



Faculty of Engineering and Technology

Electrical and Computer Engineering Department

INFORMATION SECURITY AND COMPUTER NETWORK

LABORATORY ENCS5121

Report II

Experiment # 2: Padding Oracle Attack

Student Name: Yara Darabumukho

Student Number: 1211269

Instructor: Dr. Ahmad Alsadeh

Teaching Assistant: Eng. Tariq Odeh

Section No: 1

Date: 30/10/2024

Abstract

In this experiment the main aim is to understand and apply the padding and learn about the Padding Oracle Attack.

Table of Contents

Abstract	I
Table of Figures	III
Table of Tables	III
Plaintext	1
CC1 Values	3

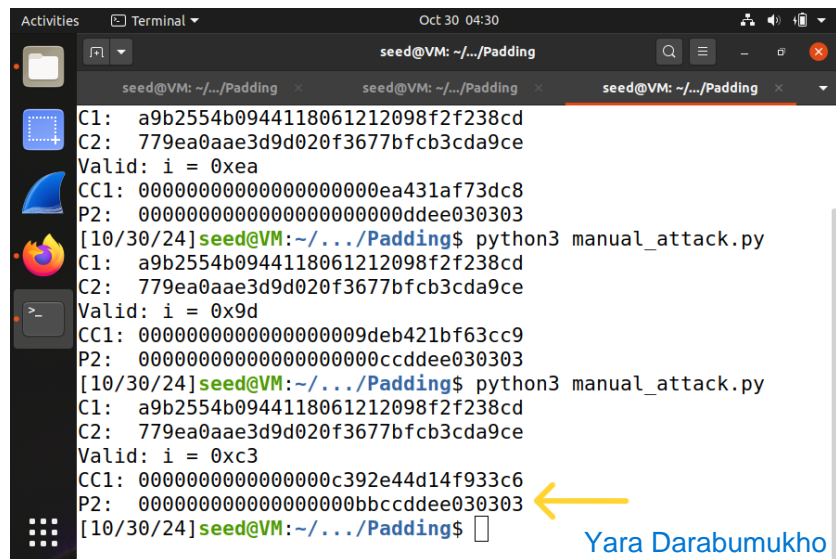
Table of Figures

Figure 1: Plaintext.	1
Figure 2: Block Cipher Values.	2
Figure 3: Valid Value.	2
Figure 4: CC1 values for the last iteration.	3

Table of Tables

Table 1: CC1 values.	3
---------------------------	---

Plaintext



```
seed@VM: ~/.../Padding
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xea
CC1: 000000000000000000000000ea431af73dc8
P2: 000000000000000000000000dde030303
[10/30/24]seed@VM:~/.../Padding$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x9d
CC1: 0000000000000000000000009deb421bf63cc9
P2: 000000000000000000000000ccdde030303
[10/30/24]seed@VM:~/.../Padding$ python3 manual_attack.py
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0xc3
CC1: 000000000000000000000000c392e44d14f933c6
P2: 000000000000000000000000bbccdde030303
[10/30/24]seed@VM:~/.../Padding$
```

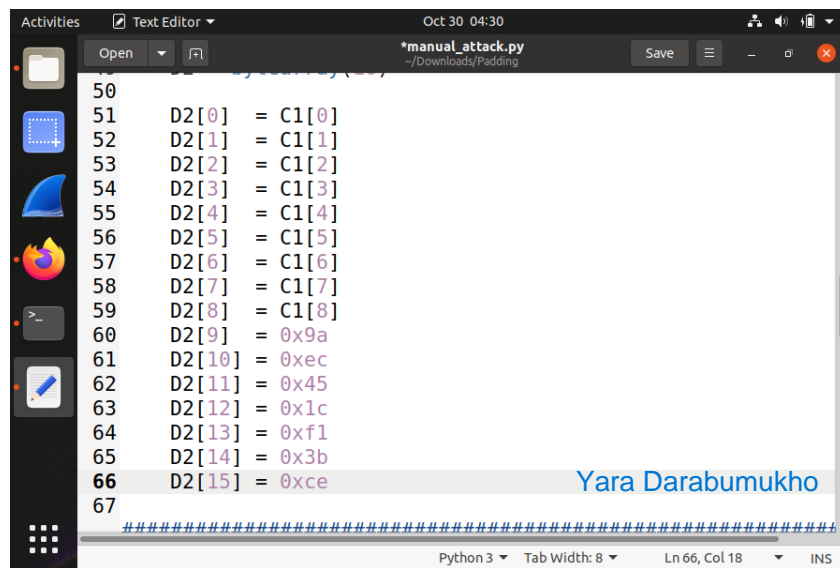
Figure 1: Plaintext.

