CNS Oral/ Practical Exam Questions

1) Write a python program to encrypt a plain text using ceaser cipher

```
import string
all_letters= string.ascii_letters #A list containing all characters

dict1 = {}
key = 3
for i in range(len(all_letters)):
    dict1[all_letters[i]] = all_letters[(i+key)%len(all_letters)]
plain_txt= "xie is best"
cipher_txt=[]
# loop to generate ciphertext
for char in plain_txt:
    if char in all_letters:
        temp = dict1[char]
        cipher_txt.append(temp)
    else:
        temp =char
        cipher_txt.append(temp)

cipher_txt= "".join(cipher_txt)
print("Cipher Text is: ",cipher_txt)
```

2) Write a program to encrypt the given text using Railfence technique

```
cipher_text=""
def railfence(plain_text,key):
    if key==2:
        return (plain_text[::2]+plain_text[1::2])
    else:
        return "number of rails not supported"
cipher_text= railfence("xie is best ", 2)
print(cipher_text)
```

3) Write a program to decrypt the text “aLH” using ceaser cipher

```
import string
all_letters= string.ascii_letters #A list containing all characters

dict1 = {}
key = 3
for i in range(len(all_letters)):
    dict1[all_letters[i]] = all_letters[(i-key)%len(all_letters)]
cipher_txt= "aLH"
plain_txt= []
# loop to generate decryptedtext
for char in cipher_txt:
    if char in all_letters:
        temp = dict1[char]
        plain_txt.append(temp)
    else:
        temp =char
        plain_txt.append(temp)

plain_txt= "".join(plain_txt)
print("Decrypted Text is: ",plain_txt)
```

4) Perform brute force attack on “aLH LV EHVW”
(Note* Any encrypted text will be given)

```
import string
all_letters= string.ascii_letters #A list containing all characters

dict1 = {}
key = 0
for key in range(0,26):
  for i in range(len(all_letters)):
    dict1[all_letters[i]] = all_letters[(i-key)%len(all_letters)]

  plain_txt= "aLH LV EHVW"
  cipher_txt=[]
# loop to generate ciphertext
  for char in plain_txt:
    if char in all_letters:
      temp = dict1[char]
      cipher_txt.append(temp)
    else:
      temp =char
      cipher_txt.append(temp)

  cipher_txt= "".join(cipher_txt)
  print("Cipher Text is:",cipher_txt)
  key = key+1
```

5) Perform product cipher on the encrypted message “aLH LV EHVW” using railfence technique
(Note* Any encrypted text will be given)

```
cipher_text=""
def railfence(plain_text,key):
    if key==2:
        return (plain_text[::2]+plain_text[1::2])
    else:
        return "number of rails not supported"
cipher_text= railfence("aLH LV EHVW ", 2)
print(cipher_text)
```

this is Decryption but have not been asked

```
#Decryption:
for i in range(2):
    block1=cipher_text[:6:]
    #print(block1)
    block2=cipher_text[6::]
    #print(block2)
    x1= (block1[0]+block2[0]+block1[1]+block2[1]+block1[2]+block2[2]+
         block1[3]+block2[3]+block1[4]+block2[4]+block1[5]+block2[5])
    print("After Decryption",i+1,"time :",x1)
    cipher_text = x1
```

