2tnb191v1: X9.62 curve over a 191 bit binary field
  c2tnb191v2: X9.62 curve over a 191 bit binary field
  c2tnb191v3: X9.62 curve over a 191 bit binary field
  c2pnb208w1: X9.62 curve over a 208 bit binary field
  c2tnb239v1: X9.62 curve over a 239 bit binary field
  c2tnb239v2: X9.62 curve over a 239 bit binary field
  c2tnb239v3: X9.62 curve over a 239 bit binary field
  c2pnb272w1: X9.62 curve over a 272 bit binary field
  c2pnb304w1: X9.62 curve over a 304 bit binary field
  c2tnb359v1: X9.62 curve over a 359 bit binary field
  c2pnb368w1: X9.62 curve over a 368 bit binary field
  c2tnb431r1: X9.62 curve over a 431 bit binary field
  wap-wsg-idm-ecid-wtls1: WTLS curve over a 113 bit binary field
  wap-wsg-idm-ecid-wtls3: NIST/SECG/WTLS curve over a 163 bit binary field
  wap-wsg-idm-ecid-wtls4: SECG curve over a 113 bit binary field
  wap-wsg-idm-ecid-wtls5: X9.62 curve over a 163 bit binary field
  wap-wsg-idm-ecid-wtls6: SECG/WTLS curve over a 112 bit prime field
  wap-wsg-idm-ecid-wtls7: SECG/WTLS curve over a 160 bit prime field
  wap-wsg-idm-ecid-wtls8: WTLS curve over a 112 bit prime field
  wap-wsg-idm-ecid-wtls9: WTLS curve over a 160 bit prime field
  wap-wsg-idm-ecid-wtls10: NIST/SECG/WTLS curve over a 233 bit binary field
  wap-wsg-idm-ecid-wtls11: NIST/SECG/WTLS curve over a 233 bit binary field
  wap-wsg-idm-ecid-wtls12: WTLS curve over a 224 bit prime field
  Oakley-EC2N-3:
        IPSec/IKE/Oakley curve #3 over a 155 bit binary field.
        Not suitable for ECDSA.
        Questionable extension field!
  Oakley-EC2N-4:
        IPSec/IKE/Oakley curve #4 over a 185 bit binary field.
        Not suitable for ECDSA.
        Questionable extension field!
  brainpoolP160r1: RFC 5639 curve over a 160 bit prime field
```

```
  brainpoolP160r1: RFC 5639 curve over a 160 bit prime field
  brainpoolP160t1: RFC 5639 curve over a 160 bit prime field
  brainpoolP192r1: RFC 5639 curve over a 192 bit prime field
  brainpoolP192t1: RFC 5639 curve over a 192 bit prime field
  brainpoolP224r1: RFC 5639 curve over a 224 bit prime field
  brainpoolP224t1: RFC 5639 curve over a 224 bit prime field
  brainpoolP256r1: RFC 5639 curve over a 256 bit prime field
  brainpoolP256t1: RFC 5639 curve over a 256 bit prime field
  brainpoolP320r1: RFC 5639 curve over a 320 bit prime field
  brainpoolP320t1: RFC 5639 curve over a 320 bit prime field
  brainpoolP384r1: RFC 5639 curve over a 384 bit prime field
  brainpoolP384t1: RFC 5639 curve over a 384 bit prime field
  brainpoolP512r1: RFC 5639 curve over a 512 bit prime field
  brainpoolP512t1: RFC 5639 curve over a 512 bit prime field
  SM2       : SM2 curve over a 256 bit prime field
thegentleman@PC:~$
```

Se identificaron las curvas P-224, P-384 y P-521, nombradas como secp[…] y etiquetadas como NIST/SECG curve […].

2. Choose three different elliptic curves over a prime field and generate the EC parameters in a .pem file. Also store the keys in .der files

```
thegentleman@PC: ~/Escritorio/Crypto
thegentleman@PC:~/Escritorio/Crypto$ openssl ecparam -name secp521r1 -out secp521r1.pem
thegentleman@PC:~/Escritorio/Crypto$ openssl ecparam -in secp521r1.pem -genkey -noout -out Cris_PrK.der
thegentleman@PC:~/Escritorio/Crypto$ openssl ec -in Cris_PrK.der -pubout -out Cris_PuK.der
read EC key
writing EC key
thegentleman@PC:~/Escritorio/Crypto$
```

Con el primer comando se guardan los parámetros de la curva P-521 en el archivo PEM.

Con el segundo comando se usó el archivo PEM con la curva P-521 para crear la llave privada de Cristofer en un archivo DER.

Con el tercer comando se usó el archivo DER que contiene la llave privada de Cristofer para crear la llave pública de Cristofer en un archivo DER.

