# NETWORK SECURITY PROJECT

**Topic: Image Ecryption using AES Algorithm and steganography**

# MADE BY

- AbssZy

# Introduction

- In computing, encryption is the method by which plaintext or any other type of data is converted from a readable form to an encoded version that can only be decoded by another entity if they have access to a decryption key. Encryption is one of the most important methods for providing data security, especially for end-to-end protection of data transmitted across networks.

# Purpose

- The purpose of this software is provide the security of Data for which no one except the customer can see the private information.

# User Interfaces

- 1. In this software, have two main buttons (Encrypt, decrypt) and a textbox for input paraphrase.
- 2. First tab i.e. encrypt tab will allow a user to select the image they wish to encrypt and provides the encrypted image.
- 3. Second tab i.e. decrypts option, will allow the user to decrypt the encrypted image.

# Hardware Interfaces

- Image security system is made in python so it is Hardware Independent software. There is no need of any hardware. It runs on any operating system and it’s not a heavy software so no extra RAM is needed. It’s required 16 MB RAM to executed software and run on any OS.

# Software Interfaces

- Implementation will be in python programming language that will execute in any system in which has a python compiler that is depend on operating system to system.

# System Features

- Encryption
- Decryption

# Encryption

- This system feature involves encrypting the image using Advanced Encryption System Algorithm which is symmetric.
- Function Requirements:
- 1. Firstly, the user types the paraphrase in the given input field.
- 2. Select the file using Encrypt option that he wants to encrypt.
- 3. Before pressing the Encrypt button, you must enter the key that helps to encrypt your file.
- 4. After selecting your file successfully, it gives the encrypted file.
  5. After encryption, file is saved in destination folder.

# Decryption

- This system feature involves decrypting the image using the same paraphrase and AES algorithm.
- Functional Requirement:
- 1. Select file using Decrypt option that you want to decrypt.
- 2. Enter the paraphrase value in key field; without entering the key value you cannot decrypt the image.
- 3. Press the decrypt button to get the decrypted file.

# Performance Requirements

- Most cryptographic ciphers rely on high computational cost operations. Therefore, keeping performance considerations in mind, for data encryption/decryption computational effort to encryption/decryption using Asymmetric key is very powerful compared to symmetric key algorithm. It provides more security compared to symmetric key and also performance of encrypting file is very good. It is general purpose software. But, for simplicity reasons we have chosen symmetric algorithm.

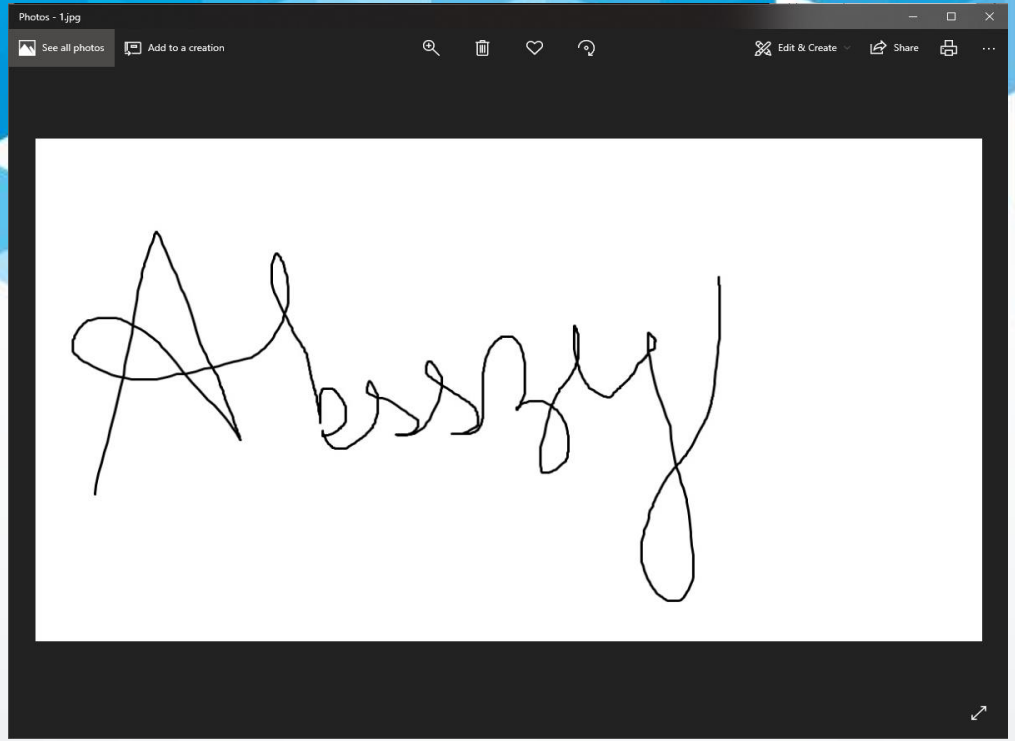
# Security Requirements

- User is required to remember his password that he/she used to encrypt data (or lock password safe) because most of secure cryptographic algorithms implemented in this suite are secure enough so that no algorithms better than brute-force can be used to recover lost password.

