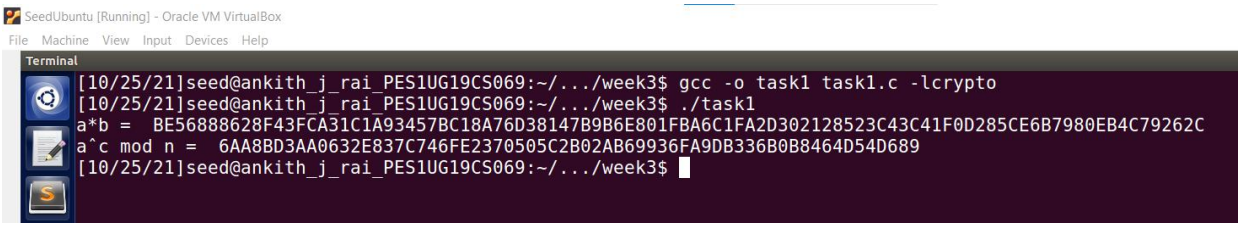
CRYPTOGRAPHY LAB- WEEK 3
RSA Public-Key Encryption and Signature Lab

NAME : Ankith J Rai
SRN : PES1UG19CS069
SEC : B

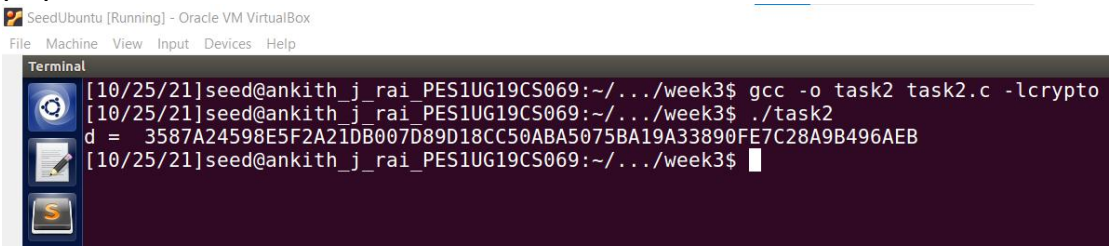
Task 1: A Complete Example of BIGNUM



We can see that on running task1.c we get the value of a*b and the value of a^c mod n.

Task 2: Deriving the private key

In this task we will be deriving private key using the hexadecimal values p,q and e.



From the above screenshot we can see that the private key has been generated.

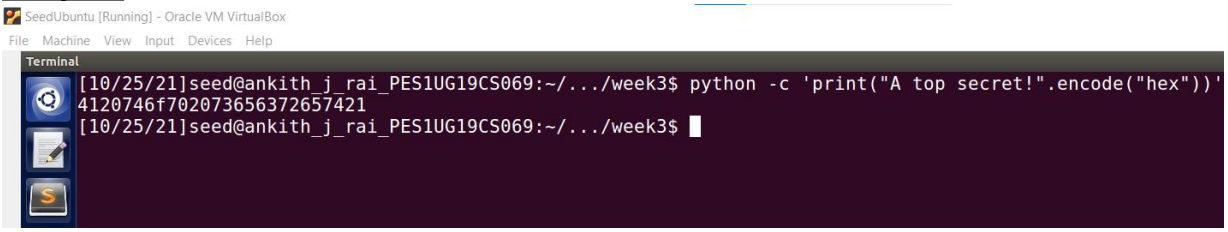
Q1. Explain your understanding (in terms of mathematical statements) of what the above code does.

Ans)