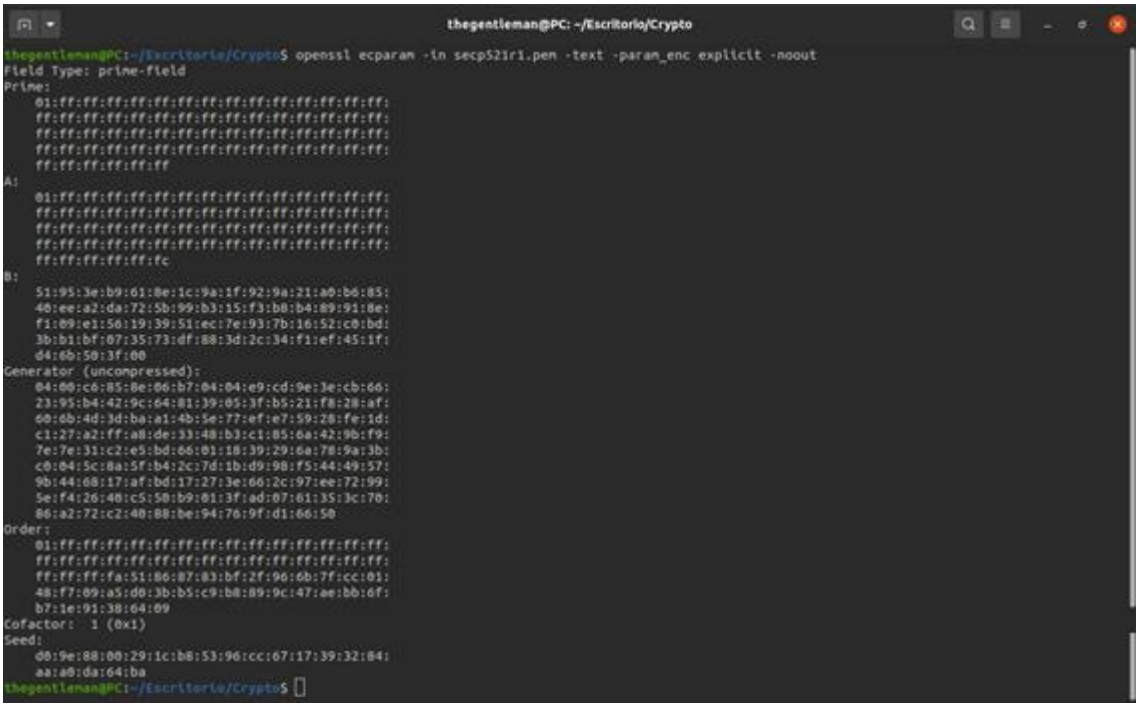
3. Choose an elliptic curve and generate a pair of keys (private and public) for each member in the team. Store each key in a different .pem file

```
thegentleman@PC: ~/Escritorio/Crypto
thegentleman@PC:~/Escritorio/Crypto$ openssl ecparam -in secp521r1.pem -genkey -noout -out Cris_PrK.pem
thegentleman@PC:~/Escritorio/Crypto$ openssl ec -in Cris_PrK.pem -pubout -out Cris_PuK.pem
read EC key
writing EC key
thegentleman@PC:~/Escritorio/Crypto$
```

Con el primer comando se usó el archivo PEM con la curva P-521 para generar las claves pública y privada de Cristofer en formato PEM.

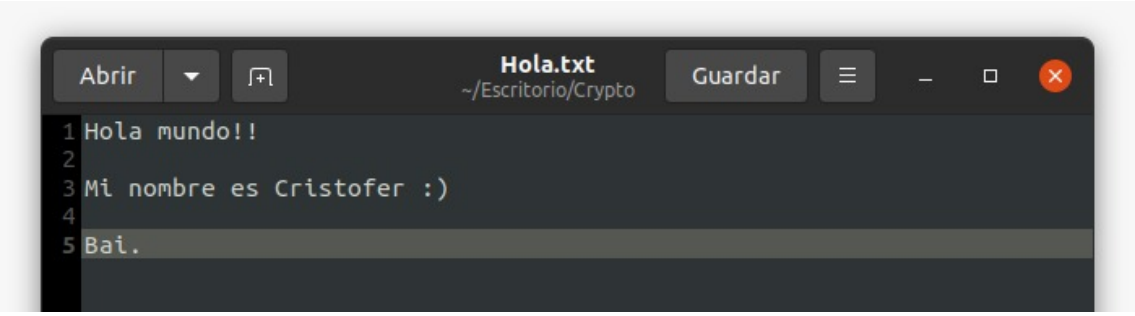
Con el segundo comando se usó el archivo PEM que contenía la clave privada de Cristofer para generar su clave pública en formato PEM.

4. Show the specific details of the parameters (prime number, values for a and b generator) associated with the elliptic curve you chose in point 2.

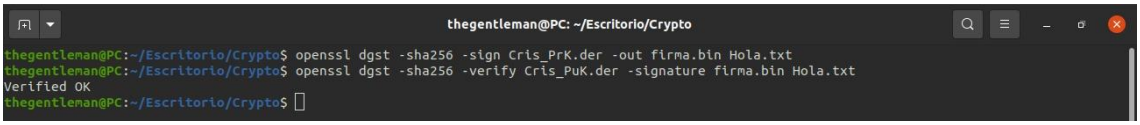


En la figura anterior se puede ver los detalles específicos de los parámetros de la curva P-521.

5. Sign and verify three files in different formats (.pdf, txt, docx, etc.) using the keys that you previouslygenerated.



Se firmó el archivo de texto Hola.txt.



El primer comando ejecuta la firma del archivo, mientras que el segundo comando hace la verificación del archivo, usando la firma obtenida del primer comando.

La salida “Verified OK” indica que el archivo es íntegro y auténtico.

Printing of the Cristofer’s pair of keys.