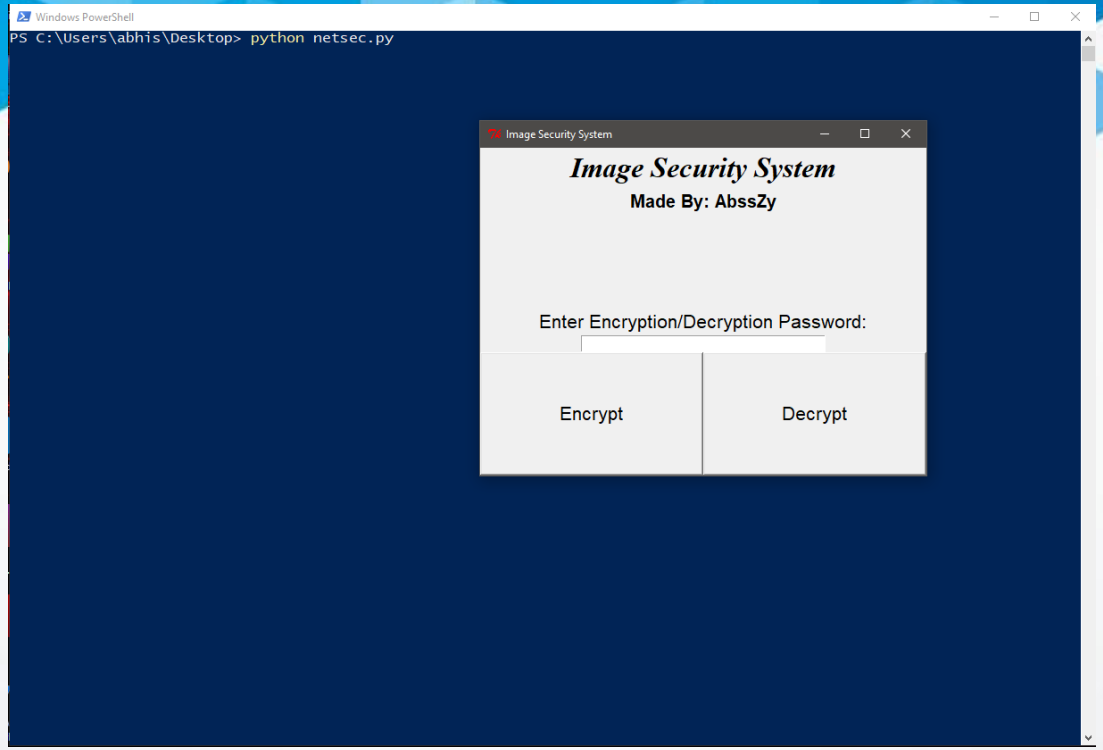
# Software Quality Attributes

- The source code should be properly documented, so that new developers will be able to understand the code as easily as possible. It should be easily available for every end user for better security.

