Computer Security Exam

Professors F. Maggi & S. Zanero

Milan, 05/02/2018

Last (family) Name _____

First (given) Name _____

Matricola or Codice Persona _____

Have you done any challenges/homework, even partially? [ ] Yes [ ] No

Professor [ ] Maggi [ ] Zanero

Instructions

- The exam is composed of 12 pages. Check that you have all of them
- 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 notebook, 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 should be written exclusively in the space provided below the questions.

READ CAREFULLY ALL THE POINTS OF EACH QUESTION BEFORE WRITING YOUR ANSWER

SOLUTION

Answer provided in this solution MUST BE CONSIDERED ONLY AS A HINT for the correct answer, and they are not necessarily complete.

Question 1 (10 points)

Consider the C program below, which runs on the usual IA-32 architecture (32 bits), with the usual "`cdecl`" calling convention.

```
 1      #include <stdio.h>
 2      #include <stdlib.h>
 3      #include <string.h>
 4      #include <fcntl.h>
 5      #include <unistd.h>
 6
 7      int login() {
 8          struct {
 9            int saved_pidgeon;
 9            char buf[16];
10            int pidgeon;
11            int guess;
13            int unused_variable;
14          } s;
15
16          s.pidgeon = 0x41424344;
17          s.saved_pidgeon = s.pidgeon;
18
19          scanf("%s", s.buf);
20
21          s.guess = atoi(s.buf) /* convert string to int */
21          s.guess = *(int *)s.buf
22
23          if(s.saved_pidgeon != s.pidgeon) {
24             abort(); /* Error: the pidgeon has died !!!*/
25          } else if(s.guess != s.pidgeon) {
26             abort(); /* Error: I do not recognize this pidgeon! */
27          } else {
28             /* Error: O-ho O-oh O-oh*/
28             return 1;
29          }
30         return 0;
31      }
32
33      int main(int argc, char** argv) {
34          login();
35      }
```

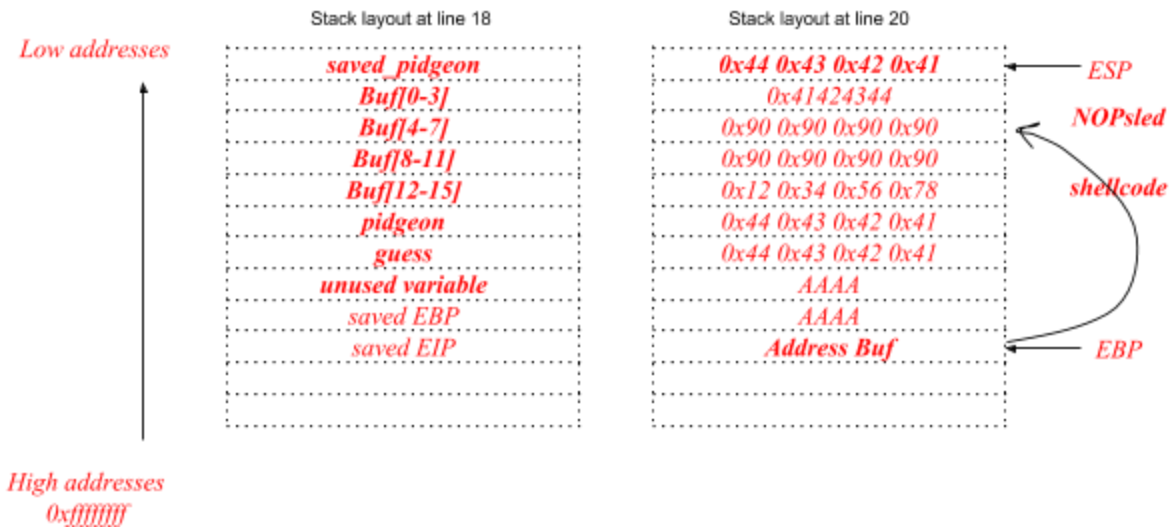
1. [4 points] Assume that the program is compiled and run with no mitigation against exploitation (**no canary, executable stack**, environment **with no ASLR** active).
The program **is affected by a typical buffer overflow vulnerability**. Write an exploit for this vulnerability to **obtain a shell**. For this purpose, assume that the following shellcode, composed by 4 bytes of instructions, will spawn a shell: `0x12 0x34 0x56 0x78`.

Write the exploit clearly, detail all the steps and assumptions you need for a successful exploitation, and draw the stack layout right before and after the execution of the `scanf` (line 19) during the program exploitation showing:

a. Direction of growth and high-low addresses;
b. The name of each allocated variable (and values if provided);
c. The boundaries of frame of the function frames (`main` and `login`).

