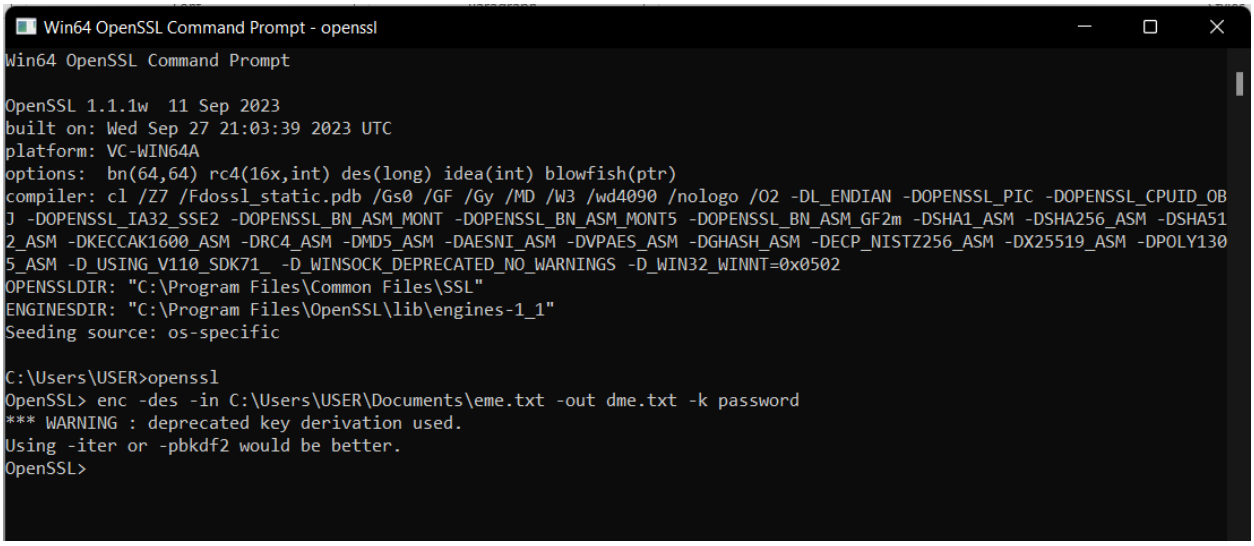
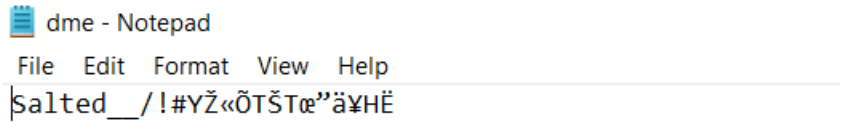
Name: OGAH DESTINY ODALEKO
Matirc: bhu/21/04/09/0073
Dept: cyber security

1d. The ENC command is used to encrypt and decrypt files.

2a.

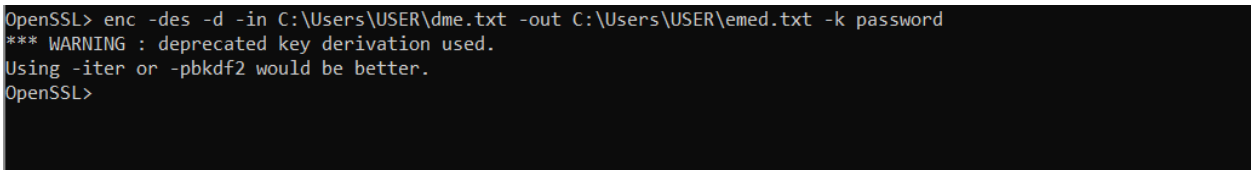


- Command for Encrypting a text in “eme.txt” to “dme.txt” with the password “password”



