Computer Security Exam

September 19, 2014

First name: _____ Last name: _____

Matricola: _____ Signature: _____

Homeworks? [ ] (Y) [ ] (N)

Instructions

- The exam is composed of 9 pages.
- Just as a cross check, tell us whether you have completed the homeworks, by putting an "X" mark appropriately.
- The exam is "closed books". Please put away in a non-suspicious place (i.e. not below the desk) any note, book, or similar. You will be expelled if, at any time, if you do not follow this rule.
- You are not allowed to communicate with other students, and you will be expelled from the exam if you do.
- Shut down and store electronic devices. They will be subject to inspection if found and you may be expelled if you are found using one.
- Please answer within the allowed space. Schemes are good, short answers are recommended.
- You can write in pen or pencil, any color, but avoid writing in red.
- No extra paper is allowed.
- The answers must be written exclusively in the space provided below the questions.

PROPOSED SOLUTION

Question 1 (2 points)

Consider the following web page, containing a simple form with an input field:

```
<body>
    <form action="/handler.do" method="POST">
          <input type="text" name="message" />
    </form>
</body>
```

The POST requests are transmitted over HTTPS to the server, which handles them with the following script:

```
string_t msg = request.post['message']  //read 'message' from request

if msg is not empty:
   database.save(msg)                          //save message in database

response.write("<p>Welcome! " + message[1] + "</p>")
```

where the **response.write** function writes the formatted string in the HTTP response.

A. Explain which is the vulnerability that you see and describe how it works.

There is a reflected cross-site scripting (XSS) vulnerability. It creates a per-request dataflow that originates from the client's rendered page (e.g., input field or other HTTP variable), ending up on the client's rendered page (e.g., interpreted HTML or JS content) as a result of the server response. The message, which could contain malicious code, is also stored in the database, and may be reused afterwards.

B. Consider the database administrator, who can modify only the database (e.g., no modifications to the above code are allowed). Is there any countermeasure that (s)he can adopt? Discuss your answer.

[1] ERRATA CORRIGE: this is msg.

The database receives just queries. In this case the query should be something like INSERT INTO Messages VALUES('<script>alert("XSS")</script'). Essentially, the value being stored is just a string. The point of view of the database does not allow to prevent this vulnerability because there is no context to determine the provenance of the string. The web application developer must fix this, not the database administrator.

Question 2 (4 points)

Consider the phenomenon of identity stealing in social networks (e.g., Facebook, Twitter, Google+), which happens when a cyber criminal steals the username and password of a user and uses them to impersonate that user (e.g., post content, send messages to friends, etc., without the user's consent).

A. What is the risk component in this scenario?

The risk is that the victim's identity could be used to negatively affect the reputation of the user. Another risk is that the stolen account is used to post malicious content (e.g., links to malicious sites), which is spread among the victim's friends. Further exacerbating the risk is the fact that the user may be using the same password for multiple websites.

B. What are the assets?

In the risk scenarios described in the previous answer, one asset is the victim's reputation, another asset is the victim's friends computers.

C. What is the threat?

The threat is a cyber criminal motivated either by hatred against the victim, or by the possibility of abusing the victim's credibility to spread malicious content.

D. What is the vulnerability that allows this scenario, and how would you mitigate it?

There is no strong authentication mechanism: if the credentials get stolen, there is no assurance on who is using them. A viable solution would be to use a second factor of authentication, such as a token sent via SMS.

Question 3 (5 points)

Answer the following questions:

A. A computer connected to a wireless network with no encryption is exposed to a high degree of risk. True or false? Discuss your answer in both cases to get any point.

False. The risk associated to a system, including a wireless network, depends from many factors. For example, an open network could be perfectly secure if there is the asset "computer" has zero value.

B. Explain how the dual signature mechanism of SET works.

C. The one time pad is a theoretical description of the perfect cipher, but cannot be implemented. True or false? Discuss your answer in both cases to get any point.