Show also the content of the caller's frame (you can ignore the environment variables, just focus on what matters for the vulnerability and its exploitation).

……..



2. [3 points] Explain the security mechanism implemented in the C program presented above, why it does not work, and provide a possible fix.

Static Canary. In this case is a static value (0x41424344): if the binary is available, it is enough to retrieve this value by reverse engineering the program; then, during the exploitation it is enough to overwrite the stack canary with this (known) value. To fix this issue, the stack canary should be randomized at the program startup and placed in a register.

4. [2 points] Consider now the scenario in which various perfectly implemented security mechanisms can be activated to avoid exploitation. Complete the following table in which the only mechanisms enabled are the ones specified in that row.

Security mechanism enabled	How it works ?	Is the exploit you wrote at Question 1.1 still working ? Why ?
ASLR	See slides ...	No, ...
Canary	See slides ...	No, ..

4. [1 points] Suppose that now only the "not executable stack" is active as a security mechanism to avoid exploitation. Is the exploit you wrote at **Question 1.1** still working ? Why? If no, provide a suitable modification to the exploit that allows an attacker to exploit the vulnerability.

No, ...
Ret to lib c ...

Question 2 (7 points)

A web application contains four pages to handle login, registration, post comments, and read comments, all served over a secure HTTPS connection.

Here you can find code snippet of these pages:

```
// Show comments
var id = request.get['id'];
var prep_query = prepared_statement("SELECT username FROM users WHERE id=? LIMIT 1")
var username = query(prep_query, id);

var comments = query("SELECT * FROM comments WHERE username='"+ username +"'");
for comment in comments{
   echo comment;
}


// Login
var password = md5(request.post['password']);
var username = request.post['username'];
var prep_query = prepared_statement("SELECT username FROM users WHERE username=? AND
password=? LIMIT 1")
var username = query(prep_query, username, password);
if (username){
   $_Session['username'] = username;
   echo "Logged in.";
}


// Registration
var password = md5(request.post['password']);
var username = request.post['username'];
var prep_query = prepared_statement("INSERT INTO users (username, password) VALUES (?,?)")
query(prep_query, username, password);


// Write Comment
var username = $_Session['username']; //You need to be logged in
var comment = request.get['comment'];

var prep_query = prepared_statement("INSERT INTO comments (username, comment, timestamp)
VALUES (?,?, NOW())")
query(prep_query, username, comment);
```

As it is clear from the code, this application uses a database to store data. These are tables and data of the database:

users

id	*name*	*password*
1	admin	d1e576b71ccef5978d
2	John	563c39089151f9df26
3	Shamano	76ceaaa34826979e77

comments

id	*username*	*comment*	*timestamp*
1	admin	Welcome to my board.	03:23:21 24/12/2000
2	Shamano	Bomber scored 9 goals today.	18:31:62 17/02/2015

2. [3 points] Only considering code snippet of all the pages**,** identify which of the following web application vulnerabilities are present:

Vulnerability class	*Is there a vulnerability belonging to this class in the code? If so, explain why it is present and specify how an adversary could exploit it.*	***If the vulnerability is present in the code above**, explain the simplest procedure to remove this vulnerability.*
Stored cross-site scripting (XSS)	Yes	The simplest procedure to prevent this vulnerability is ...
Reflected cross-site scripting (XSS)	No, ...	The simplest procedure to prevent this vulnerability is ...
Cross site request forgery (CSRF)	Yes, ...	The simplest procedure to prevent this vulnerability is ...

2. [2 points] The web application is vulnerable to Sql injection, Write down an exploit, to get the hash of the password of **admin**. You must also specify all the steps and assumptions.

First you need to register a user which username contains the injection, then you need to visit comments page of this user.
username= ' UNION ALL SELECT id, name, password, NOW() from users; -