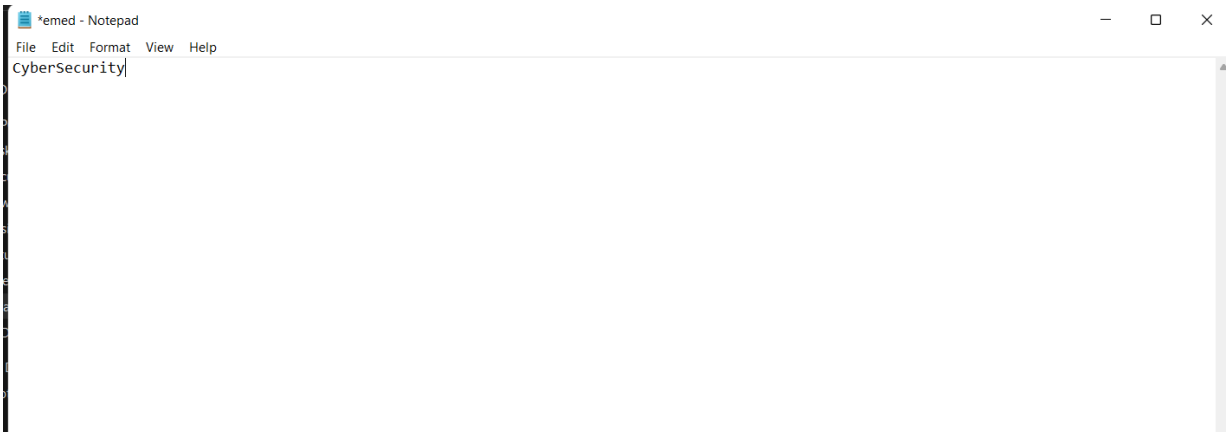
- The “eme.txt” plaintext and “dme.txt” ciphertext .

b.



- Command for decrypting “dme.txt” to “emed.txt”

COMPARING BOTH “EME.TXT” AND “EMED.TXT”

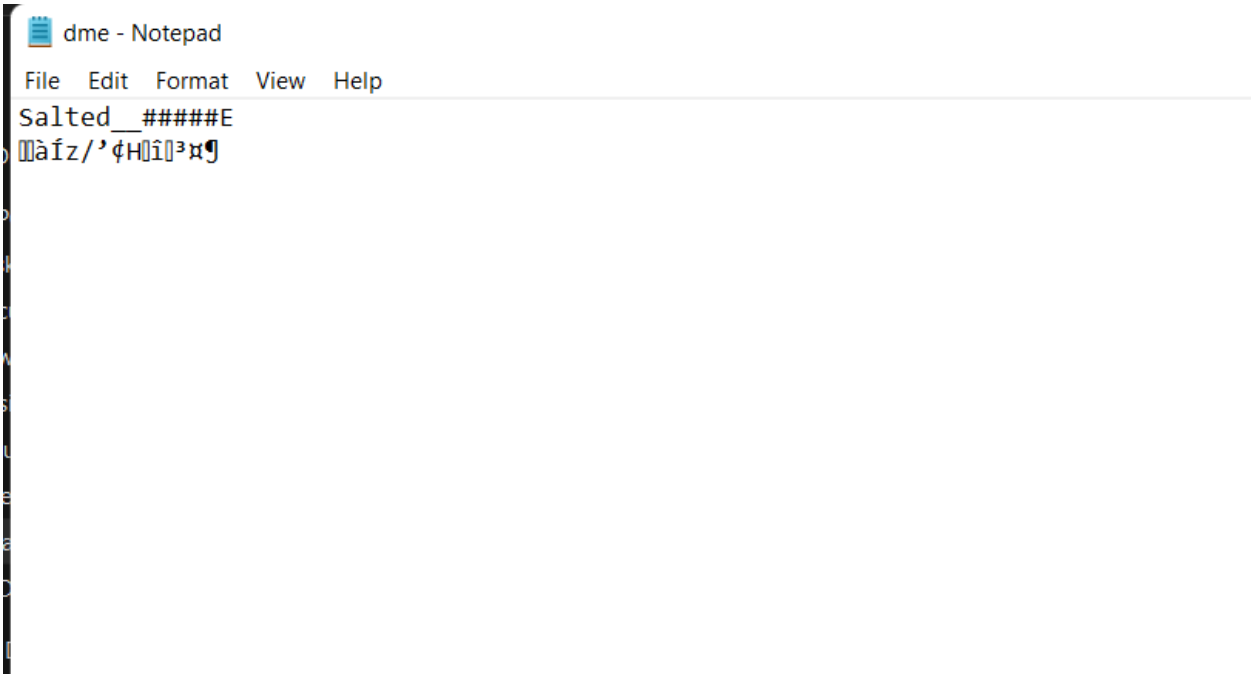


After decrypting the “dme.txt” to “eme.txt”, it was discovered that the plaintext in the “eme.txt” is the same as “emed.txt”