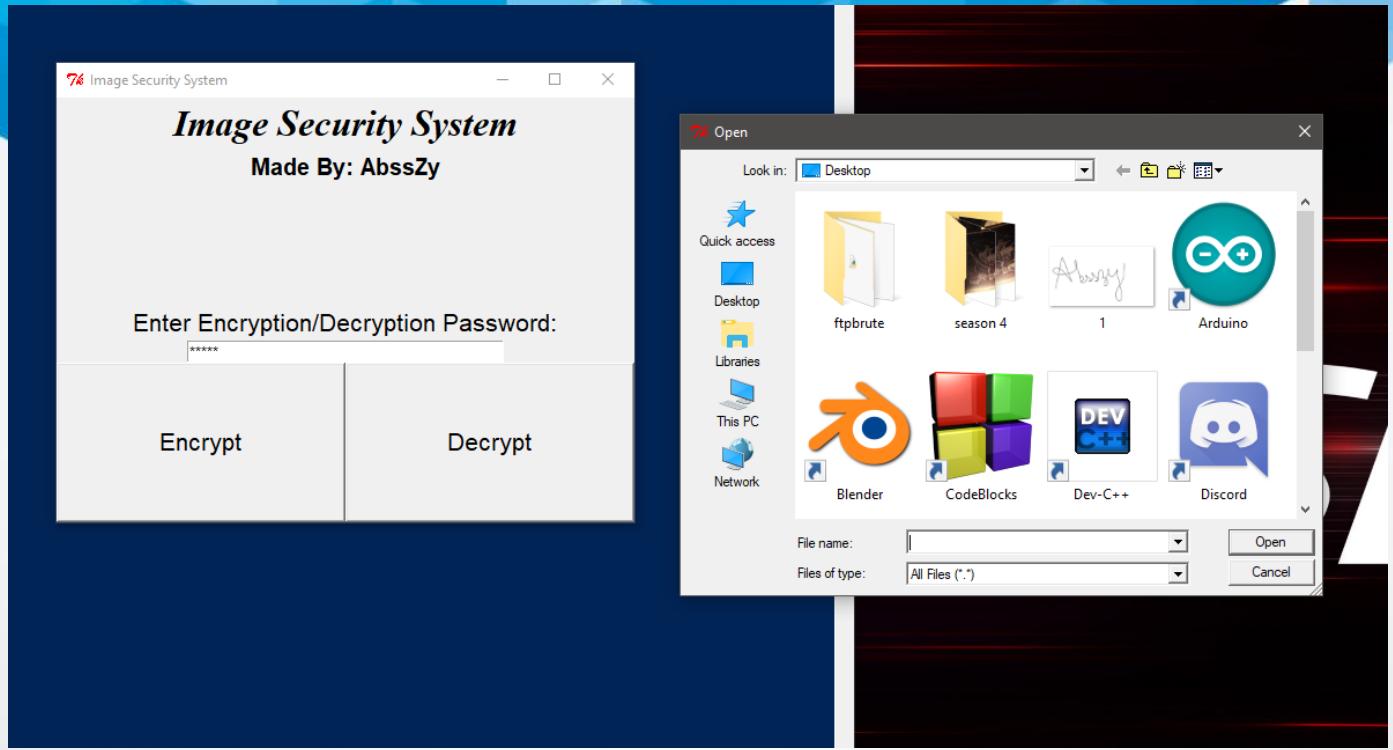
# Step 1:Select the file to be encrypted (the file should be in JPEG format)