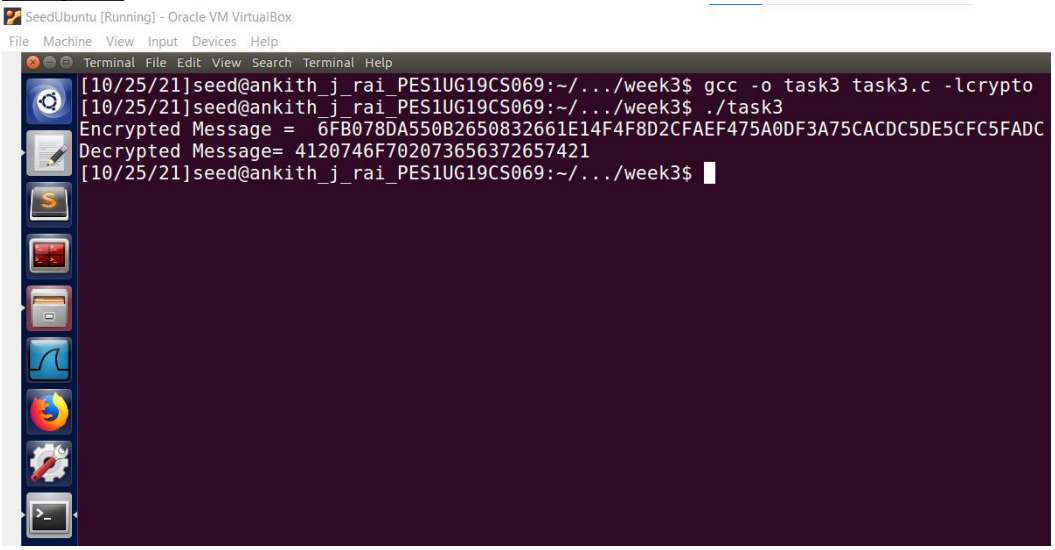
Task 3: Encrypting a message

In this task we will be encrypting “A top secret” using hexadecimal values n and e.

Step 1:

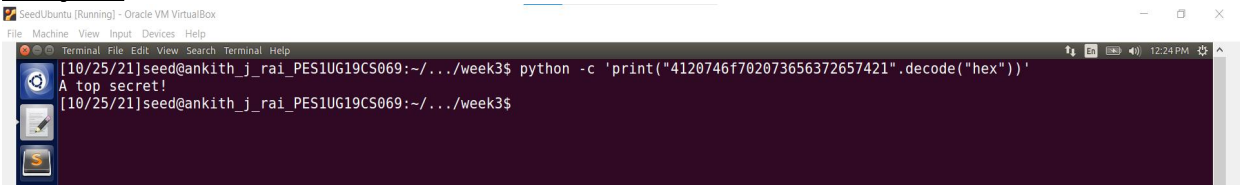


Step 2:



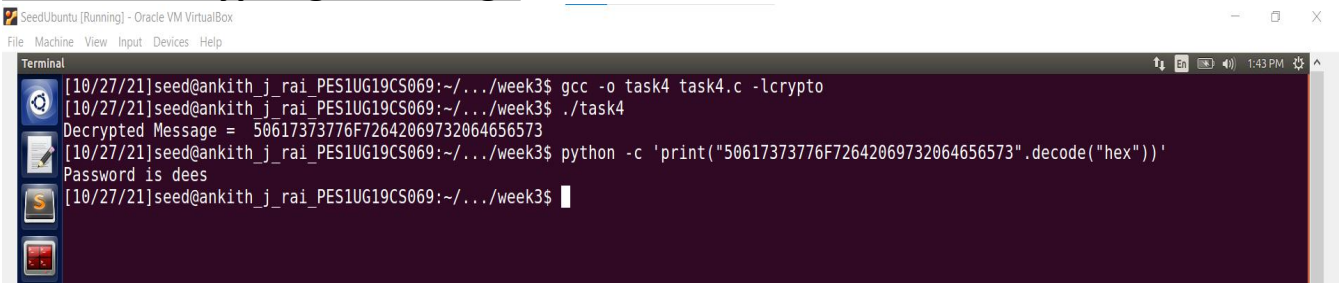
From the above screenshot we can see that we have got a encrypted and decrypted message.

Step 3:



We decode the obtained decrypted message to get the original message.

Task 4: Decrypting a message

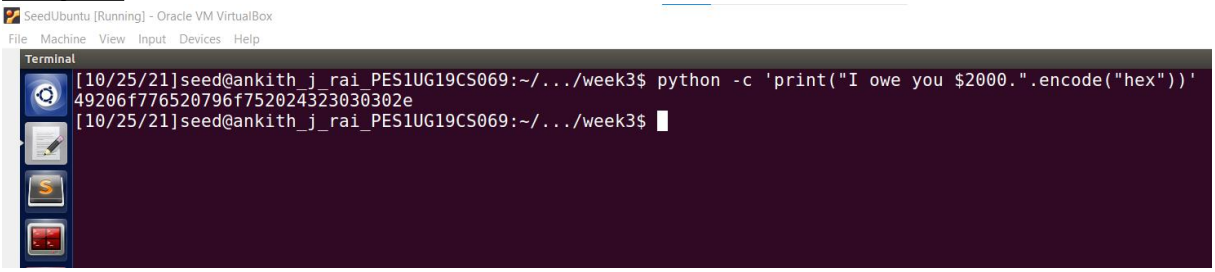


We can see that on converting the decrypted message from hex to ascii the decrypted message we get is “Password id dees”.