c.

```
OpenSSL> enc -des -in C:\Users\USER\Documents\eme.txt -out dme.txt -S 232323232345
enter des-cbc encryption password:
Verifying - enter des-cbc encryption password:
hex string is too short, padding with zero bytes to length
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
OpenSSL>
```

- This is the command to encrypt “eme.txt” with the password “password” using the salt value of “232323232345”. Although it was too short, zero values were added to complete it.



- This is the recent ciphertext found in “dme.txt” after adding the salt value.

d.

```
OpenSSL> enc -des-cbc -in C:\Users\USER\Documents\eme.txt -out dme-cbc.txt -k password
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
OpenSSL>
```

- The command that was used in encrypting “eme.txt” using DES in CBC mode to produce “dme-cbc.txt” using the same key which is “password”.

- CBC is an acronym for Cipher Block Chaning

Observation: We were told to encrypt using the ebc mode, but this mode is not known. So it is the ECB(Electronic Codebook) mode that was used.

e.

```
OpenSSL> enc -des-ecb -in C:\Users\USER\Documents\eme.txt -out dme-ecb.txt -k password
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
OpenSSL>
```

- This is the command used to encrypt “eme.txt” using DES in ECB mode to produce “dme-ecb.txt”; with the same key being password.

dme-ecb - Notepad

File Edit Format View Help

```
Salted__êæûÐ½PþªIÑ
V"õ{ì¨ ˜9„•}
```

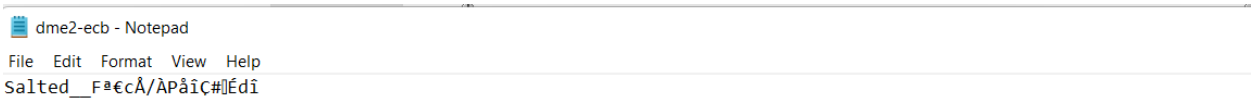
- This is the “dme-ecb” with the ciphertext in it.

f.

```
OpenSSL> enc -des-ecb -in C:\Users\USER\Documents\eme2.txt -out dme2-ecb.txt -k password
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
OpenSSL>
```

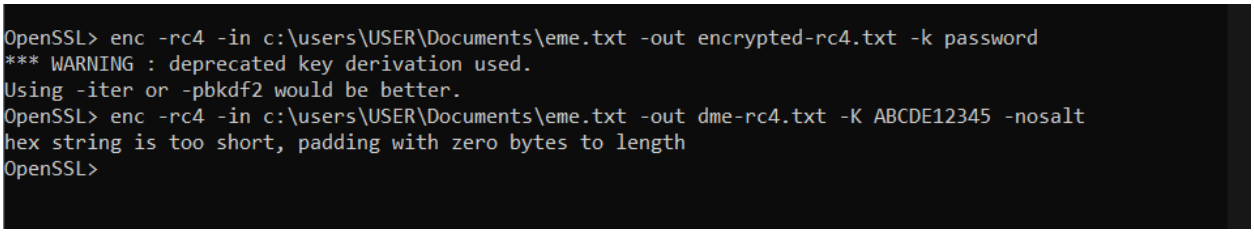
- This is the command to encrypt “eme2.txt” using DES in EBC mode to produce “dme2-ebc.txt” with the key being password

g.



- This is the “dme2-ecb.txt” result and it’s not the same as the “dme-ecb.txt”

h.



- This is the command for encrypting eme.txt” using RC4 with a 40bit key “ABCDE12345” which was too small making additional zeros to be added.