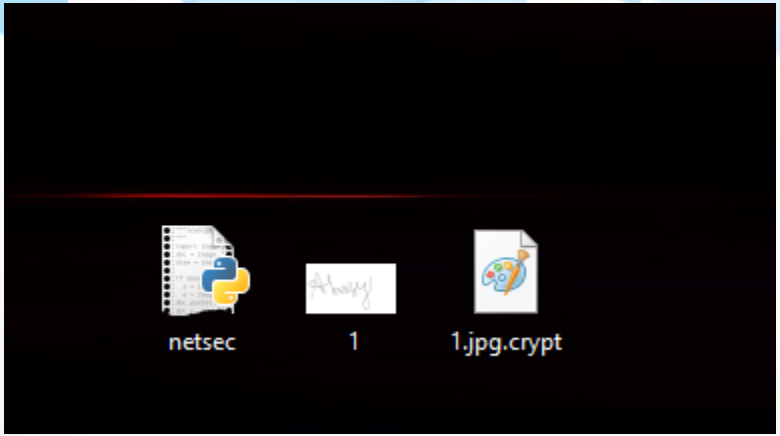
# Step2) Run the program



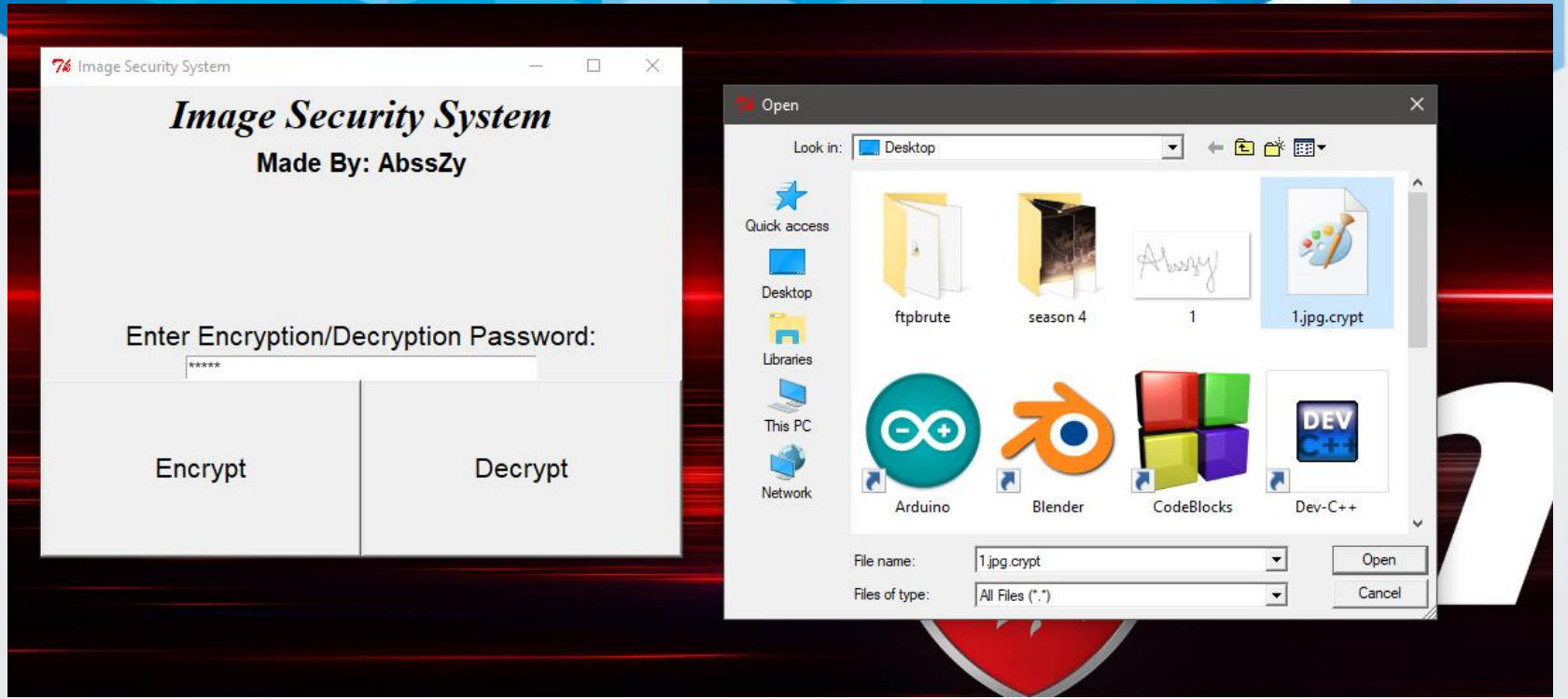
Step 3:Enter password that you want (it can be a string or numbers, here it is AB123).Click on “Encrypt” button and another dialog box will open, select the image and click open



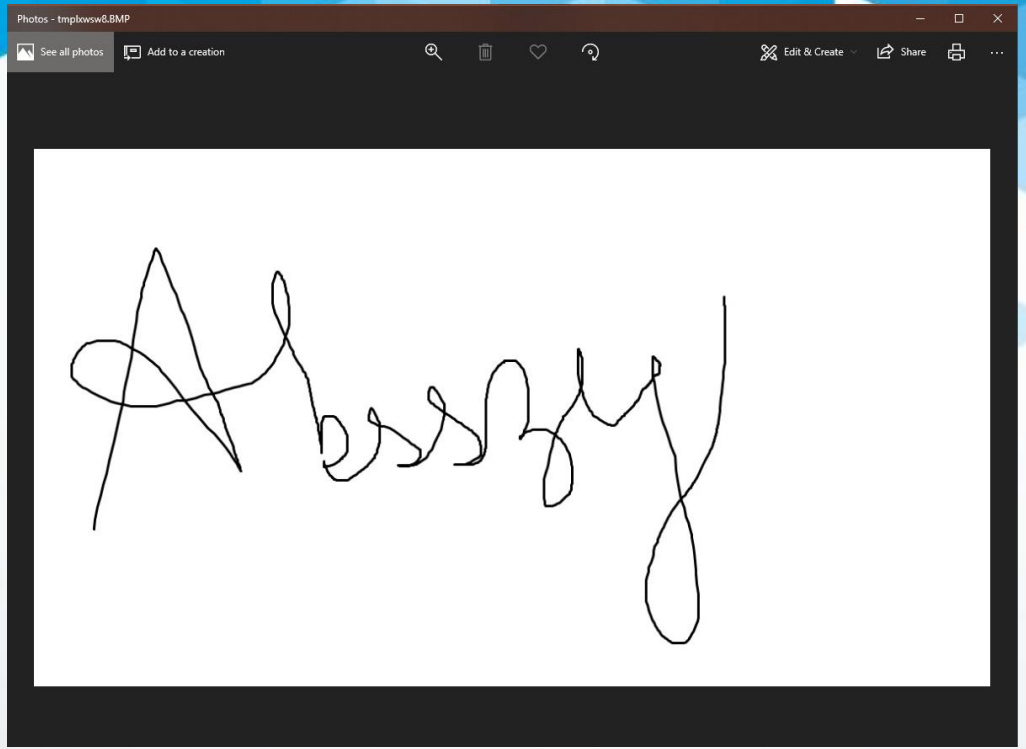
Step 4: When you press “Open”. The program will freeze and in sometime a new window will open containing a temporary file which will be the Image constructed by the program using the cypher text.
Go to the original file location and you will see a new file created there with same file name but not an executable. It will contain an extension of “.crypt”.



