

Assignment 1: Exercises for Topic 1

Important Note. Due date: **May 25, 2020 (11:59am)**. The requested submission consists of the questions with * in the assignments (e.g., in this assignment, you will submit Exercises 2, 3, and 5). If you do those with **, then you receive a 2 bonus mark for each of those questions (Exercise 9 in this assignment).

Exercise 1. Consider basic definitions of cryptography and computer system security.

- (a) Provide the definitions of confidentiality (C), integrity and authentication (I), and availability (A) (shortened as CIA) of an information/computer system.
- (b) List at least three kinds of harm a company or personnel could encounter from a failure of CIA for each case of C, I and A.
- (c) Describe a situation in which you have experienced harm as a consequence of a failure of security services. Was the failure malicious or not? Did the attack target you specifically or was it general mass attack and you were the unfortunate victim?

Exercise 2*. Consider credit cards with chips implemented for authentication and encryption, in which users use their personal identification number (PIN) (usually 4 digits) as a key to do transactions (see the example in Lecture 3).

- (a) Give examples of confidentiality, integrity and availability associated with the systems.
- (b) Identify three different threats to each of the category which result in a loss of security of the system.
- (c) Identify three different methods that the attacker can reduce the search space for recover the PIN and estimate the complexity for each of them (you may exploit some weakness in implementation, program vulnerabilities, application scenarios, \dots , anything that can help you to reduce the complexity of the exhaustive search).

Exercise 3*. Consider the man-in-the-middle attacks in the following cases.

- (a) Identify two consequences when some entries of a data base have been modified.
- (b) Detail the entity authentication in the RFID system for contactless transaction, shown in the relay attack described in Lecture 3 (page 19).
- (c) For (b), provide three countermeasures for the relay attack.

Exercise 4. Consider the vulnerabilities of the following stack structure.

(a) Draw the function stack frame for the following C function.

```
int func(char *str)
{
    char buffer[24];
    strcpy(buffer, str);
    return 1;
}
```

(b) Discuss possible vulnerabilities of this program.

Exercise 5*. Program Vulnerabilities. For the following code, assume an attacker can control the value of `basket` passed into `search_basket`. The value of n is constrained to correctly reflect the number of dogs in the basket. The code includes several security vulnerabilities.

(a) Circle three such vulnerabilities in the code and briefly explain each of the three.

(b) Describe how an attacker could exploit these vulnerabilities to obtain a shell.

Some reminders:

- `snprintf(buf, len, fmt, ...)` works like `printf`, but instead writes to `buf`, and won't write more than `len - 1` characters. It terminates the characters written with a `'0'`.
- System runs the shell command given by its first argument.

```

1 struct dog {
2     char name[1024];
3     int age;
4 };
5
6 /*
7  Searches through a basket of dogs of size at most 32.
8  Returns the number of puppies or -1 if there are no puppies.
9  */
10
11 int search_basket( struct dog basket[], size_t n ) {
12     struct dog puppies[32];
13     char puppy_names[1024], cmd[1024];
14     int i , total_puppies = 0, name_size = 0;
15
16     if (n > 32) return -1;
17     for (i = 0; i <= n; i++){
18         if (basket[i].age < 12){
19             size_t len = strlen(basket[i].name);
20             snprintf(puppy_names + name_size, len, "%s", basket[i].name)
21             puppies[total_puppies] = basket[i];
22             name_size += len;
23             total_puppies += 1;
24         }
25     }
26
27     if (total_puppies > 5){
28         const char *fmt= "adopt - puppies -- num_puppies %d --names %s";
29         snprintf(cmd, sizeof cmd, fmt, total_puppies, puppy_names);
30         system(cmd);
31     }
32     return total_puppies;
33 }


```

Exercise 6. The spectre attacks.

- (a) Explain how much information of covert side channel employed in the spectre attacks.
- (b) Identify a case that the attacker can make the conditions listed in page 19 in Lecture 5 occurred.