- This is the “dme-rcf.txt” that has the encrypted ciphertext of what s written in the “eme.txt”.

```
OpenSSL> enc -rc4 -d -in C:\Users\USER\dme-rc4.txt -out C:\Users\USER\emed-rc4.txt -K ABCDE12345 -nosal
t
hex string is too short, padding with zero bytes to length
OpenSSL>
```

- Command to decrypt the ciphertext in “dme-rc4.txt” and putting the plaintext in a new text file “emed-rc4.txt”. As seen I used the same 40-bit key used in encrypting for decrypting the text file.

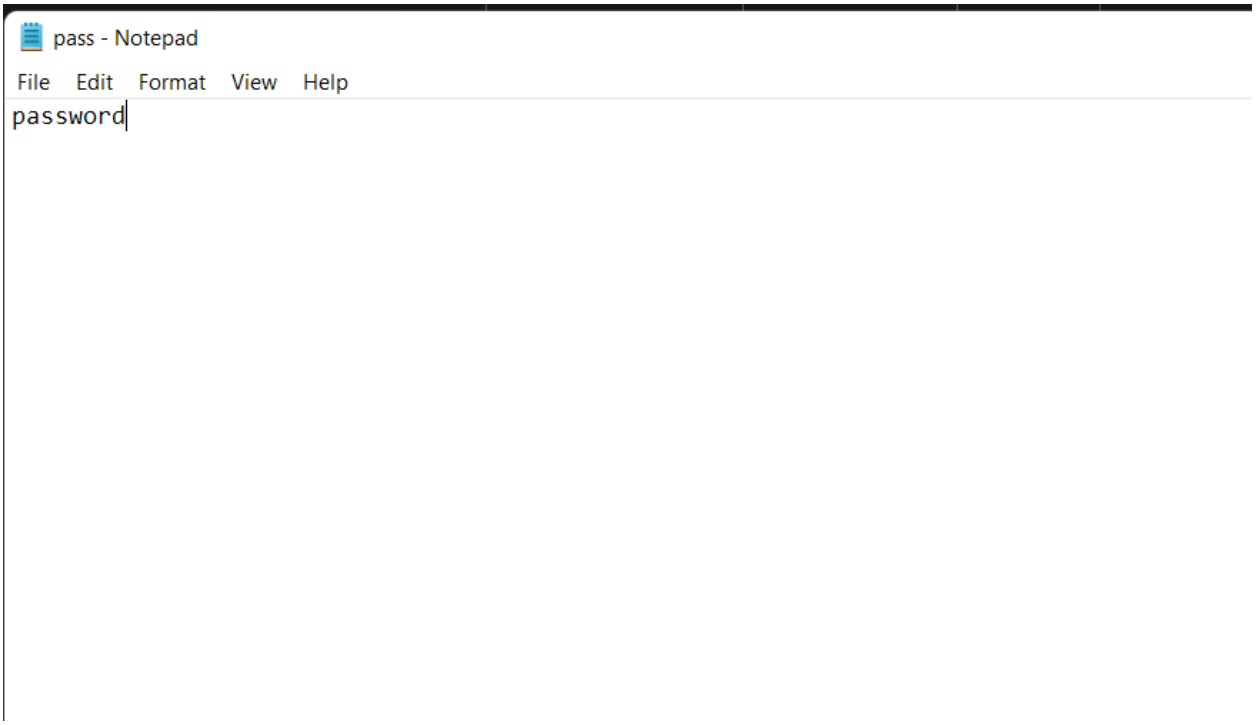
emed-rc4 - Notepad

File Edit Format View Help

```
CyberSecurity
```

- This is the decrypted file “emed-rcf.txt” that was created and as seen, it contains the same text as the one in “eme.txt”.

i.



- This is the text file ”pass.txt” which contain password

```
OpenSSL>  enc -des -in c:\users\USER\Documents\pass.txt -out dmepass.txt -S 232323232345
enter des-cbc encryption password:
Verifying - enter des-cbc encryption password:
hex string is too short, padding with zero bytes to length
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
OpenSSL>
```

- This is the command used to encrypt “pass.txt” with the password “password” using the salt value of “232323232345”. Although it was too short, zero values were added to complete it.

PART 3

a.