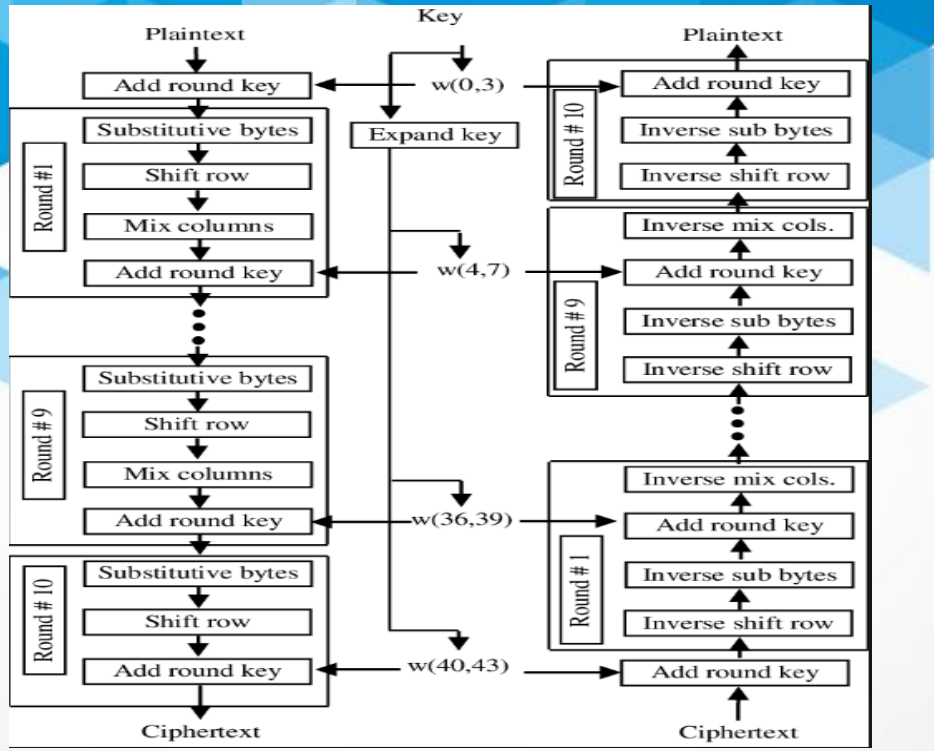
# Step 5: If you run the program again and enter the password (again here password is 12345).Click on Decrypt and a dialog box will open



# Step 6: Select the encrypted file and click on “Open” and wait for some time as program will free and a photo will open contain the original file with a random name.