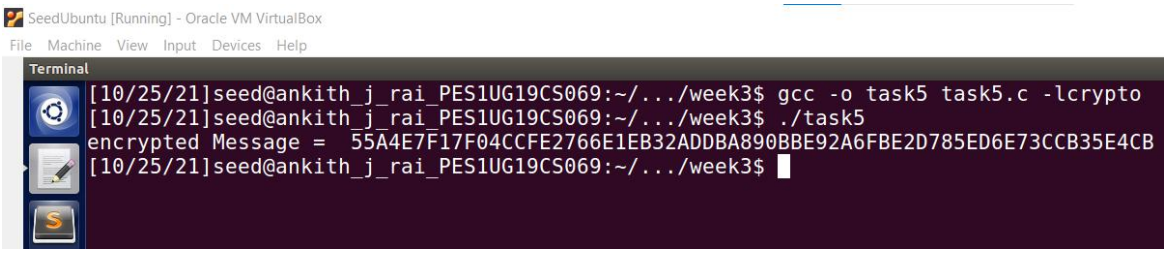
Task 5: Signing a Message

In this task we will be generating a signature for the message “I owe you $2000”

Step 1:



Step 2:

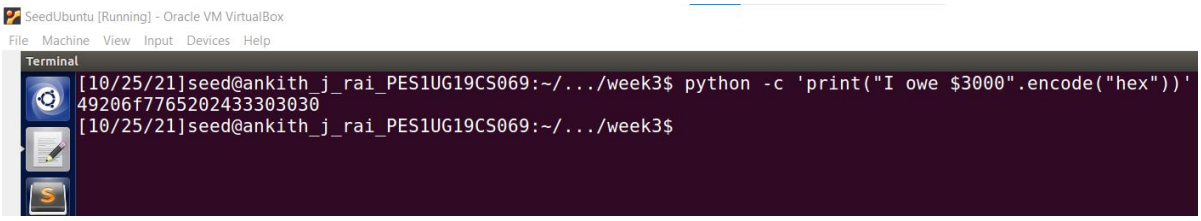


From the above screenshots we can see that we have generated the signature for the given message using M^d mod n algorithm.

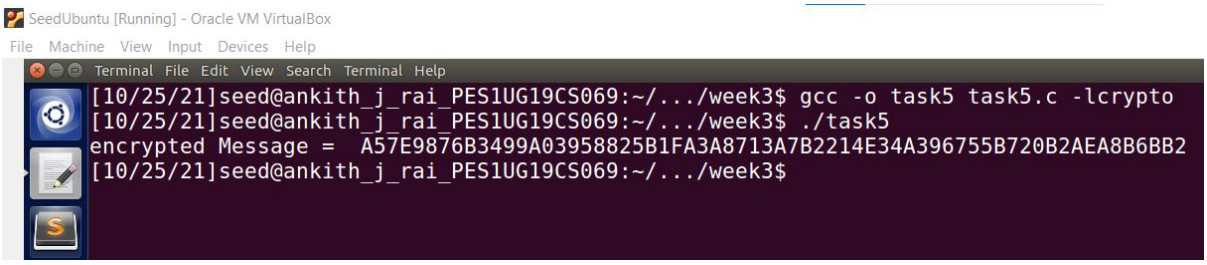
Step 3:

Now for message ‘I owe $3000’

STEP 1:

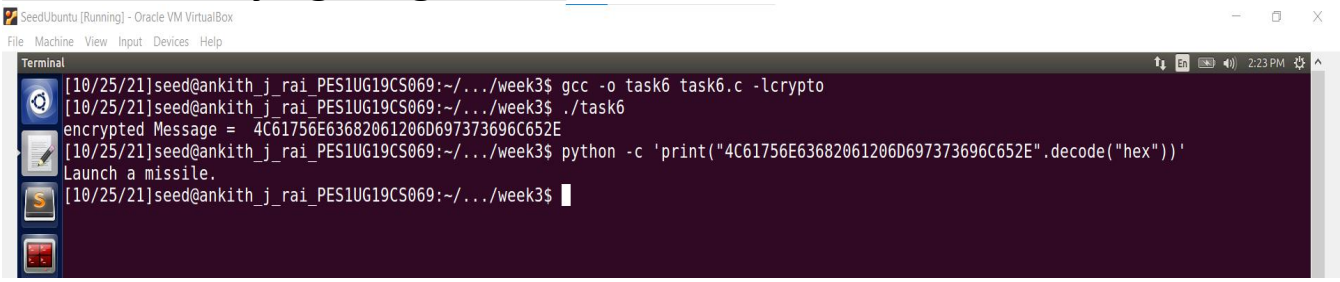


STEP 2:



Now we can see from the above screenshot that we have got the signature generated for the message “I owe $3000”

Task 6: Verifying a Signature

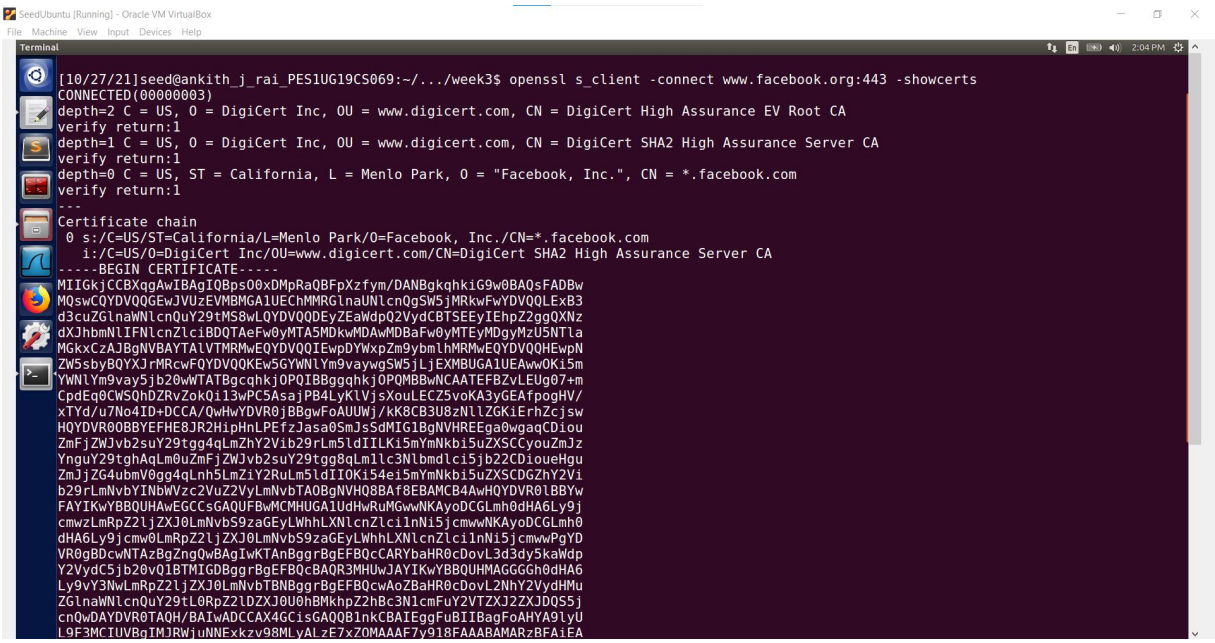


From the above screenshot we have verified that the signature received by bob is from allice.