The PEM(Privacy Enhanced Mail) is a file or files used to store SSL certificates and their associated private keys.

b.

```
writing RSA key
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEAv0ss4FcOOIkGBN4dfI1rFqeeaPvT7528Ngx0qaAPwFGjymco
xWf5MslDQuz0JtBV8FAW8fNNBLhdpO/QhP2D6/VoD+HSHzitinSqh2ciB/lViCGm
J3EEeptK5DEhwQ98/20LLX9m8g9kjYBM0OoMZxaJu3xyNjYb1WEedJUIzrbzK3ix
rjVP5BvHrqlKnu9kiH6lK90s42SwqlF2UhKXRgPl5dsl+K4wdmaZtFFi6j3U5ogq
UjtTCIW+FlxHmfrwzxTyx5pvLz4HKgKoMCgEaH1S/vG78jCGnMAwYyUvgZHareGO
28m87jPguYnNXyrRG+Zy9hyzNQ4g9FLQPNKIvwIDAQABAoIBAH4uunVpN2o3iI7h
bRXhuwgAnU+m/dALfkEwyiux93Hm6GH01kIRKaJEfGfDIfIQbv/L9La4xM8eRjcs
MaHoEIMvJ+uCDDaoDmFMP8h9UUdtQuw/NfjCZi/yC96mQhu1ecCLT1o9tcxqNuvX
+9O671940HWpyzTTUFzaaVqBjIO9qYPfWeJofTN1RRpL7C1LKcrln6xY1OfE1a4/
HUxAO/UNJ3MMy3Q2vhHdZfB5ig90M3qUEJqmqtT/Zv+131oSFE3DkaQl8+GwxCc9
rH3CITRihM+yiyJsxUaB9e75C1CbgAhAL9RWSqGH2uRA6EaX4FxH8jahM0V5TY35
UmW/dyECgYEA8KkgOagFm3cjZZ7ilzV5Xs9UHExsDGArCIPqujwl/VzSUzefQ8Z+
jVN3VVu90Tf0AU8C53YXA9GAauh/WmOB/K71ylk4cVkK8oVpg2Zbc/8X5imOxnrx
9HHDQaK0yB+uXiLae5B+HZO+k8mmcTPbyRFvtSgqHYkekGli5d7rpGkCgYEAy3yG
dYbOUSQO3utyGVfFQvpqiMQy1X3TeSjxZxBiWW01PdTm5AacuisxWwb9INIfXJIW
R4Wm8pyREWGXPogOA+n5MNeCXClWhIeYHWm+W58h4rxOOkJ7S4ZjyrrOvJNd0MbD
/hpPN5qu7dsUfwEV1TgMUtFNhUarXOimjcpp/ucCgYAU9eYEn4m9fKbsluYNXW7w
vwYIYO+YNFgLKdIr3klAw/Ddpp6MfKFRFc/y80aZDlKGOqd6GWLvxjN1HELbUIdl
xqdJtHOiC6tUbCJAe43DkJg4R65TwymHrNDgypcMRxYcxNRapr8VPGUA4jNI8MdX
v9kG4jSo1eR/l24xrzS+4QKBgGWTvN8wMw+oK1A2QhfKi6sarFGndyTBke46dP8Y
269+Z4R9IeEUN+prI1EoVPWsToF8X2jGnAciBvyhNcm+SxjDFGTAP8/XxtuTgGi2
X5MHBfhUPcyxYTDbXjdWN1tFSSCuCCpxYbgFHGuZ4Eph9U8hkm1JFCSCmYpgjLvd
HjpjAoGADB0YFcpBlg6wrMofKwqLUvx1ogimAOarZODLCYgTS+c5VB70mrosZOpB
TVO92xr16gSAOpv/epzHxHvof1HcLrryxAyYk00lAU+C6rfjqQftSnFuVM54ooAX
setsVCy0HYb2siRp12d2duae4E/0O2vyKxcRHa/yzr46YWfMJ5s=
-----END RSA PRIVATE KEY-----
```