# Working of AES

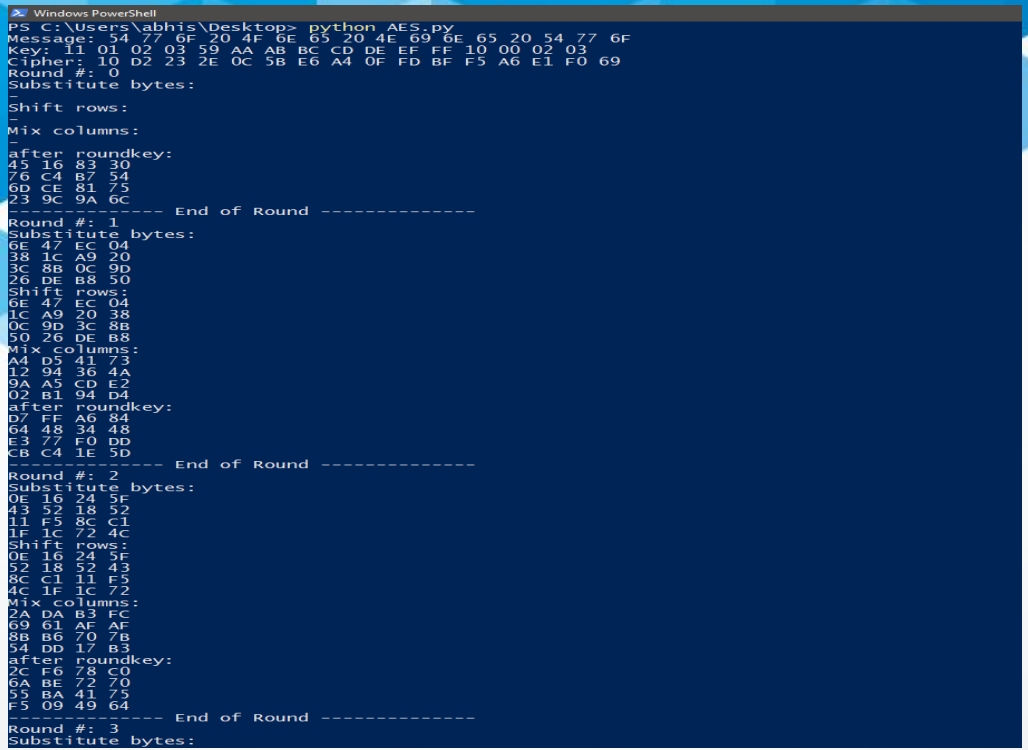


# Encryption And Decryption

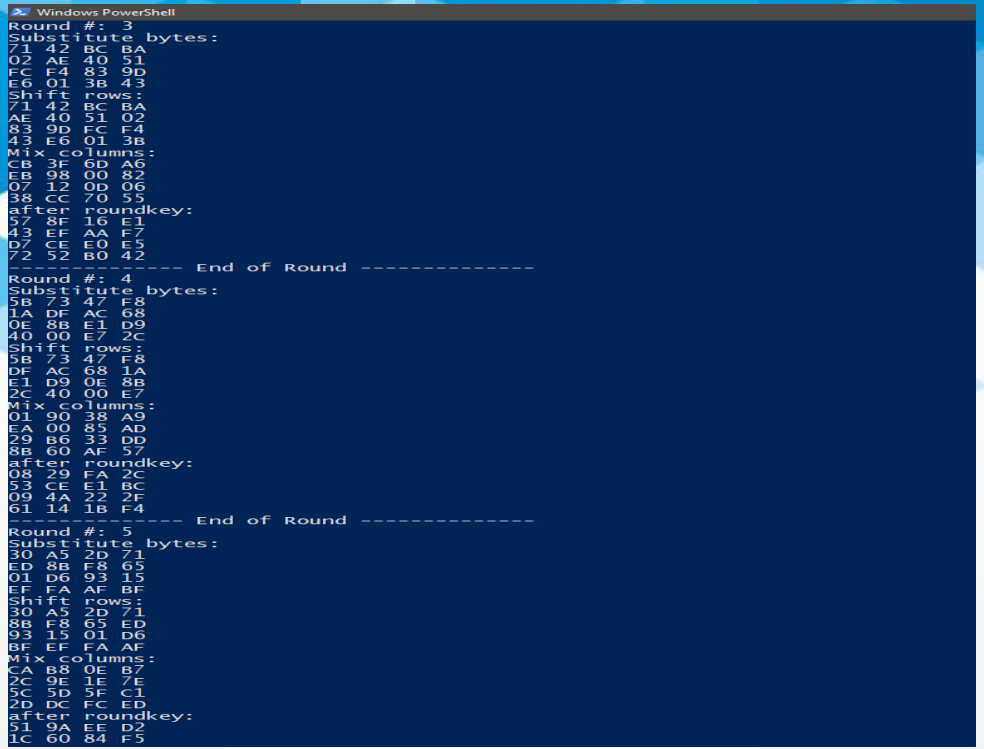
- For 1st step add round key.
- Substitute byte
- Shift Row
- Mix Column
- Add round key except for last step

- For 1st step add round key
- Inverse shift row
- Inverse substitute byte
- Add round key
- Inverse mix coulumn
- Get plaintext.

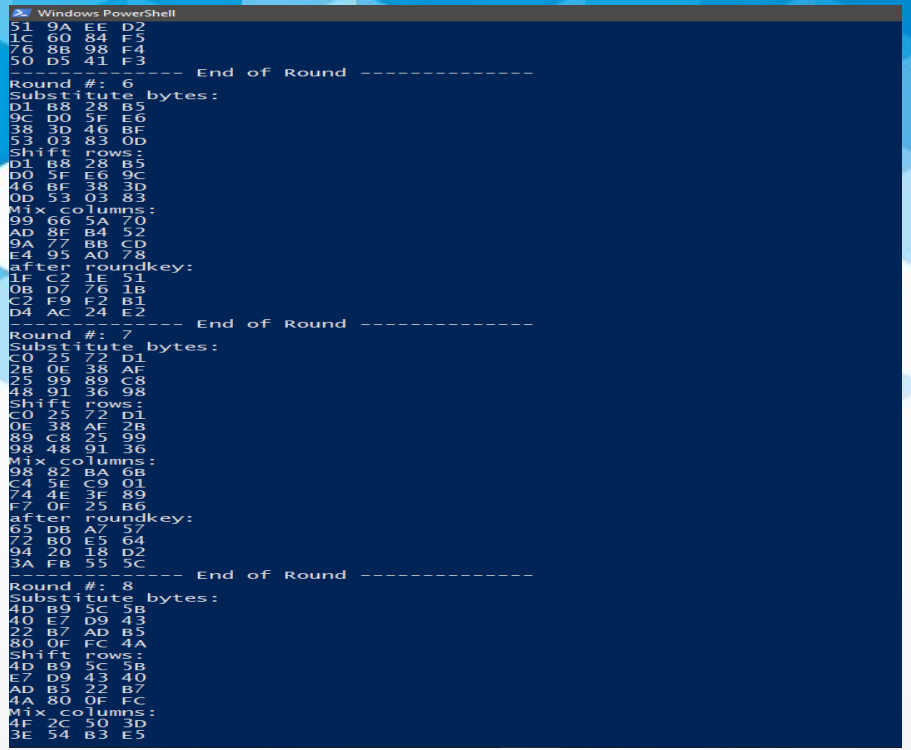
# Output (AES)


