ECE 568 – Computer Security

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering

Mid-term Examination, Part 2, October 2019

Name	Solutions
Student #	

Answer all questions. Write your answers on the exam paper. Show your work.
Each question has a different assigned value, as indicated.

Permitted: one 8.5 x 11”, two-sided page of notes.
No other printed or written material. No calculator.
NO PHOTOCOPIED MATERIAL
Total time: 50 minutes
Total marks available: 50 (roughly one mark per minute, with some extra time)
Verify that your exam has all the pages.
Only exams written in ink will be eligible for re-marking.

3 /25	4 /25	Total

Question 3: Cryptography [25 marks]

In a padding oracle attack, explain what role (if any) each of the following play in the attack. Justify your answer [4 marks each]

a) Cipher-block chaining (CBC):

CBC allows a controlled change to the plain text of last block because it XOR's the 2nd last cipher text block with the output of the decryption of the last block to get the plaintext. This enables the padding oracle attack.

b) Advanced Encryption Standard (AES):

The padding oracle attack will work on any block cipher as all such ciphers need padding. It does not rely on AES being used.

c) Padding check:

The padding check must be performed and the results **must** **be reported** to the attacker for them to determine when they have forced the plain text padding to be valid.

d) Initial Vector (IV):

No role. The initial affects the plain text of the first block, while the padding oracle attacks the last block. The only time this would have an effect is if there is only one block, but it doesn't make sense to use CBC on one block.

e) Suppose we have a protocol that is vulnerable to a padding oracle attack. We alter the padding as follows:

- Instead of using the same value for the pad, we use a counter starting with 1 up to *n* where *n* is the number of bytes of pad. For example, if there are 5 bytes of pad, the last 5 bytes of the last block will be 1, 2, 3, 4, 5.

Everything else remains the same. Suppose the last 6 bytes of the last 2 blocks of a message are as follows:

Block C_n

0x39	0xa5	0x14	0x68	0xa1	0x85

Block C_{n-1}

0x46	0x90	0xa8	0xb4	0x37	0x16

Answer the following [3 marks each]

i. By changing the last byte of Block C_{n-1} to 0xaa results in no padding error. What can the attacker infer is the plain text value of the last byte of the plain text of Block C_n ? Explain your answer

In this padding scheme, a last byte of 0x01 is always valid padding (1 byte of pad). Thus, the last byte of the decryption of Block C_n must be 0xaa XOR 0x01 = 0xab. The original value of the last byte of C_{n-1} is 0x16, so the actual value of the last byte of P_n is 0xab XOR 0x16 = 0xbd *ii.* What should the attacker set the last byte of Block C_{n-1} to if she wants to decrypt the 2nd last byte of Block C_n? Explain your answer

She needs to force the last byte to be 0x02 for 2 bytes of pad and then try different 2nd last bytes for C_{n-1} until there is no padding error. To get the last byte to be 0x02, she needs to set the last byte of C_{n-1} to be 0xaa XOR 0x01 XOR 0x02 = 0xa9

iii. In the worst case, how many decryptions must the attacker ask the server to do in order to recover the entire last block if the block size is 128 bytes? How many times more or less effort is this than it would take to bruteforce the key if the key is 128 bits? Explain your answer

Attacker needs to brute-force each byte one at a time. Each byte has a maximum of 256 tries and there are 128 bytes to try so the maximum number of trials for the attacker is $256 * 128 = 2^8 * 2^7 = 2^{15}$

Bruteforcing a 128-bit key requires 2^{128} operations, so the attacker gets a speedup of $2^{128-15} = 2^{113}$ times less effort.

Question 4: Miscellaneous [25 marks]

You observe an attacker sending the following string to a program you wrote.

```
“\xa8\xe4\xff\xbfAAAA\xaa\xe4\xff\xbf%04x%04x%04x%04x%n%244u%n\%08x\
n”
```

You suspect that the attacker is exploiting a format string vulnerability to overwrite a pointer in your program. Your computer is running 32-bit code.

a) At what address does the attacker think the pointer is located? Give the address in hex and provide an explanation [4 marks]

The address is the byte sequence at the start of the string, which is 0xbfffe4a8.

b) What value is the attacker overwriting the pointer with? Give the value in hex and provide an explanation [4 marks]

The value written is the number of characters printed, which we must count

```
\xa8\xe4\xff\xbfAAAA\xaa\xe4\xff\xbf = 12
%04x%04x%04x%04x = 16
%n = 0
%244u = 244
%n = 0
\%08x\n = not relevant as it’s after the %n’s
```

The first %n writes 12+16 = 28 = 0x1c to 0xbfffe4a8
The second %n writes 12+16+244 = 0x110 to 0xbfffe4aa

Thus the value written is 0x0110001c

c) An attacker installs a key-logger on a victim’s computer and is able to capture the victim’s password. What aspect of the victim’s security has been compromised? Circle the appropriate answer. [2 marks]:

[✓ i) Confidentiality] ii) Integrity iii) Availability

d) Which attacks do Non-Executable pages prevent? Circle the appropriate answer: [3 marks/-1 per wrong answer]:

Return-into-libc	True	[✓ False]
Code injection	[✓ True]	False
Argument Overwrite	True	[✓ False]

e) When a vendor provides a receipt for a purchase to the buyer, this is to guarantee what for the purchase? Circle the appropriate answer. [2 marks]:
i) Authentication ii) Integrity [✓ iii) Non-repudiation]

f) What type of ciphers have the greatest encryption throughput? Circle the appropriate answer. [2 marks]:
i) Public-key Ciphers ii) Block Ciphers [✓ iii) Stream Ciphers]

g) Compute the following values using modular arithmetic in a finite field as defined by the indicated modulus [2 marks each]:

i) 4 + 10 mod 11

14 mod 11 = 3

ii) 9 * 5 mod 13

45 mod 13 = 6

iii) 7 / 3 mod 13

3*x mod 13 = 7, x=11

iv) log$_5$4 mod 7

5^x mod 7 = 4, x = 2