Exercise 7. COVID-19 makes the remote meetings essential for conducting research, remote education, business, The following messages are received from our secretary of ECE. As you see from the email: a notification of the fake email, it asks an invitee to click a link for joining some on-line meeting, which was sent by an attacker.

- (a) How can you verify that the message did not come from the secretary without click that?
- (b) Provide one detailed exploit which can harm your computer/data.
- (c) Now play the role of an attacker. How could you intercept the message described in part (a) and convert it to your purposes while still making both the inviter and the client (the student) think the message is authentic and trustworthy?

[Faculty] Urgent - do not accept webex invites from me -- I did not se... [Details](#) 
To: department@ecemail.uwaterloo.ca



Hi everyone,
I have received a few emails from students saying they received a WebEx invite from me. When they click on the link, it says they are unauthorized⁴

Please note that these emails are **NOT** from me. Please do **NOT** click on any links.

If I send you a meeting request, we will have arranged it in advance. You know that I was sending it, and the subject line would specify the type of meeting.

Please see the following details:

Please be aware that now hackers have targeted remote meetings.

The massive increase in video conferencing lately has seen cyber criminals licking their lips on the newest opportunity to cause damage. They have jumped at the chance.

The largest targeted tool is Google Hangouts but Zoom, WebEx and Teams are not immune. How are they causing damage?

Through accepting fake invites.

Hackers are sending out fake meetings as calendar invites spoofing a legitimate contact within your list. For example, you will see a meeting invite from your manager which you accept for a later date. Accepting it is the first step to allow the exploit in. Then when the meeting starts, you click the link. The meeting looks like it begins but nothing happens. By then it is too late, the hacker is now in your computer causing damage or stealing key logging records such as bank account information and they cannot be traced or tracked.

Through Fake Collaboration.

You see a fake chat from a user with the same name as someone from your contact list. You begin to interact and that begins the exploit. During this time they are installing malicious software while you interact.

How can you tell?

Very subtle differences. When scrolling over the meeting link, you may miss little clues like a deliberate spelling error or an added character.

Some examples of fake domains being used are:

Googleclassroom/.com

[Goggleclassroom.com](#)

[Googieclassroom.com](#)

[Microsoftteams.com](#)

[Zoommmmeetings.com](#)

[Microsoftteams.com](#)

[Webbex.com](#)

Faculty mailing list

Faculty@ecemail.uwaterloo.ca

<http://ecemail.uwaterloo.ca/mailman/listinfo/faculty>

Exercise 8. For verification by a remote party, we have introduced two methods to validate a pair (pk_A, sk_A) of a public-key and its corresponding private key for authentication of the platform. One is through a certificate chain from the root public key on the platform, and the other is through an external certificate authority.

- (a) Can an attacker by pass this protection to install some software (it may not be malware)? Justify your answer.
- (b) It is possible to combine these remote attestation methods. One example is described as follows. Assume that a pair of platform specific keys (pk_C, sk_C) is installed on the platform such that pk_C is certified by the root public key. Use (pk_C, sk_C) to authenticate a newly generated pk_A when sending it to a certificate authority to obtain a certificate for pk_A . Describe the details for a certificate update protocol for pk_A .
- (c) Discuss the pros and cons for the above combined method. Identify at least two vulnerabilities for the combined method.

Exercise 9** The attacker intercepted a 16-bit ciphertext, given below

$$c = 0100011011000011$$

He has the knowledge of 1) the ciphertext is generated by a one-time-pad encryption, i.e., the ciphertext is the bitwise xor of a key stream and plaintext, both are 16-bits, 2) he knows that the plaintext is an English word using ASCII encoding (the right most bit is the least significant bit).

- (a) Find the plaintext in bits using a brute-force attack.
- (b) How many possible plaintext bit vectors are you getting when you conduct this attack? How do you filter out those incorrect ones and make the decision which one is correct? Justify your answer.
- (c) If attack has found the last 8 bits of the plaintext is 0110 1100, determine possible plaintext bit streams and the key stream used in each of those encryption.