- This is the RSA private key generated for the modulus of 1024 bits and an exponent of 3 saved in “private_key.pem”. There were some parameters that were used in its generation, such as: modulus, private and public exponent, prime factors and others.
- I was also told to put a secret pass phrase that I would be using for decrypting.

```
writing RSA key
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAv0ss4FcOOIkGBN4dfI1r
FqeeaPvT7528Ngx0qaAPwFGjymcoxWf5MslDQuz0JtBV8FAW8fNNBLhdpO/QhP2D
6/VoD+HSHzitinSqh2ciB/lViCGmJ3EEeptK5DEhwQ98/20LLX9m8g9kjYBM0OoM
ZxaJu3xyNjYb1WEedJUIzrbzK3ixrjVP5BvHrqlKnu9kiH6lK90s42SwqlF2UhKX
RgPl5dsl+K4wdmaZtFFi6j3U5ogqUjtTCIW+FlxHmfrwzxTyx5pvLz4HKgKoMCgE
aH1S/vG78jCGnMAwYyUvgZHareGO28m87jPguYnNXyrRG+Zy9hyzNQ4g9FLQPNKI
vwIDAQAB
-----END PUBLIC KEY-----
OpenSSL>
```

- This is the RSA public key generated and stored in “public_key.pem”. It also possessed a modulus of 1024 bits and an exponent of 3.

c.

```
OpenSSL> rsautl -encrypt -in C:\Users\USER\Documents\eme.txt -out encrypted_emetxt.bin -inkey public_key.pem -pubin
```

- This is the command used to encrypt the text file “eme.txt”. It is encrypted into a bin directory ”encrypted_emetxt.bin” using the public key located in “public_key.pem”
- At this point, the result is not in readable format, but will be changed so as to view it in readable format.

```
OpenSSL> enc -base64 -in C:\Users\USER\encrypted_emetxt.bin -out encrypted_emetxt.txt
```

- This is the command that was given to change the “encrypted_emetxt.bin” in bin to a readable format in”encrypted_emetxt.txt”.

encrypted_emetxt - Notepad

File Edit Format View Help


```
Yh8JJ1XVNDYklXeV/YAPpfNxu0NPsOrxzEwPiNPTO5GCsv5Wcw+6G+FK9aoewrL/
IRlQYRY21FC84DBS34DQ4LFNr2RikevbL0FgGEvBGVy/THUPK3EQkSoeJPAemVXv
AKWBuvkkCxdCRTkrQGew2O/kNCjYfisod6Itgq7C1RPeOjkjuwrUr4AFNpNxH5/1
mjEsUC60edpvm9BSToB5KYyXd3Uw2hpfEHB5g3Sj4t/KJQGfaZIoiYvk/bkLQU90
NfcUsgwllropCT8BrkwSkJX/Y2YgOWay5ShhYBTLgrvja1RSqcCPIKO4p7IukhXD
TYWKG9ySy255qEFpG0z1sQ==
```


- This is the ciphertext of our ”eme.txt” that is now in readable format found in “encrypte_emetxt.txt”

```
OpenSSL> rsautl -decrypt -in C:\Users\USER\encrypted_emetxt.bin -out decrypted_emetxt.txt -inkey private_key.pem
Enter pass phrase for private_key.pem:
```

- This is the command to decrypt the ciphertext in “encrypted_emetxt.txt” and give the result in “decrypted_emetxt.txt” using the private key I created
- I also inputted my pass phrase

decrypted_emetxt - Notepad

File Edit Format View Help

```
CyberSecurity
```

- This is the decrypted text in “decrypted_emetxt.txt” and as shown, it is the same as the original ”eme.txt”.
- To decrypt a text that was encrypted in public key, the private key will be needed.

d.

```
OpenSSL> rsautl -encrypt -in C:\Users\USER\Documents\eme.txt -out encrypted_emetxt.bin -inkey private_key.pem
Enter pass phrase for private_key.pem:
OpenSSL> enc -base64 -in C:\Users\USER\encrypted_emetxt.bin -out encrypted_emetxt.txt
OpenSSL>
```

- Above is the command used to encrypt “eme.txt” using a private key from the “private_key.pem”
- I was also once again asked to put my pass phrase
- The text in “eme.txt” was encrypted into “encrypted_emetxt.txt” and converted into readable form.