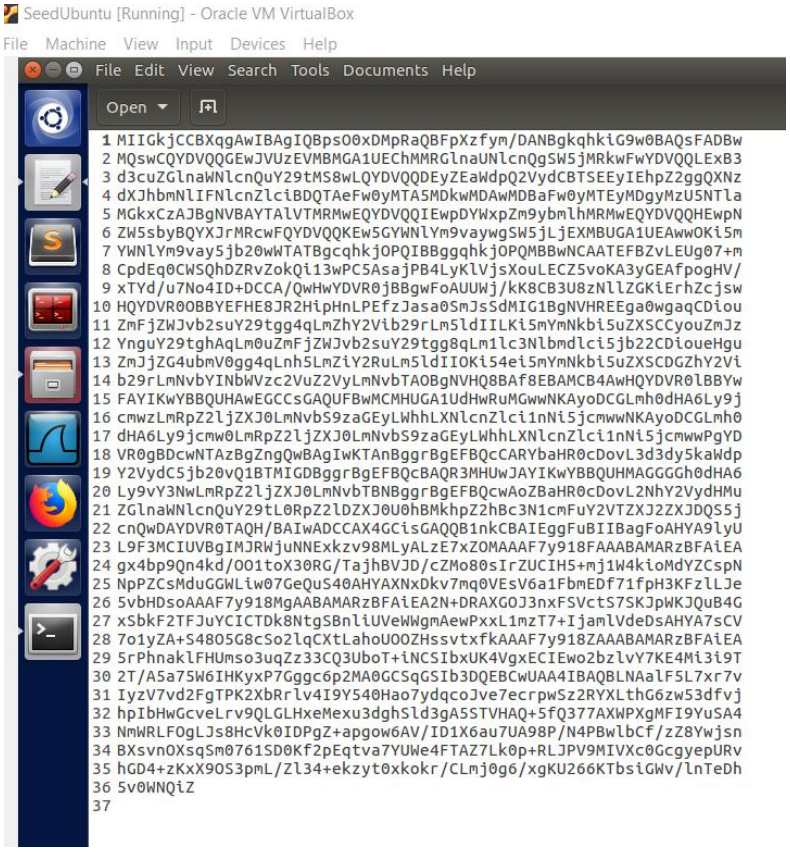
Task 7: Manually verifying an X.509 Certificate

Step 1:

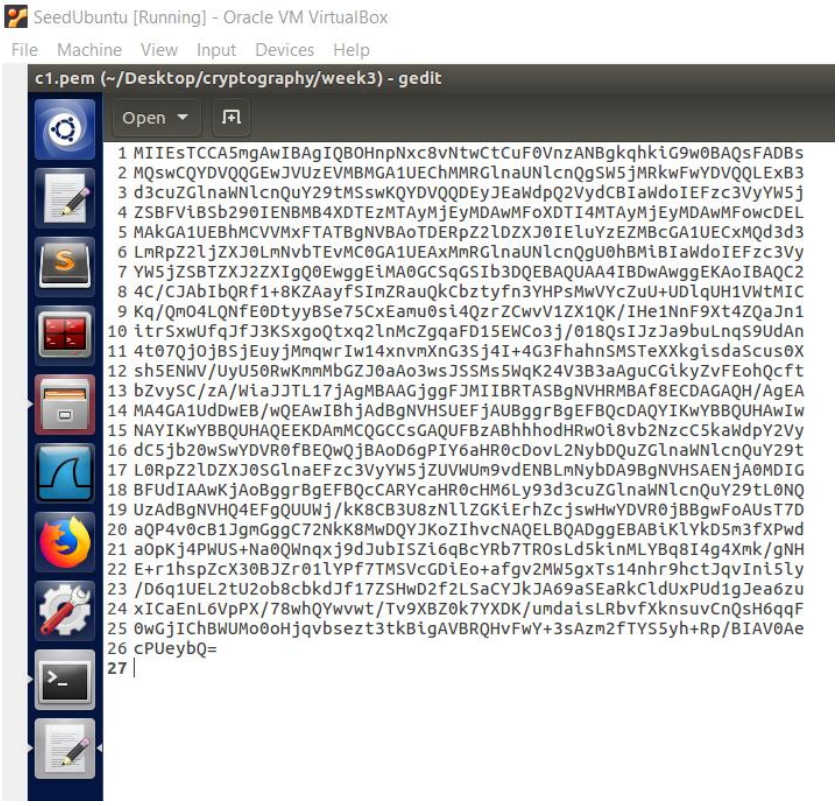
Screeshot of terminal:



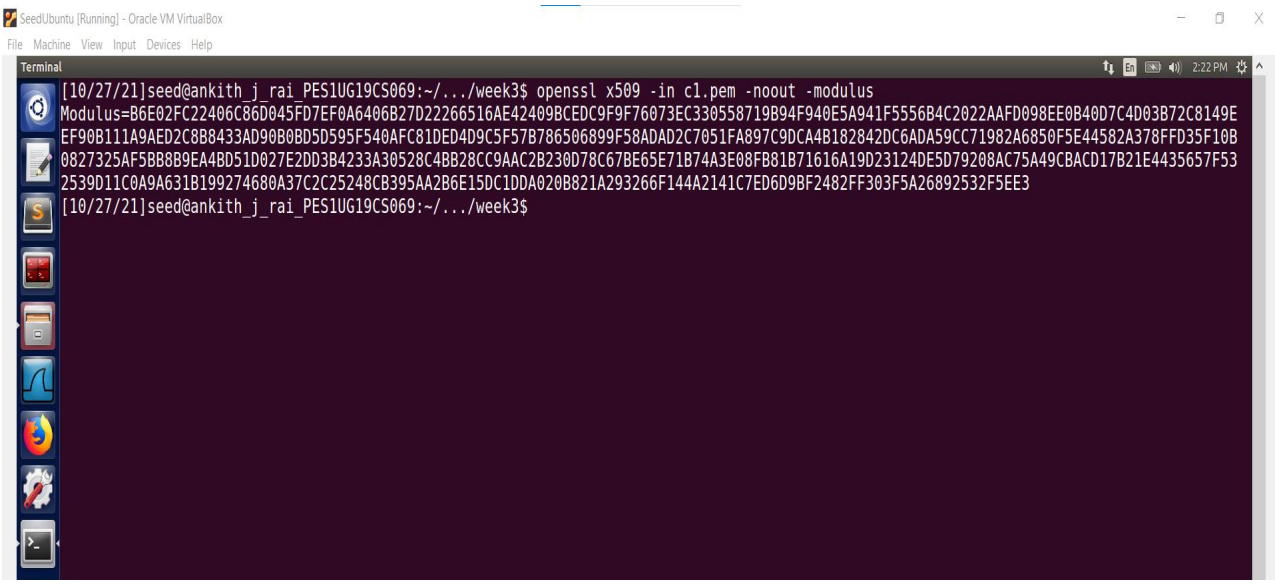
Screenshot of c0.pem:



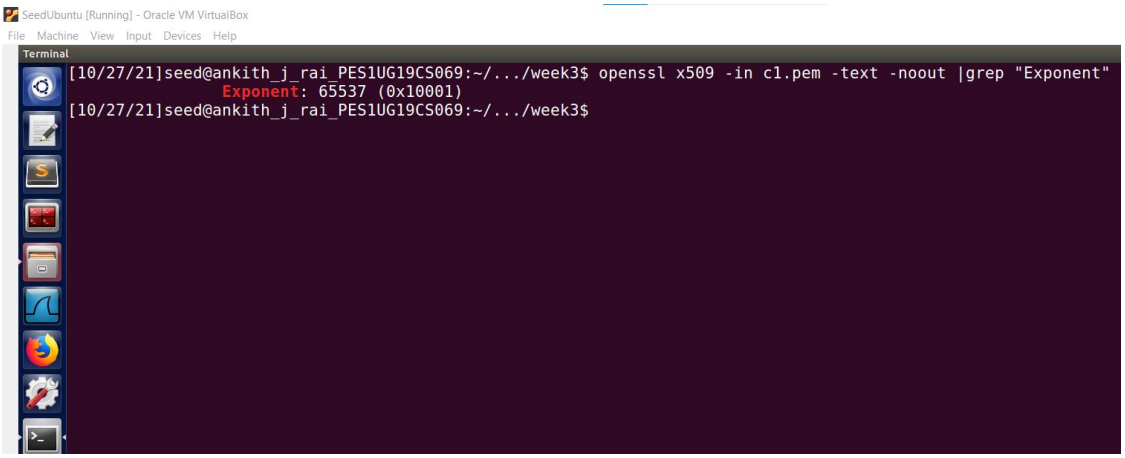
Screenshot of c1.pem:



Step 2:



We get the **n value** as
B6E02FC22406C86D045FD7EF0A6406B27D22266516AE42409BCEDC9F9F76073EC33
0558719B94F940E5A941F5556B4C2022AAFD098EE0B40D7C4D03B72C8149EEF90B1
11A9AED2C8B8433AD90B0BD5D595F540AFC81DED4D9C5F57B786506899F58ADAD
2C7051FA897C9DCA4B182842DC6ADA59CC71982A6850F5E44582A378FFD35F10B0
827325AF5BB8B9EA4BD51D027E2DD3B4233A30528C4BB28CC9AAC2B230D78C67B
E65E71B74A3E08FB81B71616A19D23124DE5D79208AC75A49CBACD17B21E443565
7F532539D11C0A9A631B199274680A37C2C25248CB395AA2B6E15DC1DDA020B821
A293266F144A2141C7ED6D9BF2482FF303F5A26892532F5EE3



We get the e value as **65537(0x10001)**.

Step 3:

Screeshot of terminal:

```
[10/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week3$ openssl x509 -in c0.pem -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            06:9b:0e:d3:10:cc:a5:16:90:04:5a:57:cd:fc:a6:fc
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
        Validity
            Not Before: Sep  9 00:00:00 2021 GMT
            Not After : Dec  8 23:59:59 2021 GMT
        Subject: C=US, ST=California, L=Menlo Park, O=Facebook, Inc., CN=*.facebook.com
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:c4:14:16:6f:2c:45:20:d3:bf:a6:0a:97:44:ab:
                    40:96:49:08:43:65:1b:d9:a2:44:22:d7:7c:0f:0b:
                    90:2c:6a:33:c1:e0:bc:8a:95:58:ec:5e:8b:8b:10:
                    26:79:be:82:80:df:21:84:01:fa:68:80:75:7f:c5:
                    36:1d:fe:ee:cd
                ASN1 OID: prime256v1
                NIST CURVE: P-256
        X509v3 extensions:
            X509v3 Authority Key Identifier:
                keyid:51:68:FF:90:AF:02:07:75:3C:CC:D9:65:64:62:A2:12:B8:59:72:3B

            X509v3 Subject Key Identifier:
                71:3C:25:1D:87:8A:91:E7:2C:F1:1F:CC:96:AC:6B:44:A6:26:C4:9D
            X509v3 Subject Alternative Name:
                DNS:*.facebook.com, DNS:*.facebook.net, DNS:*.fbcdn.net, DNS:*.fbsbx.com, DNS:*.m.facebook.com, DNS:*.messenger.com, DNS
:*.xx.fbcdn.net, DNS:*.xy.fbcdn.net, DNS:*.xz.fbcdn.net, DNS:facebook.com, DNS:messenger.com
```

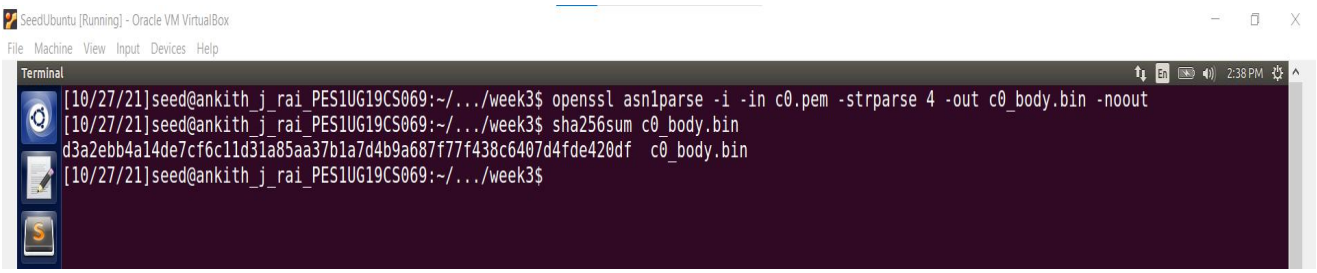
Screeshot of signature file:

```
Signature : ecdsa-with-SHA256
                                30:45:02:21:00:83:1E:1B:A7:D4:27:E2:47:7F:38:ED:
                                6D:A1:7D:F4:44:6F:D3:6A:38:41:54:90:FF:71:93:28:
                                F3:4B:08:AD:95:02:20:7E:7E:9A:3D:56:E2:48:A8:31:
                                D6:19:0A:CA:4D:36:93:D9:0A:C3:1D:B8:61:96:2E:2C:
                                34:EC:67:90:B9:2E:34
Signature : ecdsa-with-SHA256
                                30:45:02:21:00:D8:DF:83:44:05:C6:38:9D:E7:C4:54:
                                95:72:D4:BB:48:A2:69:58:A2:50:B8:1E:06:C5:26:E4:
                                17:64:C5:26:E6:02:20:24:C3:93:C3:6D:81:20:67:96:
                                25:15:79:65:A0:98:07:B0:3F:1C:4B:D6:6C:D3:EF:E2:
                                23:6A:69:55:75:E0:EC
Signature : ecdsa-with-SHA256
                                30:45:02:21:00:E6:B3:E1:9D:A9:25:14:75:26:B2:8D:
                                EE:A9:9C:F7:DC:24:37:51:BA:13:FA:23:42:48:86:F1:
                                50:AE:15:83:11:02:20:4C:28:D9:BC:E5:BD:8E:CA:13:
                                83:22:DE:2F:53:D9:3F:C0:E5:AE:F9:5B:A2:07:2B:2C:
                                4F:EC:68:20:73:AA:76
```