```
thegentleman@PC: ~/Escritorio/Crypto
thegentleman@PC:~/Escritorio/Crypto$ openssl ec -in Cris_PrK.pem -text -noout
read EC key
Private-Key: (521 bit)
priv:
    01:bf:b2:eb:59:f3:00:ee:8a:67:42:3e:f0:11:ff:
    f7:26:26:bd:b7:5b:5b:1c:d9:8f:f3:a7:82:aa:6b:
    33:01:26:bb:bc:f5:01:d8:b2:22:06:8c:5d:c3:61:
    f3:60:3b:12:51:94:6e:1e:d3:37:ce:fa:ca:ab:e5:
    4a:c6:08:f7:9f:aa
pub:
    04:01:6c:27:b4:e9:5b:73:b0:21:e1:ef:a1:08:46:
    61:02:49:ff:06:25:de:d0:64:97:ad:16:d5:54:c2:
    eb:35:0c:df:df:f1:93:30:74:68:94:19:fd:ec:00:
    6c:df:c1:38:08:d6:2d:5d:d6:64:2c:bc:28:84:eb:
    04:83:51:08:78:78:78:00:1c:28:91:35:05:7f:56:
    5a:a0:07:0e:81:c3:1e:1e:74:99:af:db:44:52:0a:
    e2:71:2c:95:f8:b3:93:71:fc:57:7d:e7:21:42:8f:
    11:b1:ca:b9:3c:c8:5e:9e:6f:ea:61:96:f0:a9:18:
    91:77:fc:8a:10:a5:8b:1a:d9:20:c0:c9:1e
ASN1 OID: secp521r1
NIST CURVE: P-521
thegentleman@PC:~/Escritorio/Crypto$
```

Córdova Pichardo Francisco Uziel.

1. List the elliptic curves available in OpenSSL. Identify if there are curves from the standard SP800-186 in Openssl.

```
uziel@uziel-X455LAB: ~/Documentos
  c2tnb359v1: X9.62 curve over a 359 bit binary field
  c2pnb368w1: X9.62 curve over a 368 bit binary field
  c2tnb431r1: X9.62 curve over a 431 bit binary field
  wap-wsg-idm-ecid-wtls1: WTLS curve over a 113 bit binary field
  wap-wsg-idm-ecid-wtls3: NIST/SECG/WTLS curve over a 163 bit binary field
  wap-wsg-idm-ecid-wtls4: SECG curve over a 113 bit binary field
  wap-wsg-idm-ecid-wtls5: X9.62 curve over a 163 bit binary field
  wap-wsg-idm-ecid-wtls6: SECG/WTLS curve over a 112 bit prime field
  wap-wsg-idm-ecid-wtls7: SECG/WTLS curve over a 160 bit prime field
  wap-wsg-idm-ecid-wtls8: WTLS curve over a 112 bit prime field
  wap-wsg-idm-ecid-wtls9: WTLS curve over a 160 bit prime field
  wap-wsg-idm-ecid-wtls10: NIST/SECG/WTLS curve over a 233 bit binary field
  wap-wsg-idm-ecid-wtls11: NIST/SECG/WTLS curve over a 233 bit binary field
  wap-wsg-idm-ecid-wtls12: WTLS curve over a 224 bit prime field
  Oakley-EC2N-3:
        IPSec/IKE/Oakley curve #3 over a 155 bit binary field.
        Not suitable for ECDSA.
        Questionable extension field!
  Oakley-EC2N-4:
        IPSec/IKE/Oakley curve #4 over a 185 bit binary field.
        Not suitable for ECDSA.
        Questionable extension field!
  brainpoolP160r1: RFC 5639 curve over a 160 bit prime field
  brainpoolP160t1: RFC 5639 curve over a 160 bit prime field
  brainpoolP192r1: RFC 5639 curve over a 192 bit prime field
  brainpoolP192t1: RFC 5639 curve over a 192 bit prime field
  brainpoolP224r1: RFC 5639 curve over a 224 bit prime field
  brainpoolP224t1: RFC 5639 curve over a 224 bit prime field
  brainpoolP256r1: RFC 5639 curve over a 256 bit prime field
  brainpoolP256t1: RFC 5639 curve over a 256 bit prime field
  brainpoolP320r1: RFC 5639 curve over a 320 bit prime field
  brainpoolP320t1: RFC 5639 curve over a 320 bit prime field
  brainpoolP384r1: RFC 5639 curve over a 384 bit prime field
  brainpoolP384t1: RFC 5639 curve over a 384 bit prime field
  brainpoolP512r1: RFC 5639 curve over a 512 bit prime field
  brainpoolP512t1: RFC 5639 curve over a 512 bit prime field
  SM2        : SM2 curve over a 256 bit prime field
uziel@uziel-X455LAB:~/Documentos$
```

Se identificaron las curvas P-224, P-384 y P-521, nombradas como secp[…] y etiquetadas como NIST/SECG curve […].