Windows PowerShell
PS C:\Users\abhis\Desktop> python AES.py
Message: 54 77 6F 20 4F 6E 65 20 4E 69 6E 65 20 54 77 6F
Key: 11 01 02 03 59 AA AB BC CD DE EF FF 10 00 02 03
Cipher: 10 D2 23 2E 0C 5B E6 A4 0F FD BF F5 A6 E1 F0 69
Round #: 0
Substitute bytes:
-
Shift rows:
-
Mix columns:
-
after roundkey:
45 16 83 30
76 C4 B7 54
6D CE 81 75
23 9C 9A 6C
---------- End of Round ----------
Round #: 1
Substitute bytes:
6E 47 EC 04
38 1C A9 20
3C 8B 0C 9D
26 DE B8 50
Shift rows:
6E 47 EC 04
1C A9 20 38
0C 9D 3C 8B
50 26 DE B8
Mix columns:
A4 D5 41 73
12 94 36 4A
9A A5 CD E2
02 B1 94 D4
after roundkey:
D7 FF A6 84
64 48 34 48
E3 77 F0 DD
CB C4 1E 5D
---------- End of Round ----------
Round #: 2
Substitute bytes:
0E 16 24 5F
43 52 18 52
11 F5 8C C1
1F 1C 72 4C
Shift rows:
0E 16 24 5F
52 18 52 43
8C C1 11 F5
4C 1F 1C 72
Mix columns:
2A DA B3 FC
69 61 AF AF
8B B6 70 7B
54 DD 17 B3
after roundkey:
2C F6 78 C0
6A BE 72 70
55 BA 41 75
F5 09 49 64
---------- End of Round ----------
Round #: 3
Substitute bytes:

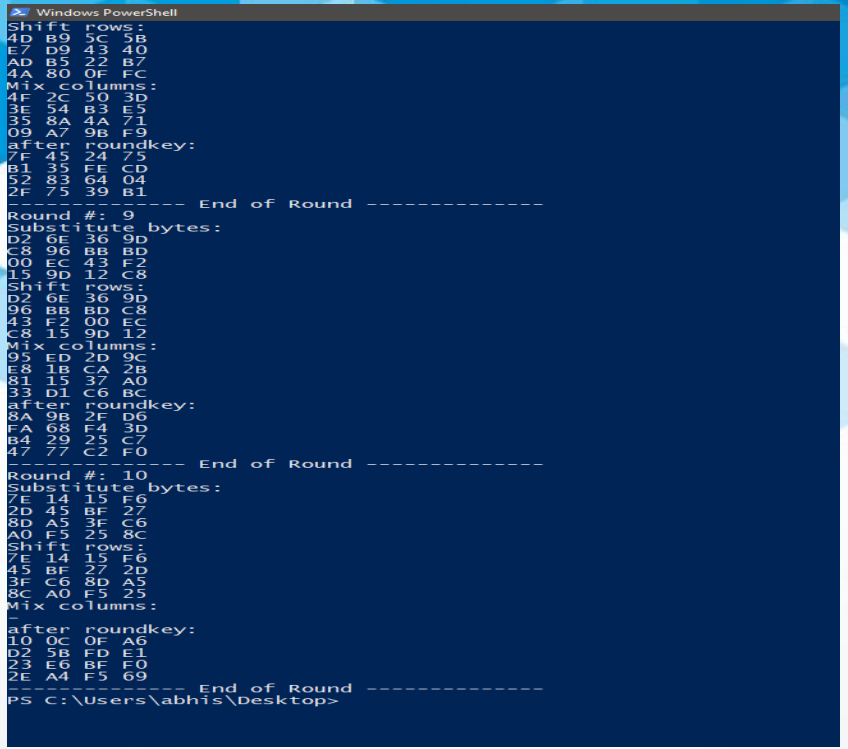
# Output (AES)


