

APPLIED CRYPTOGRAPHY

NAME: VISHWAS M

SRN: PES2UG20CS390

SEC: F

LAB: 7

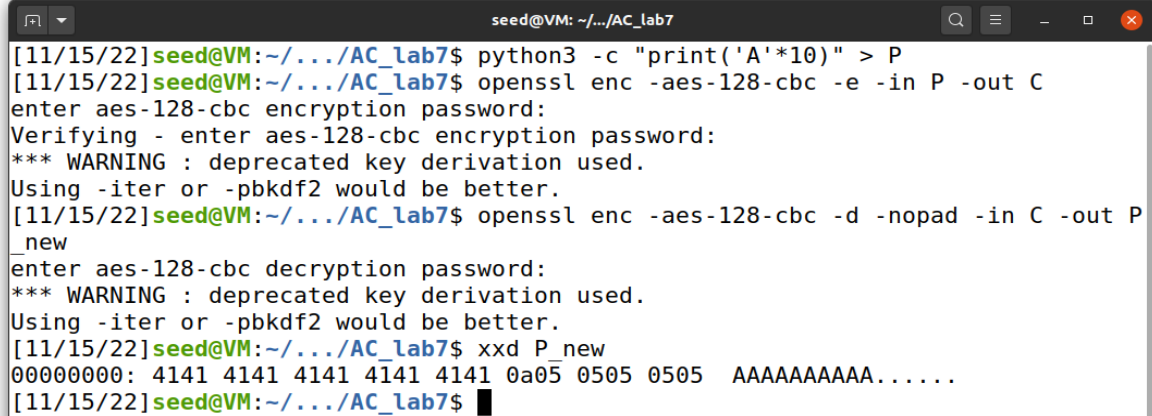
TASK 1: Getting Familiar with Padding

a) Length 5:



```
seed@VM: ~/.../AC_lab7
[11/15/22]seed@VM:~/.../AC_lab7$ python3 -c "print('A'*5)" > P
[11/15/22]seed@VM:~/.../AC_lab7$ openssl enc -aes-128-cbc -e -in P -out C
enter aes-128-cbc encryption password:
Verifying - enter aes-128-cbc encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
[11/15/22]seed@VM:~/.../AC_lab7$ openssl enc -aes-128-cbc -d -nopad -in C -out P_new
enter aes-128-cbc decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
[11/15/22]seed@VM:~/.../AC_lab7$ xxd P_new
00000000: 4141 4141 410a 0a0a 0a0a 0a0a 0a0a 0a0a  AAAAAA.....
[11/15/22]seed@VM:~/.../AC_lab7$
```

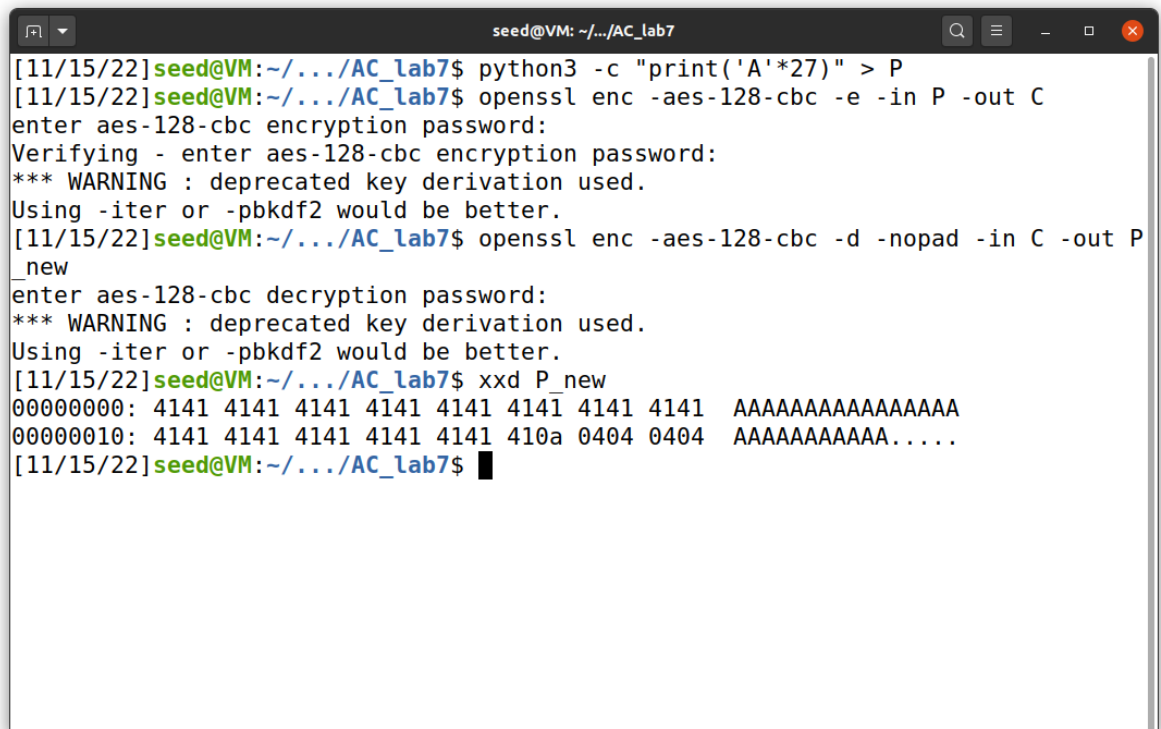
b) Length 10:



A terminal window titled 'seed@VM: ~/.../AC_lab7' showing the following commands and output:

```
[11/15/22]seed@VM:~/.../AC_lab7$ python3 -c "print('A'*10)" > P
[11/15/22]seed@VM:~/.../AC_lab7$ openssl enc -aes-128-cbc -e -in P -out C
enter aes-128-cbc encryption password:
Verifying - enter aes-128-cbc encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
[11/15/22]seed@VM:~/.../AC_lab7$ openssl enc -aes-128-cbc -d -nopad -in C -out P_new
enter aes-128-cbc decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
[11/15/22]seed@VM:~/.../AC_lab7$ xxd P_new
00000000: 4141 4141 4141 4141 4141 0a05 0505 0505  AAAAAAAAAA.....
[11/15/22]seed@VM:~/.../AC_lab7$
```

c) Length 27:



```
seed@VM: ~/.../AC_lab7
[11/15/22]seed@VM:~/.../AC_lab7$ python3 -c "print('A'*27)" > P
[11/15/22]seed@VM:~/.../AC_lab7$ openssl enc -aes-128-cbc -e -in P -out C
enter aes-128-cbc encryption password:
Verifying - enter aes-128-cbc encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
[11/15/22]seed@VM:~/.../AC_lab7$ openssl enc -aes-128-cbc -d -nopad -in C -out P_new
enter aes-128-cbc decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
[11/15/22]seed@VM:~/.../AC_lab7$ xxd P_new
00000000: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAAAA
00000010: 4141 4141 4141 4141 4141 410a 0404 0404  AAAAAAAAAAAAA....
[11/15/22]seed@VM:~/.../AC_lab7$
```

What do you deduce about the encryption scheme?