encrypted_emetxt - Notepad

File Edit Format View Help


```
gmcpttFGMJmMVDreoZ2OE0nyryzuzgXfP1ubbULKpiRQZfBfCx4EU7ixJpB0xEcr
6M0K1pas6qCaEDkjHnkgRD5IA5YUoTt9wr2vGAiCGE3957YnEug89ZaGc/UY1qV9
9ClpnL44jRzhTSWD0ZjU8ziIocgG131ODY0DhcfuJSFb7GDOpUFKEYN1rd1XI0u1
Hgd1DBrwOV5RjbFoXJ1TcXQMd243APc7Eu3LvpXky6ESGHcR5n75ZAe3uQTGVQl1
tzULBh+gtlajvu1WG/6/xMAnK4e8sIjBhBjr7MaL7sDnOH8thzoMcIVTMXKku03a
eMBDUjajuJWPdbYGZS7GGw==
```


- This is the updated encrypted text from “encrypted_emetxt.txt” after encrypting with the private key.

```
OpenSSL> rsautl -decrypt -in C:\Users\USER\encrypted_emetxt.bin -out decrypted_emetxt.txt -inkey private_key.pem
Enter pass phrase for private_key.pem:
OpenSSL>
```

- Command to decrypt the ciphertext in “encrypted_emetxt.txt” to “decrypted_emetxt.txt”. It was decrypted using the private key in”private_key.pem”. And the secret pass phrase was required.
- After decrypting, it was noted that the results were the same in “decrypted_emetxt.txt” and “eme.txt”.

PART 4

a.

```
C:\Users\USER>openssl
OpenSSL> dgst -md5 -hex -out eme_md5.txt C:\Users\USER\Documents\eme.txt
OpenSSL>
```

- This is that command that was used to hash “eme.txt” using MD5.

eme_md5

File Edit View

```
MD5(C:\Users\USER\Documents\eme.txt)= 45115c7bcaa67a28f5b5cff1188fdffe
```

- This is the “eme_md5.txt” text document containing the hashed value of the text in “eme.txt”.

b.

```
OpenSSL> dgst -sha1 -hex -out eme_sha1.txt C:\Users\USER\Documents\eme.txt
OpenSSL>
```

- The command to hash “eme.txt” using SHA1 with “**dgst**” representing the openssl command for digest operations.
- The “eme.txt” is hashed into a text document”eme_sha1.txt”.

```
eme_sha1
File    Edit    View

SHA1(C:\Users\USER\Documents\eme.txt)= 3550f04b326ba938dc4a8141d093cdb7ae8300ed
```

- This is the hash value found in “eme_sha1.txt”.

c.

```
OpenSSL> dgst -sha1 -sign private_key.pem -out eme.txt.sha1 C:\Users\USER\Documents\eme.txt
Enter pass phrase for private_key.pem:
OpenSSL>
```

- This is the command to bring out the SHA-1 hash of “” and it is thereby assigned a private key.
- After successfully putting my secret pass phrase, the result was saved in a file named”eme.txt.sha1”.

```
OpenSSL> dgst -sha1 -verify public_key.pem -signature eme.txt.sha1 C:\Users\USER\Documents\eme.txt
Verified OK
OpenSSL>
```

- I also went on to verify that my pubic key would open the hashed file and it was verified **OK**.

PART 5

a.

- A CSR(Certificate signing Request) is one of the first steps towards getting a SSL/TLS (Secure Sockets Layer/Transport Layer Security) certificate.
- The CSR is usually generated on the server where the SSL/TLS certificate will be used.

b.

- The ASN.1(Abstract Syntax Notation One) is a standard interface used for defining data structures in a cross-platform way.
- In the context of certificate, it is used to define the structure and encoding rules for certificate data.

c.