6) Perform encryption and decryption of a Plain text message using Open SSL Commands

```
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ cat >> msg
HI IN AM IN YOUR PC HOW ARE YOU ?
^C
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ cat msg
HI IN AM IN YOUR PC HOW ARE YOU ?
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ openssl version
OpenSSL 1.1.1q  5 Jul 2022
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ openssl list -cipher-commands
aes-128-cbc       aes-128-ecb       aes-192-cbc       aes-192-ecb
aes-256-cbc       aes-256-ecb       aria-128-cbc      aria-128-cfb
aria-128-cfb1     aria-128-cfb8     aria-128-ctr      aria-128-ecb
aria-128-ofb      aria-192-cbc      aria-192-cfb      aria-192-cfb1
aria-192-cfb8     aria-192-ctr      aria-192-ecb      aria-192-ofb
aria-256-cbc      aria-256-cfb      aria-256-cfb1     aria-256-cfb8
aria-256-ctr      aria-256-ecb      aria-256-ofb      base64
bf                bf-cbc            bf-cfb            bf-ecb
bf-ofb            camellia-128-cbc  camellia-128-ecb  camellia-192-cbc
camellia-192-ecb  camellia-256-cbc  camellia-256-ecb  cast
cast-cbc          cast5-cbc         cast5-cfb         cast5-ecb
cast5-ofb         des               des-cbc           des-cfb
des-ecb           des-ede           des-ede-cbc       des-ede-cfb
des-ede-ofb       des-ede3          des-ede3-cbc      des-ede3-cfb
des-ede3-ofb      des-ofb           des3              desx
idea              idea-cbc          idea-cfb          idea-ecb
idea-ofb          rc2               rc2-40-cbc        rc2-64-cbc
rc2-cbc           rc2-cfb           rc2-ecb           rc2-ofb
rc4               rc4-40            seed              seed-cbc
seed-cfb          seed-ecb          seed-ofb          sm4-cbc
sm4-cfb           sm4-ctr           sm4-ecb           sm4-ofb

ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ openssl enc -aes-256-cbc -base64 -in msg -out emsg
enter aes-256-cbc encryption password:
Verifying - enter aes-256-cbc encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ ls
emsg  msg
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ cat emag
cat: emag: No such file or directory
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ cat emsg
U2FsdGVkX19ZyevOK4fHGEnRuszNAhUGaFPEALhP/qdqOSaa8416tUJ5qT6w/BAY
98jZQ5GlyQLSJKGltYyUBg==
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ openssl enc -aes-256-cbc -d -base64 -in emsg -out dmsg
enter aes-256-cbc decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
ubuntu@ubuntu-VirtualBox:~/Desktop/HI$ ls
dmsg  emsg  msg
```

7) Perform RSA encryption and decryption

```
openssl genrsa -out Private.pem 2048
cat Private.pem
openssl rsa -in Private.pem -text -noout
openssl rsa -in Private.pem -pubout -out Public.pem
ls
cat Public.pem
openssl rsault -encrypt -in msg -out emsg -inkey Public.pem -pubin
openssl rsautl -encrypt -in msg -out emsg -inkey Public.pem -pubin
ls
cat emsg
openssl rsautl -decrypt -in emsg -out dmsg -inkey Private.pem
ls
cat dmsg
ls
```

8) Find the person who performed password attack and modified text on your machine using network commands (exp - 6)
(Note* Attack will be performed / may ask you perform and the content will be in encrypted form. You need to decrypt and view the message)

9) Analyze the http and https packet using wireshark in ubuntu

10) Analyze the ssh and ftp connection using wireshark in windows

```
ubuntu@ubuntu-VirtualBox:~$ ftp ftp.cdc.gov
Trying [64:ff9b::c6f6:756a]:21 ...
ftp: Can't connect to `64:ff9b::c6f6:756a:21': Connection timed out
Trying 198.246.117.106:21 ...
Connected to ftp.cdc.gov.
220 Microsoft FTP Service
Name (ftp.cdc.gov:ubuntu): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> ascii
200 Type set to A.
ftp> dir
229 Entering Extended Passive Mode (|||50528|)
125 Data connection already open; Transfer starting.
drwxrwxrwx   1 owner    group               0 Nov  9 14:50 pub
226 Transfer complete.
ftp> get Readme
local: Readme remote: Readme
229 Entering Extended Passive Mode (|||50529|)
125 Data connection already open; Transfer starting.
100% |***********************************|  1428        5.66 KiB/s    00:00 ETA
226 Transfer complete.
WARNING! 36 bare linefeeds received in ASCII mode.
File may not have transferred correctly.
1428 bytes received in 00:00 (5.65 KiB/s)
ftp> quit
221 Goodbye.
ubuntu@ubuntu-VirtualBox:~$
```