False, a one-time pad can be implemented by generating a key "as long as the message", for instance by pre-exchanging a very large random keystream and using it for short, high-relevance messages. However, this is not practical in almost any real world case.

Question 4 (5 points)

A web page contains the following code:

```
//get the username from the HTTP request
var username = request.get['username'];
```

```
query = "SELECT salary FROM users, role WHERE name = '" + username + "' AND id = user_id "
```

```
res = db.execute(query);
for record in res:
   response.writeline("<p>" + record.toString() + "</p>")
```

These are tables and data of the database used by the application:

users

id	name	gender	birthyear
1	Adams	M	1970
2	Baker	M	1992
3	Marco	M	1953
4	Dood	F	1972

role

user_id	role	salary
1	Prof	80
2	Student	20
3	Prof	100
4	Admin	70

A. Complete the code above by initializing the **`query`** string variable with a proper query to retrieve the salary of the user, whose name is stored in the username variable.

B. Write the pseudo code or explain how the query should be processed before passing it to **`db.execute()`**. The resulting code should not introduce any vulnerability.

Escape the ' character.

"SELECT Salary FROM users, role WHERE role.user_id = users.id AND username = ?". Then I will process the query using a prepared statement, or equivalent function. Alternatively, instead of prepared statements, the username string should be escaped before concatenating it

to the query, and the query could be "SELECT Salary FROM users, role WHERE role.user_id = users.id AND username = '" + escape(username) + "'".

C. Now write a vulnerable query and an exploit to demonstrate how its vulnerability could be used to read the salary of any user.

*Query: "SELECT * FROM users JOIN role ON (id = user_id) WHERE username = '" + username + "'" (no escape)*
Exploit: username = ' OR 1=1--

Question 5 (6 points)

Consider the following snippet of code, which prints the word "**secre**" on the standard output.

```
#include <stdio.h>
char *data;
int c = 0;

void output() {
    int a = 0;
    scanf("%d", &a);
    if(a > 5) {
       printf("Max data exceeded\n");
       a = 0;
       return;
    }
    c += a;
    for(a = 0; a < c; a++)
       printf("%c", data+a);
}
```

```
int main () {
    char *sens = "secre";
    data = sens;
    int choice = 0;
    while (1) {
        printf("Choice: ");
        scanf("%d", &choice);

        if (choice == 0)
            output();

        return;
    }
}
```

There is one vulnerability.

A. Indicate the line(s) of code affected and explain why the code is vulnerable

[2] ERRATA CORRIGE: %d - does not change anything in the solution: it's an obvious typo.
[3] ERRATA CORRIGE: this is a copy-paste error from the previous exam. If this return is considered, then there is no vulnerability. However, we accepted both answers if properly discussed.

It's a logic vulnerability. The output() function is executed as long as 0 is submitted at each iteration of the while(1) loop. Even if the a variable is reset in case it is above 5, it is used to increment c in c += a, so c is increased 4 units per loop. The for() loop is controlled by the value of c, and with the printf() allows the attacker to read addresses from the stack via the pointer data+a.

B. Briefly describe how you would fix the security bug you identified.

The access to memory via the data pointer should be bounded. One solution would be to pre-allocate a given amount of memory, making data an array, and then controlling the bounds in the for() loop.

Question 6 (4 points)

A. Explain what a rootkit is and how it infects a system.

A rootkit is a malware that infects the machine at user or kernel level. The goal of a rootkit is to give the attacker persistence access to the machine and, optionally, hide the attacker's actions (including the presence of the rootkit itself) by modifying how the (operating) system behaves (e.g., by hiding processes or files).

B. You are analyzing a computer that behaves abnormally. You suspect that it has been infected by a rootkit. How would you proceed to check if there is indeed a rootkit infection?

Rootkits are hard to detect, especially if they are planted in the operating system kernel, because they alter its behavior to make detection difficult or impossible. Usually, a reasonable first approach is to analyze the computer's disk on a different computer, and try to spot any differences (e.g. any files or processes that show up on the other computer and not on the supposedly infected one). This is called cross-layer examination.