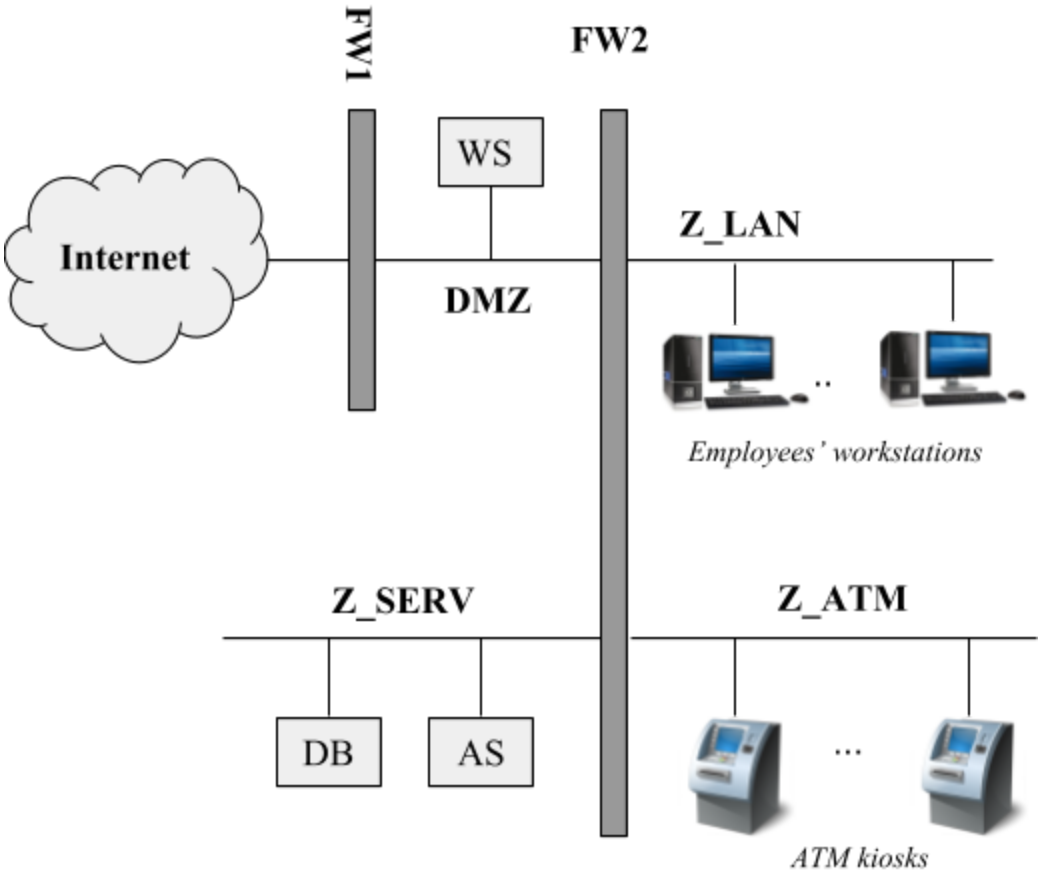
3. [2 points] How can you obtain the cleartext of the password from the hash obtained ? Provide the simplest attack and its mitigation.

You can build (or use one already avaiable) rainbow table to de-hash the password performing a bruteforce attack. Then, you own both username and password to log in.
To avoid hash-password cracking, all the passwords should be hashed with a salt.

Question 3 (10 points)

A small bank is in the process of setting up the network of their only, very small, branch.
The branch employees, from their **desktop computers**, need to access the **Internet** for work purposes (e.g., accessing their web-based email) as well as use an **internal web application**, served from the branch web server over the **HTTP protocol**. The web server also hosts the customer-facing **online banking application**, available over the Internet and served over the **HTTPS protocol**. Furthermore, the web server is backed by (i.e., communicates with) an **application server**, which stores its data on a **relational database server**. As the information processed by the application server and stored in the database server is sensitive, there is a strong requirement to prevent the employees from directly accessing those servers.
Besides the employee computers, the branch has some **ATM kiosks** that allow self-service cash withdrawals and account balance inquiries. To process those transactions, ATMs communicate with the application server over a proprietary protocol. The ATMs do not have access to either the Internet or any other network.

The layout of this network is the following:



1. [3 points] Write the firewall rules, assuming firewalls to be stateful packet filters (i.e. you can consider the response rules implicit)

Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description
FW1 (example)	*10.0.0.1 (example)*	*ANY*	*zone 1 -> zone 2*	*192.168.0.2 (example)*	*443*	*DENY*	*(example: the X server in zone 1 cannot contact the Y server)*
FW1, FW2,	ANY	ANY	ANY	ANY	ANY	DENY	DENY ALL
FW1	ANY	ANT	Internet → DMZ	WS_IP	443	ALLOW	Online banking access
FW1	Any IP in Z_LAN	ANY	DMZ → Internet	ANY	80, 443	ALLOW	Internet access for employees
FW2	Any IP in Z_LAN	ANY	Z_LAN → DMZ	ANY	80, 443	ALLOW	Internet access for employees + internal webapp access
FW2	WS_IP	ANY	DMZ → Z_SERV	AS_IP	AS_PORT	ALLOW	Web server → application server
FW2	Any IP in Z_ATM	ANY	Z_ATM → Z_SERV	AS_IP	AS_PORT	ALLOW	Web server → application server

2. [1 point] Let's consider a more realistic scenario: the bank is now part of a larger banking group. While keeping its own web server locally, the application server and database server are now shared among various branches and kept in a central location, where they need to be made remotely accessible from each branch (and from the bank branches only).
How would you securely realize this architecture? Please state your assumption and detail any changes to the network diagram for this scenario.