- DER(Distinguished Encoding Rules) is used with ASN.1 to represent structured data in a binary format.
- It is widely used to encoding format to serialize certificate data.

d.

```
C:\Users\USER>openssl
OpenSSL> req -new -x509 -key private_key.pem -out self_signed_cert.pem
Enter pass phrase for private_key.pem:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:NG
State or Province Name (full name) [Some-State]:Akwa Ibom
Locality Name (eg, city) []:Uyo
Organization Name (eg, company) [Internet Widgits Pty Ltd]:2GAY Entrprises
Organizational Unit Name (eg, section) []:Section
Common Name (e.g. server FQDN or YOUR name) []:FCB
Email Address []:favcharles2003@gmail.com
OpenSSL>
```

- This is the command used to create the X.509 certificate with the private key pass phrase that was set earlier
- I was also asked specific questions like Country Name, State or Province, etc.
- And the self-signed certificate was generated in “self_signed_cert.pem”

`

self_signed_cert

File Edit View


```
-----BEGIN CERTIFICATE-----
MIIEBzCCAu+gAwIBAgIUTs3VDYWVdJ7MDNE8cGPPsKJOzL4wDQYJKoZIhvcNAQEL
BQAwgZIxCzAJBgNVBAYTAk5HMRIwEAYDVQQIDAlBa3dhIElib20xDDAKBgNVBAcM
A1V5bzEYMBYGA1UECgwPMkdBWSBFbnRycHJpc2VzMRAwDgYDVQQLDAdTZWN0aW9u
MQwwCgYDVQQDDANGQ0IxJzAlBgkqhkiG9w0BCQEWGGZhdmNoYXJsZXMyMDAzQGdt
YWlsLmNvbTAeFw0yMzExMDQyMDMzNThaFw0yMzEyMDQyMDMzNThaMIGSMQswCQYD
VQQGEwJORzESMBAGA1UECAwJQWt3YSBJYm9tMQwwCgYDVQQHDANVeW8xGDAWBgNV
BAoMDzJHQVkgRW50cnByaXNlczEQMA4GA1UECwwHU2VjdGlvbjEMMAoGA1UEAwwD
RkNCMScwJQYJKoZIhvcNAQkBFhhmYXZjaGFybGVzMjAwM0BnbWFpbC5jb20wggEi
MA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC/SyzgVw44iQYE3h18jWsWp55o
+9Pvnbw2DHSpoA/AUaPKZyjFZ/kyyUNC7PQm0FXwUBbx800EuF2k79CE/YPr9WgP
4dIfOK2KdKqHZyIH+VWIIaYncQR6m0rkMSHBD3z/bQstf2byD2SNgEzQ6gxnFom7
fHI2NhvVYR50lQjOtvMreLGuNU/kG8euqUqe72SIfqUr3SzjZLCqUXZSEpdGA+Xl
2yX4rjB2Zpm0UWLqPdTmiCpSO1MIhb4WXEeZ+vDPFPLHmm8vPgcqAqgwKARofVL+
8bvyMIacwDBjJS+Bkdqt4Y7bybzuM+C5ic1fKtEb5nL2HLM1DiD0UtA80oi/AgMB
AAGjUzBRMB0GA1UdDgQWBBRQ2deQFcSijpR7xacvWqWO7YBD0jAfBgNVHSMEGDAW
gBRQ2deQFcSijpR7xacvWqWO7YBD0jAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3
DQEBCwUAA4IBAQB1gjtaFfgodDqYtbXmuCTLRUIRvOFp8K3525ll+BGYVCquUyBD
LZMVECu/G4AEOvDeP01htp4rmWpql26hYkwkxc+Kbg1JxqxnzQdIbAb/4qXDfsS+
i8qPG8pCa/AFQoNFu1pa7c3eIz9PJN/IizcdFlba/azgfoJxqgnvFD3UWwcw5aoJ
wWgRWgf/pJVCegPLdiNgG1VG2URvtjt7RQanJG25qMIWbDiHvHSvAMw9EcG7GsnJ
c3/wmjvzsyYbExqkNl92fUMLk9WPbOGsHLIdnMJb48kwGp+IbkotV9ke2QZkTM53
qLzXlwlNd0xU/QrwfrKQ2ai6ChX2tiNGbX7w
-----END CERTIFICATE-----
```


Ln 1, Col 1

- This is the self-signed certificate that was created in “seld_signed_cert.pem”