ftp

No.	Time	Source	Destination	Protocol	Length	Info
33	120.872999567	198.246.117.106	192.168.43.9	FTP	93	Response: 220 Microsoft FTP Service
37	131.163494186	192.168.43.9	198.246.117.106	FTP	82	Request: USER anonymous
38	131.405235479	198.246.117.106	192.168.43.9	FTP	138	Response: 331 Anonymous access allowed, send identity (e-mail name) as password.
40	144.181617069	192.168.43.9	198.246.117.106	FTP	91	Request: PASS anything@gmail.com
41	144.472209205	198.246.117.106	192.168.43.9	FTP	87	Response: 230 User logged in.
43	144.483329063	192.168.43.9	198.246.117.106	FTP	72	Request: SYST
44	144.738779959	198.246.117.106	192.168.43.9	FTP	82	Response: 215 Windows_NT
46	144.739021801	192.168.43.9	198.246.117.106	FTP	72	Request: FEAT
47	144.987345251	198.246.117.106	192.168.43.9	FTP	100	Response: 211-Extended features supported:
49	144.987788886	198.246.117.106	192.168.43.9	FTP	84	Response: LANG EN*
50	144.987789467	198.246.117.106	192.168.43.9	FTP	119	Response: AUTH TLS;TLS-C;SSL;TLS-P;
51	144.987789584	198.246.117.106	192.168.43.9	FTP	73	Response: HOST
52	144.987789703	198.246.117.106	192.168.43.9	FTP	103	Response: SIZE
59	150.598942304	192.168.43.9	198.246.117.106	FTP	74	Request: TYPE A
60	150.894076636	198.246.117.106	192.168.43.9	FTP	86	Response: 200 Type set to A.
62	153.660660187	192.168.43.9	198.246.117.106	FTP	72	Request: EPSV
63	153.949319856	198.246.117.106	192.168.43.9	FTP	114	Response: 229 Entering Extended Passive Mode (|||50528|)
68	154.183494027	192.168.43.9	198.246.117.106	FTP	72	Request: LIST
69	154.426516534	198.246.117.106	192.168.43.9	FTP	120	Response: 125 Data connection already open; Transfer starting.
71	154.426517199	198.246.117.106	192.168.43.9	FTP	90	Response: 226 Transfer complete.
80	173.594128986	192.168.43.9	198.246.117.106	FTP	79	Request: SIZE Readme
81	173.899025239	198.246.117.106	192.168.43.9	FTP	76	Response: 213 1428
83	173.899390350	192.168.43.9	198.246.117.106	FTP	72	Request: EPSV
85	174.156391224	198.246.117.106	192.168.43.9	FTP	114	Response: 229 Entering Extended Passive Mode (|||50529|)
91	174.414362429	192.168.43.9	198.246.117.106	FTP	79	Request: RETR Readme
92	174.675639921	198.246.117.106	192.168.43.9	FTP	120	Response: 125 Data connection already open; Transfer starting.

```
▸ Frame 37: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface enp0s3, id 0
▸ Ethernet II, Src: PcsCompu_94:da:71 (08:00:27:94:da:71), Dst: a6:c9:39:4b:ad:2f (a6:c9:39:4b:ad:2f)
▸ Internet Protocol Version 4, Src: 192.168.43.9, Dst: 198.246.117.106
▸ Transmission Control Protocol, Src Port: 39352, Dst Port: 21, Seq: 1, Ack: 28, Len: 16
▾ File Transfer Protocol (FTP)
  ▾ USER anonymous\r\n
      Request command: USER
      Request arg: anonymous
  [Current working directory: ]
```