Windows PowerShell
Round #: 3
Substitute bytes:
71 42 BC BA
02 AE 40 51
FC F4 83 9D
E6 01 3B 43
Shift rows:
71 42 BC BA
AE 40 51 02
83 9D FC F4
43 E6 01 3B
Mix columns:
CB 3F 6D A6
EB 98 00 82
07 12 0D 06
38 CC 70 55
after roundkey:
57 8F 16 E1
43 EF AA F7
D7 CE E0 E5
72 52 B0 42
---------- End of Round ----------
Round #: 4
Substitute bytes:
5B 73 47 F8
1A DF AC 68
0E 88 E1 D9
40 00 E7 2C
Shift rows:
5B 73 47 F8
DF AC 68 1A
E1 D9 0E 8B
2C 40 00 E7
Mix columns:
01 90 38 A9
EA 00 85 AD
29 B6 33 DD
8B 60 AF 57
after roundkey:
08 29 FA 2C
53 CE E1 BC
09 4A 22 2F
61 14 1B F4
---------- End of Round ----------
Round #: 5
Substitute bytes:
30 A5 2D 71
ED 8B F8 65
01 D6 93 15
EF FA AF BF
Shift rows:
30 A5 2D 71
8B F8 65 ED
93 15 01 D6
BF EF FA AF
Mix columns:
CA B8 0E B7
2C 9E 1E 7E
5C 5D 5F C1
2D DC FC ED
after roundkey:
51 9A EE D2
1C 60 84 F5

# Output (AES)


Windows PowerShell
51 9A EE D2
1C 60 84 F5
76 8B 98 F4
50 D5 41 F3
---------- End of Round ----------
Round #: 6
Substitute bytes:
D1 B8 28 B5
9C D0 5F E6
38 3D 46 BF
53 03 83 0D
Shift rows:
D1 B8 28 B5
D0 5F E6 9C
46 BF 38 3D
0D 53 03 83
Mix columns:
99 66 5A 70
AD 8F B4 52
9A 77 BB CD
E4 95 A0 78
after roundkey:
1F C2 1E 51
0B D7 76 1B
C2 F9 F2 B1
D4 AC 24 E2
---------- End of Round ----------
Round #: 7
Substitute bytes:
C0 25 72 D1
2B 0E 38 AF
25 99 89 C8
48 91 36 98
Shift rows:
C0 25 72 D1
0E 38 AF 2B
89 C8 25 99
98 48 91 36
Mix columns:
98 82 BA 6B
C4 5E C9 01
74 4E 3F 89
F7 0F 25 B6
after roundkey:
65 DB A7 57
72 B0 E5 64
94 20 18 D2
3A FB 55 5C
---------- End of Round ----------
Round #: 8
Substitute bytes:
4D B9 5C 5B
40 E7 D9 43
22 B7 AD B5
80 0F FC 4A
Shift rows:
4D B9 5C 5B
E7 D9 43 40
AD B5 22 B7
4A 80 0F FC
Mix columns:
4F 2C 50 3D
3E 54 B3 E5

# Output (AES)


Windows PowerShell
Shift rows:
4D B9 5C 5B
E7 D9 43 40
AD B5 22 B7
4A 80 0F FC
Mix columns:
4F 2C 50 3D
3E 54 B3 E5
35 8A 4A 71
09 A7 9B F9
after roundkey:
7F 45 24 75
B1 35 FE CD
52 83 64 04
2F 75 39 B1
-------------- End of Round --------------
Round #: 9
Substitute bytes:
D2 6E 36 9D
C8 96 BB BD
00 EC 43 F2
15 9D 12 C8
Shift rows:
D2 6E 36 9D
96 BB BD C8
43 F2 00 EC
C8 15 9D 12
Mix columns:
95 ED 2D 9C
E8 1B CA 2B
81 15 37 A0
33 D1 C6 BC
after roundkey:
8A 9B 2F D6
FA 68 F4 3D
B4 29 25 C7
47 77 C2 F0
-------------- End of Round --------------
Round #: 10
Substitute bytes:
7E 14 15 F6
2D 45 BF 27
8D A5 3E C6
A0 F5 25 8C
Shift rows:
7E 14 15 F6
45 BF 27 2D
3F C6 8D A5
8C A0 F5 25
Mix columns:
-
after roundkey:
10 0C 0F A6
D2 5B FD E1
23 E6 BF F0
2E A4 F5 69
-------------- End of Round --------------
PS C:\Users\abhis\Desktop>

# References


- Python 2.7.15 Release: https://www.python.org/downloads/release/python-2715/
- Pillow and Pycrypto:

https://web.archive.org/web/20161201092248/http://www.voidspace.org.uk/downloads/pycrypto26/py crypto-2.6.win-amd64-py2.7.exe

https://web.archive.org/web/20161201092248/http://www.voidspace.org.uk/downloads/pycrypto26/py crypto-2.6.win32-py2.7.exe

- General Steganography:
  https://pdfs.semanticscholar.org/a85d/59cefb53375ad0e5e23df7b5a0431b2eefc5.pdf
  https://learncryptography.com/steganography/what-is-steganography
- Cryptography and network security by Bose and Vijaya Kumar.
- Cryptography and network security by William Stallings.