ANS: Padding is used to decrypt the ciphertext

Task 2: Padding Oracle Attack

Step 1 – Step 3 for 16 rounds

```
seed@VM: ~/.../Labsetup
[11/15/22]seed@VM:~/.../Labsetup$ cd ..
[11/15/22]seed@VM:~/.../AC_lab7$ nc 10.9.0.80 5000
01020304050607080102030405060708a9b2554b0944118061212098f2f238cd779ea0aae3d9d020f3677bfc3cda9ce
^C
[11/15/22]seed@VM:~/.../AC_lab7$ cd Labsetup/
[11/15/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xcf
CC1: 000000000000000000000000000000cf
P2: 00000000000000000000000000000000
[11/15/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x39
CC1: 000000000000000000000000000039cc
P2: 00000000000000000000000000000003
[11/15/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
P2: 00000000000000000000000000000303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
```

```
seed@VM: ~/.../Labsetup
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xf2
CC1: 00000000000000000000000000f238cd
P2: 00000000000000000000000000000303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x18
CC1: 0000000000000000000000000018f53fca
P2: 0000000000000000000000000000030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x40
CC1: 000000000000000000000000004019f43ecb
P2: 0000000000000000000000000000ee030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xea
CC1: 000000000000000000000000ea431af73dc8
P2: 000000000000000000000000ddee030303
```

```
seed@VM: ~/.../Labsetup
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x9d
CC1: 000000000000000009deb421bf63cc9
P2: 000000000000000000ccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xc3
CC1: 0000000000000000c392e44d14f933c6
P2: 0000000000000000bbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x01
CC1: 0000000000000001c293e54c15f832c7
P2: 0000000000000000aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x6c
CC1: 00000000000006c02c190e64f16fb31c4
P2: 0000000000000088aabbccdde030303
```

```
seed@VM: ~/.../Labsetup
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
P2: 00000000000006d88aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x29
CC1: 0000000000296d03c091e74e17fa30c5
P2: 0000000000007788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x50
CC1: 00000000502e6a04c796e04910fd37c2
P2: 0000000000667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x02
CC1: 00000002512f6b05c697e14811fc36c3
P2: 0000000055667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
```

```
seed@VM: ~/.../Labsetup
P2: 0000000055667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x68
CC1: 00006801522c6806c594e24b12ff35c0
P2: 0000004455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x9f
CC1: 009f6900532d6907c495e34a13fe34c1
P2: 0000334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
P2: 0020334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
P2: 0022334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
```

```
seed@VM: ~/.../Labsetup
Valid: i = 0x9f
CC1: 009f6900532d6907c495e34a13fe34c1
P2: 0000334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
P2: 0020334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
P2: 0022334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xa8
CC1: a880761f4c327618db8afc550ce12bde
P2: 0022334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xcf
CC1: a981771e4d337719da8bfd540de02acf
P2: 1122334455667788aabbccdde030303
[11/16/22]seed@VM:~/.../Labsetup$
```

CODE:

```
46 # The result is 0. This is not required for the attack.
47 # Its sole purpose is to make the printout look neat.
48 # In the experiment, we will iteratively replace these values.
49 D2 = bytearray(16)
50
51 D2[0] = 0xb8
52 D2[1] = 0x90
53 D2[2] = 0x66
54 D2[3] = 0xf
55 D2[4] = 0x5c
56 D2[5] = 0x22
57 D2[6] = 0x66
58 D2[7] = 0x8
59 D2[8] = 0xcb
60 D2[9] = 0x9a
61 D2[10] = 0xec
62 D2[11] = 0x45
63 D2[12] = 0x1c
64 D2[13] = 0xf1
65 D2[14] = 0x3b
66 D2[15] = 0xce
67 #####
68 # In the experiment, we need to iteratively modify CC1
69 # We will send this CC1 to the oracle, and see its response.
70 CC1 = bytearray(16)
71
```