2. Choose three different elliptic curves over a prime field and generate the EC parameters in a .pem file. Also store the keys in .der files

```
uziel@uziel-X455LAB: ~/Documentos
uziel@uziel-X455LAB:~/Documentos$ openssl ecparam -name brainpoolP384t1 -out brainpoolP384t1.pem
uziel@uziel-X455LAB:~/Documentos$ openssl ecparam -in brainpoolP384t1.pem -genkey -noout -out Uziel_PrK.der
uziel@uziel-X455LAB:~/Documentos$ openssl ec -in Uziel_PrK.der -pubout -out Uziel_PuK.der
read EC key
writing EC key
uziel@uziel-X455LAB:~/Documentos$ ls
android-studio-2022.2.1.19-linux         CurvaEliptica.pdf
android-studio-2022.2.1.19-linux.tar.gz  jdk-20.0.1
brainpool384t1.pem                       Uziel_PrK.der
brainpoolP384t1.pem                      Uziel_PuK.der
uziel@uziel-X455LAB:~/Documentos$
```

Con el primer comando se guardan los parámetros de la curva P-384 en el archivo **.pem**

Con el segundo comando se usó el archivo **.pem** con la curva P-384 para crear la llave privada de Uziel en un archivo **.der**

Con el tercer comando se usó el archivo **.pem** que contiene la llave privada de Uziel para crear la llave pública de Uziel en un archivo **.der**

3. Choose an elliptic curve and generate a pair of keys (private and public) for each member in the team. Store each key in a different .pem file

```
uziel@uziel-X455LAB: ~/Documentos
uziel@uziel-X455LAB:~/Documentos$ openssl ecparam -in brainpoolP384t1.pem -genkey -noout -out Uziel_PrK.pem
uziel@uziel-X455LAB:~/Documentos$ openssl ec -in Uziel_PrK.pem -pubout -out Uziel_PuK.pem
read EC key
writing EC key
uziel@uziel-X455LAB:~/Documentos$ ls
android-studio-2022.2.1.19-linux         brainpool384t1.pem   CurvaEliptica.pdf  Uziel_PrK.der  Uziel_PuK.der
android-studio-2022.2.1.19-linux.tar.gz  brainpoolP384t1.pem  jdk-20.0.1         Uziel_PrK.pem  Uziel_PuK.pem
uziel@uziel-X455LAB:~/Documentos$
```

Con el primer comando se usó el archivo **.pem** con la curva P-384 para generar las claves pública y privada de Uziel en formato **.pem**

Con el segundo comando se usó el archivo **.pem** que contenía la clave privada de Uziel para generar su clave pública en formato **.pem**

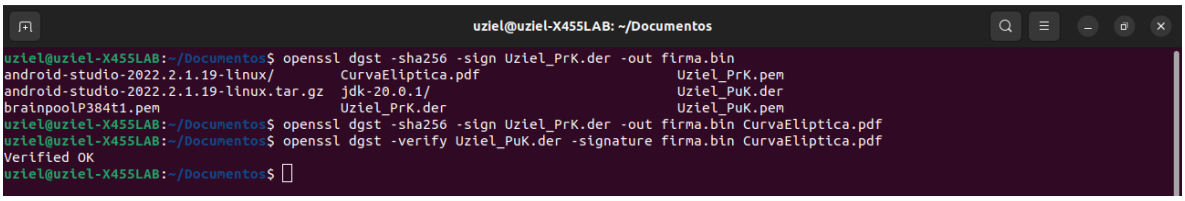
4. Show the specific details of the parameters (prime number, values for a and b generator) associated with the elliptic curve you chose in point 2.

```
uziel@uziel-X455LAB: ~/Documentos
uziel@uziel-X455LAB:~/Documentos$ openssl ecparam -in brainpoolP384t1.pem -text -param_enc explicit -noout
EC-Parameters: (384 bit)
Field Type: prime-field
Prime:
    00:8c:b9:1e:82:a3:38:6d:28:0f:5d:6f:7e:50:e6:
    41:df:15:2f:71:09:ed:54:56:b4:12:b1:da:19:7f:
    b7:11:23:ac:d3:a7:29:90:1d:1a:71:87:47:00:13:
    31:07:ec:53
A:
    00:8c:b9:1e:82:a3:38:6d:28:0f:5d:6f:7e:50:e6:
    41:df:15:2f:71:09:ed:54:56:b4:12:b1:da:19:7f:
    b7:11:23:ac:d3:a7:29:90:1d:1a:71:87:47:00:13:
    31:07:ec:50
B:
    7f:51:9e:ad:a7:bd:a8:1b:d8:26:db:a6:47:91:0f:
    8c:4b:93:46:ed:8c:cd:c6:4e:4b:1a:bd:11:75:6d:
    ce:1d:20:74:aa:26:3b:88:80:5c:ed:70:35:5a:33:
    b4:71:ee
Generator (uncompressed):
    04:18:de:98:b0:2d:b9:a3:06:f2:af:cd:72:35:f7:
    2a:81:9b:80:ab:12:eb:d6:53:17:24:76:fe:cd:46:
    2a:ab:ff:c4:ff:19:1b:94:6a:5f:54:d8:d0:aa:2f:
    41:88:08:cc:25:ab:05:69:62:d3:06:51:a1:14:af:
    d2:75:5a:d3:36:74:7f:93:47:5b:7a:1f:ca:3b:88:
    f2:b6:a2:08:cc:fe:46:94:08:58:4d:c2:b2:91:26:
    75:bf:5b:9e:58:29:28
Order:
    00:8c:b9:1e:82:a3:38:6d:28:0f:5d:6f:7e:50:e6:
    41:df:15:2f:71:09:ed:54:56:b3:1f:16:6e:6c:ac:
    04:25:a7:cf:3a:b6:af:6b:7f:c3:10:3b:88:32:02:
    e9:04:65:65
Cofactor:  1 (0x1)
uziel@uziel-X455LAB:~/Documentos$
```

En la imagen anterior se puede ver los detalles específicos de los parámetros de la curva P-384.

5. Sign and verify three files in different formats (.pdf, txt, docx, etc.) using the keys that you previouslygenerated.

Se firmó y verificó el archivo CurvaEliptica.pdf



El primer comando ejecuta la firma del archivo, mientras que el segundo comando hace la verificación del archivo, usando la firma obtenida del primer comando.

La salida “Verified OK” indica que el archivo es íntegro y auténtico.

Garcia Jimenez Juan Carlos.

1. List the elliptic curves available in OpenSSL. Identify if there are curves from the standard SP800-186 in Openssl.

Las curvas recomendadas por la SP800-186 se muestran en la siguiente tabla y se encuentran marcadas en color rojo en la lista de curvas soportadas por OpenSSL.

Specified Curves	Allowed Usage
K-233, B-233 K-283, B-283 K-409, B-409 K-571, B-571	Deprecated
P-224 P-256 P-384 P-521	ECDSA, EC key establishment (see [SP_800-56A])
Edwards25519 Edwards448	EdDSA
Curve25519, W-25519 Curve448, E448, W-448	Alternative representations included for implementation flexibility. Not to be used for ECDSA or EdDSA directly.