ftp

No.	Time	Source	Destination	Protocol	Length	Info
33	120.872999567	198.246.117.106	192.168.43.9	FTP	93	Response: 220 Microsoft FTP Service
37	131.163494186	192.168.43.9	198.246.117.106	FTP	82	Request: USER anonymous
38	131.405235479	198.246.117.106	192.168.43.9	FTP	138	Response: 331 Anonymous access allowed, send identity (e-mail name) as password.
40	144.181617069	192.168.43.9	198.246.117.106	FTP	91	Request: PASS anything@gmail.com
41	144.472209205	198.246.117.106	192.168.43.9	FTP	87	Response: 230 User logged in.
43	144.483329063	192.168.43.9	198.246.117.106	FTP	72	Request: SYST
44	144.738779959	198.246.117.106	192.168.43.9	FTP	82	Response: 215 Windows_NT
46	144.739021801	192.168.43.9	198.246.117.106	FTP	72	Request: FEAT
47	144.987345251	198.246.117.106	192.168.43.9	FTP	100	Response: 211-Extended features supported:
49	144.987788886	198.246.117.106	192.168.43.9	FTP	84	Response: LANG EN*
50	144.987789467	198.246.117.106	192.168.43.9	FTP	119	Response: AUTH TLS;TLS-C;SSL;TLS-P;
51	144.987789584	198.246.117.106	192.168.43.9	FTP	73	Response: HOST
52	144.987789703	198.246.117.106	192.168.43.9	FTP	103	Response: SIZE
59	150.598942304	192.168.43.9	198.246.117.106	FTP	74	Request: TYPE A
60	150.894076636	198.246.117.106	192.168.43.9	FTP	86	Response: 200 Type set to A.
62	153.660660187	192.168.43.9	198.246.117.106	FTP	72	Request: EPSV
63	153.949319856	198.246.117.106	192.168.43.9	FTP	114	Response: 229 Entering Extended Passive Mode (|||50528|)
68	154.183494027	192.168.43.9	198.246.117.106	FTP	72	Request: LIST
69	154.426516534	198.246.117.106	192.168.43.9	FTP	120	Response: 125 Data connection already open; Transfer starting.
71	154.426517199	198.246.117.106	192.168.43.9	FTP	90	Response: 226 Transfer complete.
80	173.594128986	192.168.43.9	198.246.117.106	FTP	79	Request: SIZE Readme
81	173.899025239	198.246.117.106	192.168.43.9	FTP	76	Response: 213 1428
83	173.899390350	192.168.43.9	198.246.117.106	FTP	72	Request: EPSV
85	174.156391224	198.246.117.106	192.168.43.9	FTP	114	Response: 229 Entering Extended Passive Mode (|||50529|)
91	174.414362429	192.168.43.9	198.246.117.106	FTP	79	Request: RETR Readme
92	174.675639921	198.246.117.106	192.168.43.9	FTP	120	Response: 125 Data connection already open; Transfer starting.

```
▸ Frame 40: 91 bytes on wire (728 bits), 91 bytes captured (728 bits) on interface enp0s3, id 0
▸ Ethernet II, Src: PcsCompu_94:da:71 (08:00:27:94:da:71), Dst: a6:c9:39:4b:ad:2f (a6:c9:39:4b:ad:2f)
▸ Internet Protocol Version 4, Src: 192.168.43.9, Dst: 198.246.117.106
▸ Transmission Control Protocol, Src Port: 39352, Dst Port: 21, Seq: 17, Ack: 100, Len: 25
▾ File Transfer Protocol (FTP)
  ▾ PASS anything@gmail.com\r\n
      Request command: PASS
      Request arg: anything@gmail.com
  [Current working directory: ]
```

11) Analyze the open ports available in the network and make a report
12) Perform encryption and decryption of email security in Kleopatra

***VIVA Questions will be asked from the syllabus**