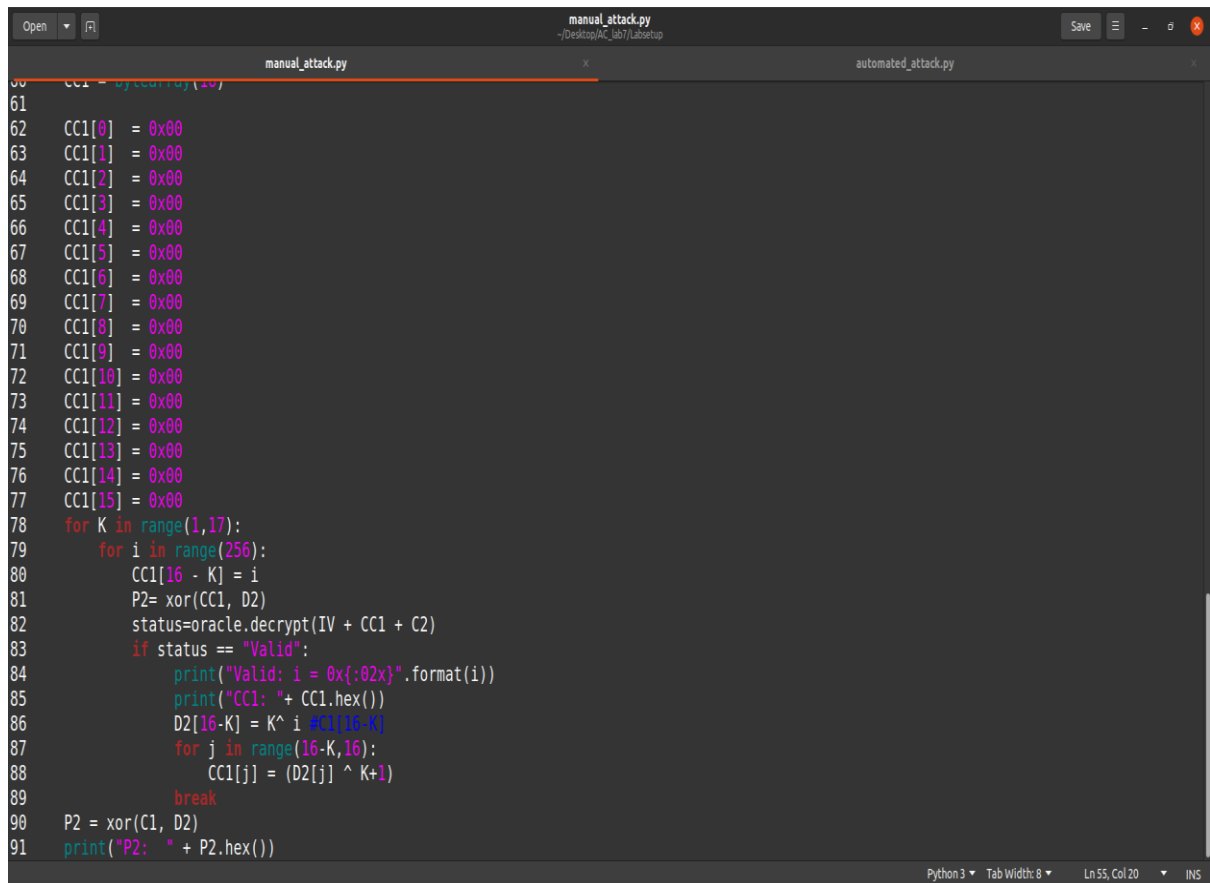
Python 3 Tab Width: 8 Ln 101, Col 31 INS

```
manual_attack.py
~/Desktop/AC_lab7/Labsetup

67 #####
68 # In the experiment, we need to iteratively modify CC1
69 # We will send this CC1 to the oracle, and see its response.
70 CC1 = bytearray(16)
71
72 CC1[0] = 0xa9
73 CC1[1] = 0x81
74 CC1[2] = 0x77
75 CC1[3] = 0x1e
76 CC1[4] = 0x4d
77 CC1[5] = 0x33
78 CC1[6] = 0x77
79 CC1[7] = 0x19
80 CC1[8] = 0xda
81 CC1[9] = 0x8b
82 CC1[10] = 0xfd
83 CC1[11] = 0x54
84 CC1[12] = 0xd
85 CC1[13] = 0xe0
86 CC1[14] = 0x2a
87 CC1[15] = 0xdf
88
89 #####
90 # In each iteration, we focus on one byte of CC1.
91 # We will try all 256 possible values, and send the constructed
92 # ciphertext CC1 + C2 (plus the IV) to the oracle, and see
```

Python 3 Tab Width: 8 Ln 101, Col 31 INS

Task 3: Padding Oracle Attack(level 2)