As shown in the previous figure, the last 7-bytes of the plain test is:

0x*****bbccdde030303

From the plaintext we can notice that the padding done for 3-bytes only, that's mean the encrypted block contains 13-Bytes.

Block Cipher (D)



```
50
51 D2[0] = C1[0]
52 D2[1] = C1[1]
53 D2[2] = C1[2]
54 D2[3] = C1[3]
55 D2[4] = C1[4]
56 D2[5] = C1[5]
57 D2[6] = C1[6]
58 D2[7] = C1[7]
59 D2[8] = C1[8]
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
```

Figure 2: Block Cipher Values.

We get one value of the Block Cipher data in each iteration; after running the code we got a valid value, after that we just find the result of xoring this valid value with the iteration number “Key number”.

This is the valid value for Key equal to 7. “Seventh iteration”

```
C1: a9b2554b0944118061212098f2f238cd
C2: 779ea0aae3d9d020f3677bfc3cda9ce
Valid: i = 0x9d ← Yara Darabumukho
CC1: 0000000000000000000009deb421bf63cc9
P2: 000000000000000000000000ccdde030303
```

Figure 3: Valid Value.

- ⇒ Valid value from the last figure is 0x9d if u check the value of its xoring with the iteration number which is 7 u will get 0x9a. “The value of D2[9] in the Block Cipher Values figure”

CC1 Values

CC1 values in each iteration is the Block Cipher values xoring with the previous iteration number.

```

70 CC1 = bytearray(16)
71
72 CC1[0] = 0x00
73 CC1[1] = 0x00
74 CC1[2] = 0x00
75 CC1[3] = 0x00
76 CC1[4] = 0x00
77 CC1[5] = 0x00
78 CC1[6] = 0x00
79 CC1[7] = 0x00
80 CC1[8] = 0x00
81 CC1[9] = 0x92
82 CC1[10] = 0xe4
83 CC1[11] = 0x4d
84 CC1[12] = 0x14
85 CC1[13] = 0xf9
86 CC1[14] = 0x33
87 CC1[15] = 0xc6
88

```

Figure 4: CC1 values for the last iteration.

	K = 1	K = 2	K = 3	K = 4	K = 5	K = 6	K = 7	K = 8
CC1[0]	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
...
CC1[9]	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x92
CC1[10]	0x00	0x00	0x00	0x00	0x00	0x00	0xeb	0xe4
CC1[11]	0x00	0x00	0x00	0x00	0x00	0x43	0x42	0x4d
CC1[12]	0x00	0x00	0x00	0x00	0x19	0x1a	0x1b	0x14
CC1[13]	0x00	0x00	0x00	0xf5	0xf4	0xf7	0xf6	0xf9
CC1[14]	0x00	0x00	0x38	0x3f	0x3e	0x3d	0x3c	0x33
CC1[15]	0x00	0xcc	0xcd	0xca	0xcb	0xc8	0xc9	0xc6

Table 1: CC1 values.

Let's take K = 4 for example "Which is the Fourth iteration":

- ⇒ From the first Key we got D = 0xce, 0x3b from the second one, and 0xf1 from the third one.
- ⇒ To calculate the values of CC1 we just do xoring for the previous determined values in D with K = 4.