A continuación, se muestra el comando implementado en la terminal de Linux para obtener la lista de curvas soportadas, la lista de curvas soportadas y las curvas descritas por el SP800-186 marcadas en rojo:

Nota_1: Para P-256, secp256r1 no figura en la lista.

Nota_2: Algunas de las curvas pueden ser soportadas por OpenSSL, pero de manera genérica y no aparecen en la lista presentada.

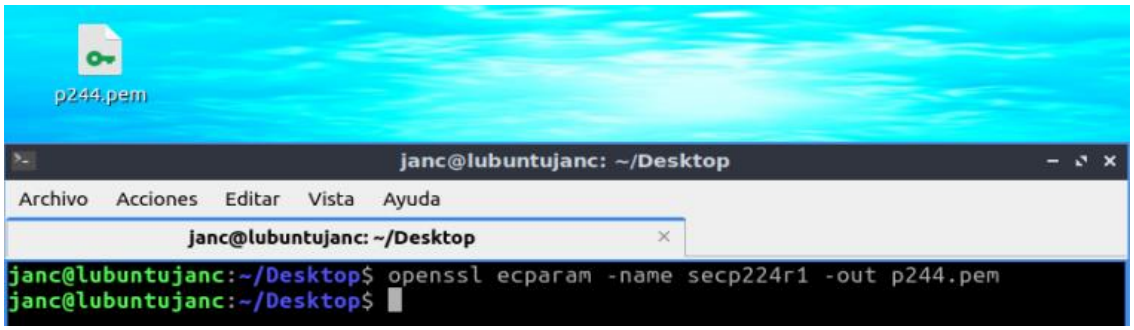
```
janc@lubuntujanc:~$ openssl ecparam -list_curves
  secp112r1 : SECG/WTLS curve over a 112 bit prime field
  secp112r2 : SECG curve over a 112 bit prime field
  secp128r1 : SECG curve over a 128 bit prime field
  secp128r2 : SECG curve over a 128 bit prime field
  secp160k1 : SECG curve over a 160 bit prime field
  secp160r1 : SECG curve over a 160 bit prime field
  secp160r2 : SECG/WTLS curve over a 160 bit prime field
  secp192k1 : SECG curve over a 192 bit prime field
```

secp224k1 : SECG curve over a 224 bit prime field
secp224r1 : NIST/SECG curve over a 224 bit prime field
secp256k1 : SECG curve over a 256 bit prime field
secp384r1 : NIST/SECG curve over a 384 bit prime field
secp521r1 : NIST/SECG curve over a 521 bit prime field
prime192v1: NIST/X9.62/SECG curve over a 192 bit prime field
prime192v2: X9.62 curve over a 192 bit prime field
prime192v3: X9.62 curve over a 192 bit prime field
prime239v1: X9.62 curve over a 239 bit prime field
prime239v2: X9.62 curve over a 239 bit prime field
prime239v3: X9.62 curve over a 239 bit prime field
prime256v1: X9.62/SECG curve over a 256 bit prime field
sect113r1 : SECG curve over a 113 bit binary field
sect113r2 : SECG curve over a 113 bit binary field
sect131r1 : SECG/WTLS curve over a 131 bit binary field
sect131r2 : SECG curve over a 131 bit binary field
sect163k1 : NIST/SECG/WTLS curve over a 163 bit binary field