The image shows a code editor window with two tabs: 'manual_attack.py' and 'automated_attack.py'. The 'manual_attack.py' tab is active, displaying a Python script for a manual padding oracle attack. The script starts with a comment on line 60: `CC1 = oracle.decrypt(D2)`. Lines 62 through 77 initialize a list `CC1` of 16 elements, each set to `0x00`. A nested loop starting at line 78 iterates over `K` from 1 to 17 and `i` from 0 to 256. Inside the inner loop, `CC1[16 - K]` is set to `i`, `P2` is calculated as `xor(CC1, D2)`, and the oracle's `decrypt` method is called with `IV + CC1 + C2`. If the status is 'Valid', the script prints the valid `i` and `CC1` in hex, then calculates `D2[16-K]` as `K ^ i ^ CC1[16-K]`. A second inner loop then iterates over `j` from `16-K` to 16, setting `CC1[j]` to `(D2[j] ^ K+1)` and breaking the loop. After the loops, `P2` is calculated as `xor(C1, D2)` and printed. The status bar at the bottom indicates 'Python3', 'Tab Width: 8', 'Ln 55, Col 20', and 'INS'.

```
60 CC1 = oracle.decrypt(D2)
61
62 CC1[0] = 0x00
63 CC1[1] = 0x00
64 CC1[2] = 0x00
65 CC1[3] = 0x00
66 CC1[4] = 0x00
67 CC1[5] = 0x00
68 CC1[6] = 0x00
69 CC1[7] = 0x00
70 CC1[8] = 0x00
71 CC1[9] = 0x00
72 CC1[10] = 0x00
73 CC1[11] = 0x00
74 CC1[12] = 0x00
75 CC1[13] = 0x00
76 CC1[14] = 0x00
77 CC1[15] = 0x00
78 for K in range(1,17):
79     for i in range(256):
80         CC1[16 - K] = i
81         P2= xor(CC1, D2)
82         status=oracle.decrypt(IV + CC1 + C2)
83         if status == "Valid":
84             print("Valid: i = 0x{:02x}".format(i))
85             print("CC1: " + CC1.hex())
86             D2[16-K] = K^ i ^ CC1[16-K]
87             for j in range(16-K,16):
88                 CC1[j] = (D2[j] ^ K+1)
89             break
90 P2 = xor(C1, D2)
91 print("P2: " + P2.hex())
```

```
seed@VM: ~/.../Labsetup
[11/16/22]seed@VM:~/.../Labsetup$ python3 manual_attack.py
C1: ea36acb330cf727ebac12a84b37d114b
C2: 166b03a3e173a4a55d2b7b1d9fbf8370
Valid: i = 0x2f
CC1: 0000000000000000000000000000002f
Valid: i = 0x61
CC1: 000000000000000000000000000000612c
Valid: i = 0x19
CC1: 00000000000000000000000000000019602d
Valid: i = 0x97
CC1: 000000000000000000000000000000971e672a
Valid: i = 0xe4
CC1: 0000000000000000000000000000e4961f662b
Valid: i = 0x5e
CC1: 000000000000000000000000005ee7951c6528
Valid: i = 0xa7
CC1: 000000000000000000000000a75fe6941d6429
Valid: i = 0x92
CC1: 000000000000000000000092a850e99b126b26
Valid: i = 0x04
CC1: 0000000000000000000000493a951e89a136a27
Valid: i = 0x1a
CC1: 000000000000000000001a0790aa52eb99106924
Valid: i = 0xa5
```

```
seed@VM: ~/.../Labsetup
Valid: i = 0x5e
CC1: 0000000000000000000000005ee7951c6528
Valid: i = 0xa7
CC1: 0000000000000000000000a75fe6941d6429
Valid: i = 0x92
CC1: 000000000000000000000092a850e99b126b26
Valid: i = 0x04
CC1: 0000000000000000000000493a951e89a136a27
Valid: i = 0x1a
CC1: 000000000000000000001a0790aa52eb99106924
Valid: i = 0xa5
CC1: 000000000000a51b0691ab53ea98116825
Valid: i = 0x70
CC1: 0000000070a21c0196ac54ed9f166f22
Valid: i = 0x9e
CC1: 0000009e71a31d0097ad55ec9e176e23
Valid: i = 0xe6
CC1: 0000e69d72a01e0394ae56ef9d146d20
Valid: i = 0x7c
CC1: 007ce79c73a11f0295af57ee9c156c21
Valid: i = 0xbf
CC1: bf63f8836cbe001d8ab048f1830a733e
P2: 454544204c6162732061726520677265
[11/16/22]seed@VM:~/.../Labsetup$
```