The AS and DB are now in a remote location. As we don't want to expose them over the Internet, we need to set up a VPN between our network and the central branch. Basically we can accomplish this by setting up a VPN between the remote location and placing the VPN client in Z_SERV to bridge the Z_SERV network with the remote network (or assuming to set up a firewall-to-firewall VPN with the appropriate policies). As the overall network structure is unchanged, except for the VPN tunnel, the firewall policies would be the same.

3. [1 point] The bank is worried that, as employees have full access to the Internet, their computer could become infected with malware. Thus, he decides to install a system to analyze the content of any HTTP response and scan it with an anti-virus for the presence of known malware. Assume we're interested in filtering traffic to HTTP pages only. What kind of packet filter should the bank put between the employees' LAN and the Internet zone? Why?

As we need to analyze the content, we need an application proxy (an HTTP proxy in particular).

4. [1 point] Is it possible to reach the same goal if the pages are served via HTTPS? How?

…… HTTPS man in the middle ….. Trusted cert on employees computers …..

5. [4 points] Whoops! You discover that a network expert customer was able to enter the branch, locate a spare network outlet connected to the employees network (Z_LAN), connect his\her laptop and *intercept all the HTTP communication between the a bank employee and the branch web server exploiting the gateway*, with terrible consequences! Now the bank CISO wants a detailed report on what could have happened and on how we could prevent this to ever happen in the future. Please answer the following questions:

(a) Can you guess a technique that the customer could have used to reach this goal? State the name and briefly describe how it works *in general*.

ARP spoofing, see slides

(b) Detail all the steps that the customer could have performed in order to intercept the communication between a bank employee's computer and the web server *in this specific scenario*.

The customer uses ARP spoofing to pose as the network gateway and sniff all the communication between the employee's computer and the gateway, including the traffic to the WS.
1. The customer learns the IP address (e.g., by obtaining it via DHCP, or, if DHCP is not enabled, by passively sniffing the network broadcast traffic) and the real MAC address of the gateway (via ARP);
2. The customer broadcasts ARP messages with the gateway IP address and the customer's own MAC address;

3. If the spoofing succeeds, the traffic to the gateway is directed to the customer (the customer also forwards traffic to the real gateway). This, way, the customer is able to sniff all the information between the client and the gateway and, thus, between the client and the local web server.

(c) Describe a possible way for the branch to prevent, or mitigate, this type of attack *in this specific scenario, besides disabling/locking/damaging the* spare network outlet found by the customer.

From the application point of view: use HTTPS with a certificate trusted by the browsers of the employee computers (BONUS: in this case it is important also to enable HSTS to prevent the customer to try to downgrade the communication to unencrypted HTTP or to train the employees to always check whether the communication is encrypted). From the network point of view: various techniques; for example, 802.1x to authenticate clients connected to ethernet ports or attempt, ... (in general this approach is complementary to the use of HTTPS).

(d) Can the same attack be used to intercept communication between the web server and the application server?

No, as they are on different networks.

Question 4 (5 points)

1. [1 point] Explain what is a rootkit

See slides

2. [1 point] Explain the difference between a user-land and kernel-land rootkit

See slides.

3. [2 point] You suspect that your machine have been compromised with a kernel rootkit. You tried to use network traffic tools from your machine but you do not see any malicious traffic. Can you conclude that your machine is safe? If is not there are other way to prove you have been compromised?

No you cannot conclude that the machine have not been compromised. Because the malware can hide its own traffic from tools running on the compromised machine. You could inspect network traffic using an external machine as a MitM between your machine and the router.

4. [1 point] A colleague suggests to replace the hard drive of a machine to be sure to get rid of a very sophisticated rootkit. However, after reinstalling the operating system, it seems like that the machine is infected by the same rootkit. Provide an explanation of what happened. Whatever your answer is, explain why.

If it is a BIOS rootkit then No. If it is a kernel rootkin it is ok to just replace the HD or even just reinstall the OS.