sect163r1 : SECG curve over a 163 bit binary field
sect163r2 : NIST/SECG curve over a 163 bit binary field
sect193r1 : SECG curve over a 193 bit binary field
sect193r2 : SECG curve over a 193 bit binary field
sect233k1 : NIST/SECG/WTLS curve over a 233 bit binary field
sect233r1 : NIST/SECG/WTLS curve over a 233 bit binary field
sect239k1 : SECG curve over a 239 bit binary field
sect283k1 : NIST/SECG curve over a 283 bit binary field
sect283r1 : NIST/SECG curve over a 283 bit binary field
sect409k1 : NIST/SECG curve over a 409 bit binary field
sect409r1 : NIST/SECG curve over a 409 bit binary field
sect571k1 : NIST/SECG curve over a 571 bit binary field
sect571r1 : NIST/SECG curve over a 571 bit binary field
c2pnb163v1: X9.62 curve over a 163 bit binary field
c2pnb163v2: X9.62 curve over a 163 bit binary field
c2pnb163v3: X9.62 curve over a 163 bit binary field
c2pnb176v1: X9.62 curve over a 176 bit binary field
c2tnb191v1: X9.62 curve over a 191 bit binary field
c2tnb191v2: X9.62 curve over a 191 bit binary field
c2tnb191v3: X9.62 curve over a 191 bit binary field
c2pnb208w1: X9.62 curve over a 208 bit binary field
c2tnb239v1: X9.62 curve over a 239 bit binary field
c2tnb239v2: X9.62 curve over a 239 bit binary field
c2tnb239v3: X9.62 curve over a 239 bit binary field
c2pnb272w1: X9.62 curve over a 272 bit binary field
c2pnb304w1: X9.62 curve over a 304 bit binary field
c2tnb359v1: X9.62 curve over a 359 bit binary field
c2pnb368w1: X9.62 curve over a 368 bit binary field
c2tnb431r1: X9.62 curve over a 431 bit binary field
wap-wsg-idm-ecid-wtls1: WTLS curve over a 113 bit binary field

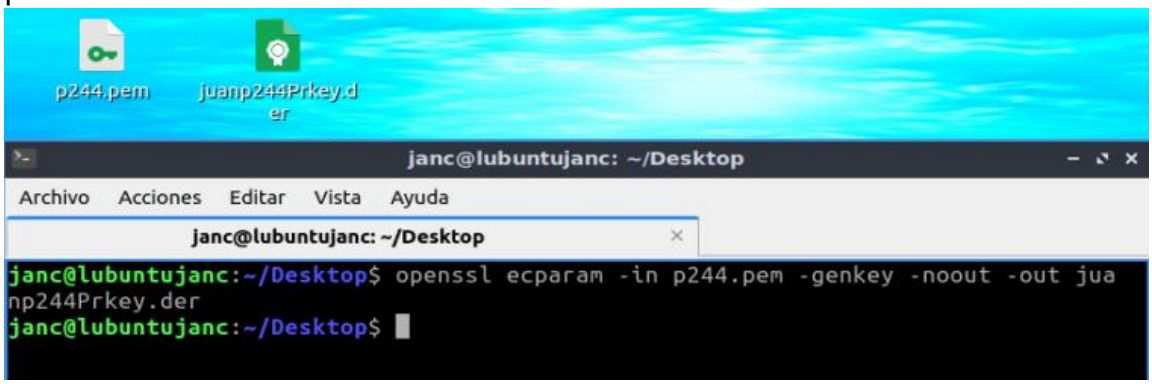
```
wap-wsg-idm-ecid-wtls3: NIST/SECG/WTLS curve over a 163 bit binary field
wap-wsg-idm-ecid-wtls4: SECG curve over a 113 bit binary field
wap-wsg-idm-ecid-wtls5: X9.62 curve over a 163 bit binary field