Question 7 (4 points)

A. WPA2 is the evolution of WPA because it introduces an authentication mechanism, 802.11X, which was not present in WPA. True or false? Discuss your answer to get any point.

False. WPA2 is the evolution of WPA, but WPA already had an authentication mechanism based on 802.1X. WPA2 introduced PEAP, which is based on CHAPv2, while WPA had LEAP/CHAPv1.

B. The vulnerability of LEAP/CHAPv1 is due to the fact that it uses a weak initialization vector. True or false? Discuss your answer to get any point.

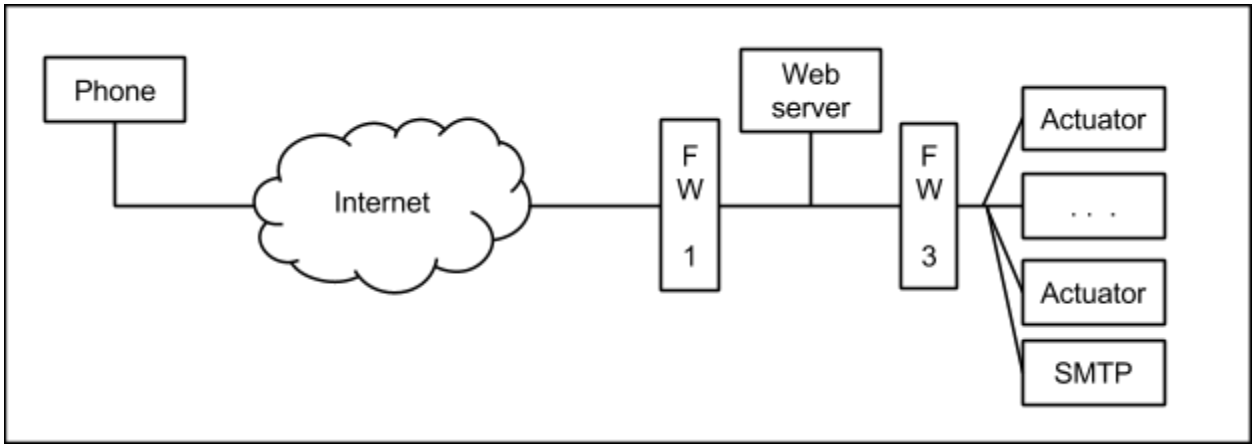
False. CHAPv1 is vulnerable because it is based on a weak hash, which is further weakened by adding 5 null bytes and splitting it into 3 subkeys, then used as DES keys.

Question 8 (3 points)

The OpenRemote system allows to control your home appliances (e.g., air conditioner, washing machine, refrigerator, lights, door locks, gates) from your smartphone over the Internet. For example, you can turn on the air conditioner from the office or open the gate when you're approaching your house. The system is composed of a web server, which is connected to actuators (e.g., door lock, light on-off switch) over a local network. You can assume that the actuators are TCP/IP-based devices. You send commands to the web server over HTTP via their Internet-connected smartphones. The web server interprets each

command received and activates or de-activate the actuators. The web server uses a local email server to send daily reports to the home owner about the conditions of the house (e.g., temperature).

A. Draw a network layout.



B. (1) Identify the most valuable asset and (2) describe a risk scenario against that asset, clarifying the (3) threats and the (4) vulnerabilities that cause it.

The most valuable asset is the actuator of the door lock or gates. In an hypothetical risk scenario an attacker can open the door and break into the house. Specifically, a man in the middle attacker (threat agent) can read the commands and authentication session over the Internet. The vulnerability that cause this attack could be that the network traffic is not encrypted.

C. Propose any network-level security mechanism or protocol to ensure that the above risk scenario is properly mitigated.

We propose to use HTTPS or a VPN between the smartphone and the web server.