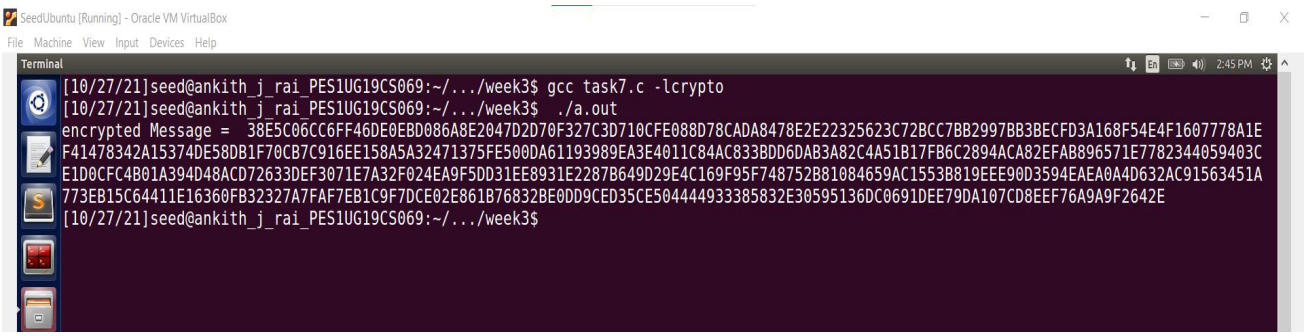
```
[10/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week3$ cat signature | tr -d '[:space:]:'
Signatureecdsa-with-SHA2563045022100831E1BA7D427E2477F38ED6DA17DF4446FD36A38415490FF719328F34B08AD9502207E7E9A3D56E248A831D6190ACA4D3693
D90AC31DB861962E2C34EC6790B92E34Signatureecdsa-with-SHA2563045022100D8DF834405C6389DE7C4549572D4BB48A26958A250B81E06C526E41764C526E60220
24C393C36D8120679625157965A09807B03F1C4BD66CD3EFE2236A695575E0ECSignatureecdsa-with-SHA2563045022100E6B3E19DA925147526B28DEEA99CF7DC2437
51BA13FA23424886F150AE15831102204C28D9BCE5BD8ECA138322DE2F53D93FC0E5AEF95BA2072B2C4FEC682073AA76[10/27/21]seed@ankith_j_rai_PES1UG19CS06
9:~/.../week3$
```

2563045022100831E1BA7D427E2477F38ED6DA17DF4446FD36A38415490FF719328
F34B08AD9502207E7E9A3D56E248A831D6190ACA4D3693D90AC31DB861962E2C34
EC6790B92E34SHA2563045022100D8DF834405C6389DE7C4549572D4BB48A26958
A250B81E06C526E41764C526E6022024C393C36D8120679625157965A09807B03F1
C4BD66CD3EFE2236A695575E0EC2563045022100E6B3E19DA925147526B28DEEA99
CF7DC243751BA13FA23424886F150AE15831102204C28D9BCE5BD8ECA138322DE2
F53D93FC0E5AEF95BA2072B2C4FEC682073AA76

Step 4:



Step 5:



From the above screenshot we can see that we have verified the signature.