wap-wsg-idm-ecid-wtls6: SECG/WTLS curve over a 112 bit prime field
wap-wsg-idm-ecid-wtls7: SECG/WTLS curve over a 160 bit prime field
wap-wsg-idm-ecid-wtls8: WTLS curve over a 112 bit prime field
wap-wsg-idm-ecid-wtls9: WTLS curve over a 160 bit prime field
wap-wsg-idm-ecid-wtls10: NIST/SECG/WTLS curve over a 233 bit binary field
wap-wsg-idm-ecid-wtls11: NIST/SECG/WTLS curve over a 233 bit binary field
wap-wsg-idm-ecid-wtls12: WTLS curve over a 224 bit prime field
Oakley-EC2N-3:
    IPSec/IKE/Oakley curve #3 over a 155 bit binary field.
    Not suitable for ECDSA.
    Questionable extension field!
Oakley-EC2N-4:
    IPSec/IKE/Oakley curve #4 over a 185 bit binary field.
    Not suitable for ECDSA.
    Questionable extension field!
brainpoolP160r1: RFC 5639 curve over a 160 bit prime field
brainpoolP160t1: RFC 5639 curve over a 160 bit prime field
brainpoolP192r1: RFC 5639 curve over a 192 bit prime field
brainpoolP192t1: RFC 5639 curve over a 192 bit prime field
brainpoolP224r1: RFC 5639 curve over a 224 bit prime field
brainpoolP224t1: RFC 5639 curve over a 224 bit prime field
brainpoolP256r1: RFC 5639 curve over a 256 bit prime field
brainpoolP256t1: RFC 5639 curve over a 256 bit prime field
brainpoolP320r1: RFC 5639 curve over a 320 bit prime field
brainpoolP320t1: RFC 5639 curve over a 320 bit prime field
brainpoolP384r1: RFC 5639 curve over a 384 bit prime field
brainpoolP384t1: RFC 5639 curve over a 384 bit prime field
brainpoolP512r1: RFC 5639 curve over a 512 bit prime field
brainpoolP512t1: RFC 5639 curve over a 512 bit prime field
SM2       : SM2 curve over a 256 bit prime field
```

2. Choose three different elliptic curves over a prime field and generate the EC parameters in a .pem file. Also store the keys in .der files

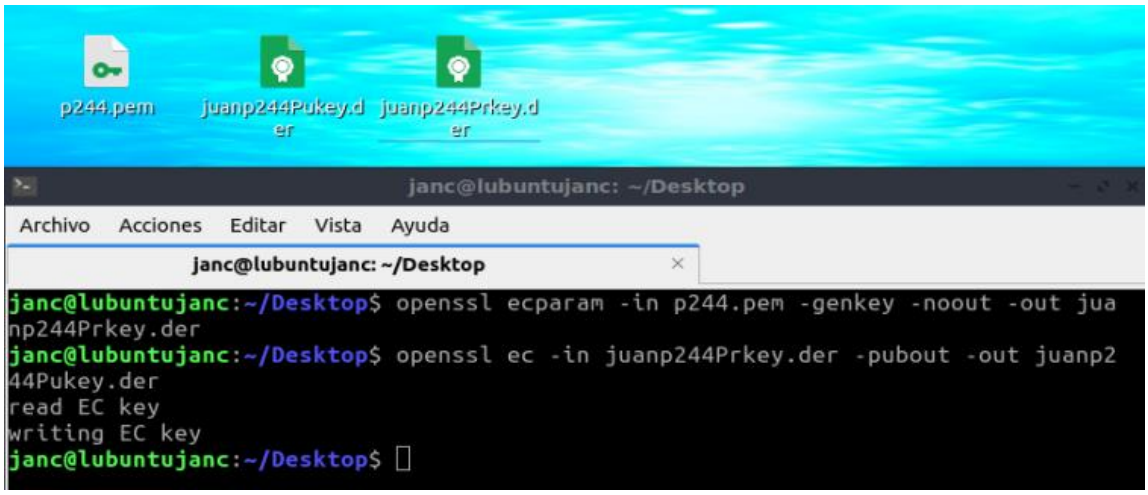
Como se muestra en la imagen, se genera y guardan los parámetros de la curva P-224 en forma de archive “pem”:



Posteriormente, se implementa el archivo “pem” de la curva para obtener la llave privada en un archivo “der”:

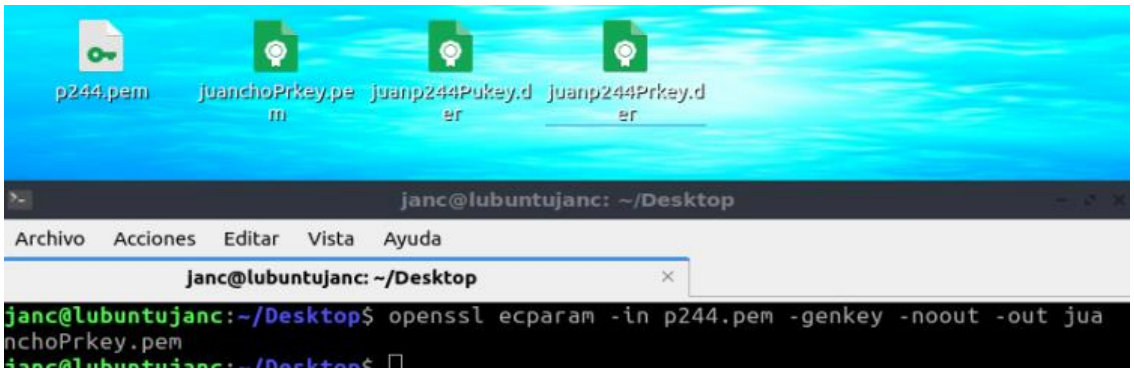


Finalmente, se implementa el archivo “der” (llave privada) para obtener la llave publica en formato “der”:

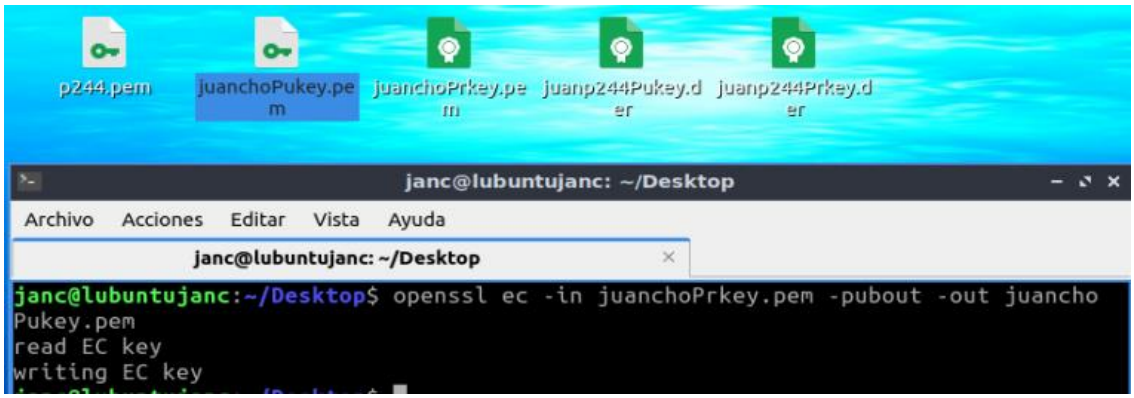


3. Choose an elliptic curve and generate a pair of keys (private and public) for each member in the team. Store each key in a different .pem file

Con base en la curva P-244, se hace uso del archivo contenedor de la curva (pem) para generar la llave privada con formato “pem”:

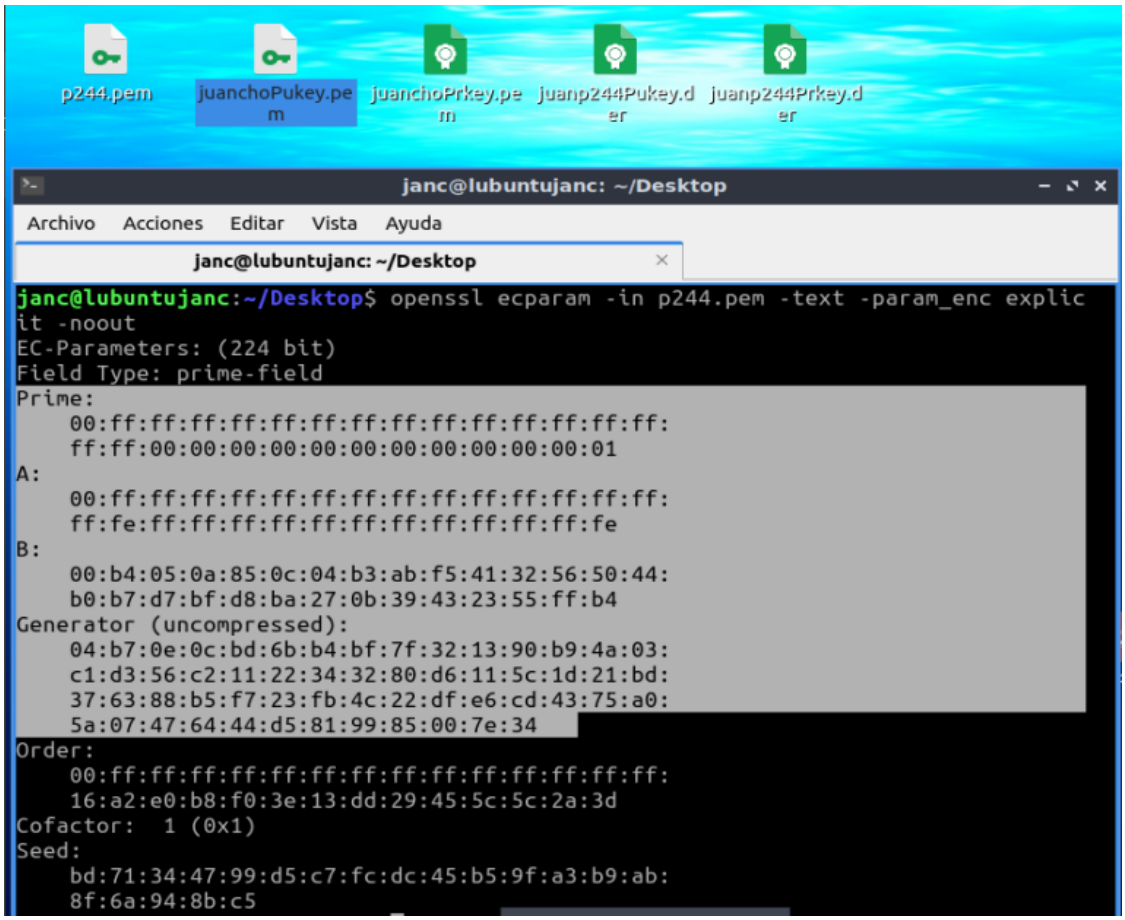


Posteriormente, se genera la llave publica “pem” con la llave publica generada en la imagen anterior:



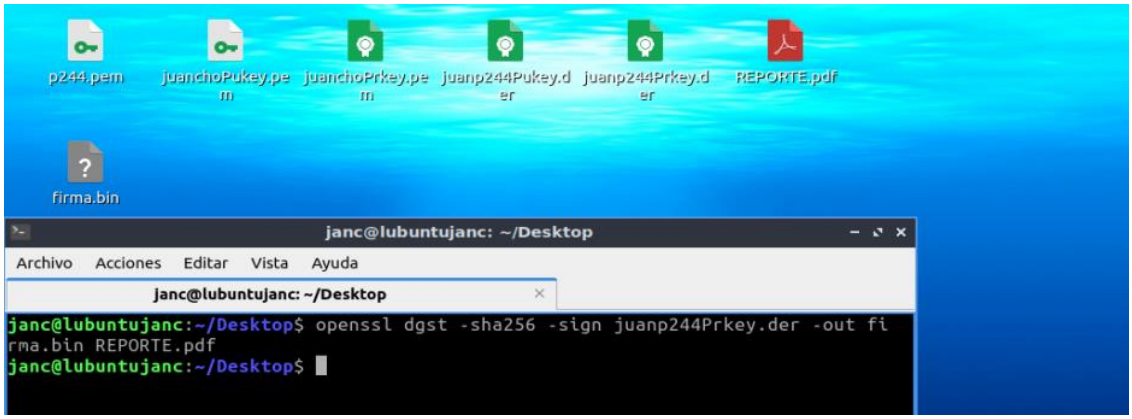
4. Show the specific details of the parameters (prime number, values for a and b generator) associated with the elliptic curve you chose in point 2.

Para dicho efecto, se aplica el siguiente comando sobre el archivo “pem” contenedor de la curva:

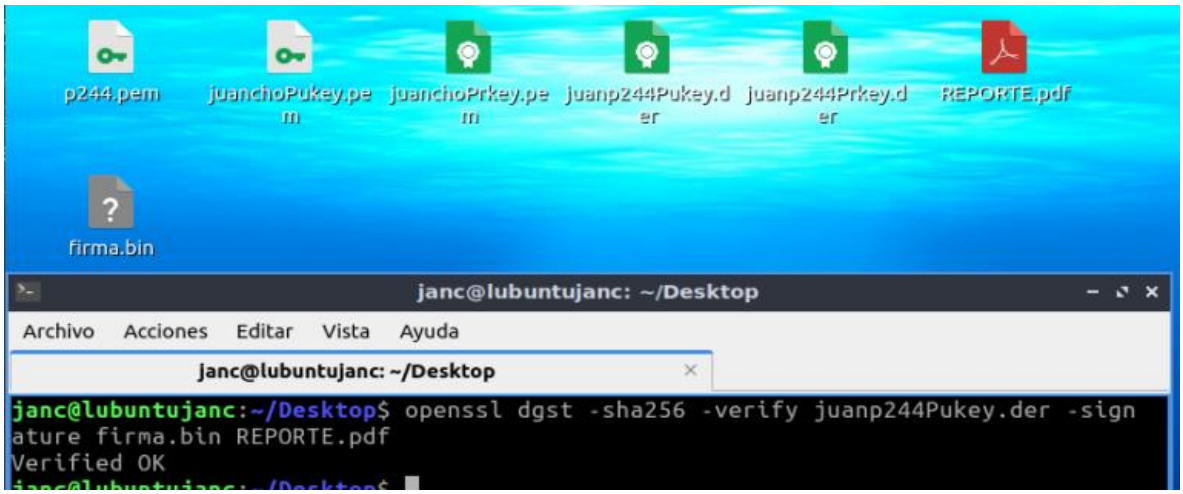


5. Sign and verify three files in different formats (.pdf, txt, docx, etc.) using the keys that you previously generated.

Se tiene preparado con antelación un archivo PDF a firmar; para hacerlo, se implementa el siguiente comando, tal y como se muestra en la siguiente imagen:



Finalmente, para verificar la integridad del archivo:


p244.pem
juanchoPukey.pem
juanchoPrkey.pem
juanp244Pukey.der
juanp244Prkey.der
REPORTE.pdf
firma.bin
janc@lubuntujanc: ~/Desktop
Archivo Acciones Editar Vista Ayuda
janc@lubuntujanc: ~/Desktop
janc@lubuntujanc:~/Desktop$ openssl dgst -sha256 -verify juanp244Pukey.der -signature firma.bin REPORTE.pdf
Verified OK