Jashandeep Justin Bains
J24bains
20832856

Written Response

1. (6 marks) For each of the reported security breaches, please 1) identify which one(s) of the security CIA properties is (are) compromised, and 2) describe a possible attack approach leading to the breach. (Note:: There are various security flaws in Cryptopia's cyber infrastructure.)

    a. Cryptopia's internet-connected and smart production machines were functioning normally since their deployment; However, on the day Cryptopia's executives were meeting with an important investor for checking the facility, the employees found they lost control over all the machines.

        i. Availability has been compromised as the system is not available when the employees want it to be available (i.e. the system should be in their control at all times and accessible unless it is undergoing routine maintenance)

        ii. The timing of this attack would lead someone to believe that it was caused by a malicious insider to sabotage the company. The attack type is most likely a logic bomb that affected all of the machines and activated on its own when the attacker knew the executives would be meeting an important investor.

    b. Cryptopia's marketing leader lost a bid for an important transaction to a competing company. An internal investigation shows that the bidding details were leaked when the marketing leader was uploading them to the auction portal.

        i. Confidentiality has been compromised in this scenario because data that should have only been privy to the marketing leader was leaked

        ii. The attack vector is in the series of man in the middle attacks because the data was uploaded correctly with no interference or at least that is what the marketing leader thought. When it was actually intercepted during upload by the attacker. Maybe the marketing leader followed a link to the auction portal from an unsuspicious looking email that redirected to a website identical to the auction portal that took the file copied it to the attackers machine then sent it to the auction portal. This attack type is an interface illusion.

    c. The company's logs revealed that a user, with username "surprise", had tampered with the configuration of the production system to cause it to slow down. Further investigation revealed that user "surprise" had no password.

        i. Confidentiality has been compromised as the system was accessed most likely by an unauthorized party (maybe an employee who was fired) and the system was tampered with. Depending on the severity of the slow down availability could also be considered to be compromised if the data takes too show up for users who need it.

ii. This attack is most likely caused by a backdoor that was probably installed maliciously as the username sounds malicious

2. (6 marks) Cryptopia did not have a formalized way to review, test, and maintain the codebases powering its production systems. You have been tasked with formalizing the process to ensure proper security controls. Please state and explain a security control you would recommend during each of the following phases of development: Code Review/Testing/Maintenance
   a. Because Cryptopia seems to be infested with malicious actors and possible corporate spys it would be wise to conduct guided code reviews so that the author must convince the reviewers why code is written the way that it is. This interrogation process could also ensure backdoors and suspicious/vulnerable code is removed before any changes are made to the systems. It would be best to have a select number of employees that are used in the code review and not to use an extra-company audit.
   b. Testing should be performed for activities a wide range of employees do. Code should be tested in both black box and white box testing methods. Supplying garbage inputs as fuzzing and checking edge cases for white box testing and keeping the tests to check if any updates made to the code still pass these tests.
   c. Because vulnerabilities can escape both code review and testing there should be an update schedule that ensures that when vulnerabilities are found they are updated and sent out as patches

3. (8 marks) After identifying the attack methods, you have been asked to reinforce the defence mechanisms. For each defence method studied in class: prevent, deter, deflect, detect, recover, discuss how they could be used to defend against any one of the attack approaches mentioned in question (1). More specifically, explain how the defence method would apply in the narrative context of the attack, and why it would help against it. (Note: You only need to explain the defences for one attack). Example: Prevent. In case of a Man-in-the-middle attack, encrypt bidding data sent to the auction portal. Encrypted bidding data would be of no use to the attacker.
   a. Deflect
      i. In the case of a man-in-the-middle attack like in case (2) we can upload false documents of competing or dummy companies to the auction portal so that the data gathered by the man-in-the-middle would have their targets set on another company that is bidding higher than the marketing manager for cryptopia.
   b. Detect
      i. In the case of the backdoor attack like in attack (3) keep track of the known users that log in and check the user's activity such as their username, how often they access the system, how long they access it for, if behaviour for a specific user changes abruptly, or if behaviour for a lot of users change drastically (such as in attack (3) all of the user's would be waiting for data

longer/using the system for longer) which can all be used to check if another backdoor that changes the system can be detected.

c. Recover

i. Like in the case of the first attack there could be a back up server that could shut down all machines and then enable them to be connected to a secondary system to allow user's to access the system in the case the main system goes down, this system would also need to be maintained and made separate from the main system to ensure the type of attack used to take down the machines don't affect the auxiliary system.

d. Deter

i. To deter against man in the middle attacks such as in case (2) upload the document as a pin/password protected file that the marketing leader would keep private such as on their secure work laptop. This would make stealing the document require more effort for the attacker to gain access to bidding details.

4. (6 marks) The IT department in Cryptopia proposed a list of custom two-factor authentication schemes that protect access to the production mobile controllers (company-owned smartphones used to control the machines). You have been asked to review these proposals. Indicate whether you would accept or reject the proposals below and explain the reason(s) behind your decision. If you reject a proposal, propose an alternative.

a. The scheme unlocks a controller if a correct password and a correct PIN are entered.

i. This scheme is not very secure as it is vulnerable to key logging, dictionary attacks, shoulder surfing, interface illusions/phishing and password/pin reuse. Passwords and pins are both vulnerable to those attacks listed above and most likely if one of those attacks reveals the password it could very well know the pin aswell. As an example if the password was shoulder surfed then the pin was probably discovered at the same time. This turns something the user "has" to something the user "knows" and things the user "knows" can be stolen

ii. Instead of a password and pin an effective 2 factor solution can be used instead which does not use sms. This will utilize a password as well as another authorization token such as from a phone app or email.

b. The scheme unlocks a controllers if a correct password is entered or a correct number (received via email) is entered.

i. I would reject this schema because the password is still vulnerable to an attack like shoulder surfing, key logging etc. But the number received via email is a good idea.

ii. Instead of just one of the authentication measures needing to be fulfilled it would be more secure to use both measures at the same time so that a password is required and the number from the email authentication is required afterward. The first measure is to ensure that the user is attempting to login and it is not just an attacker who knows the username and not the password is just guessing. Then the second measure is to ensure that if an attacker knows the username and the password they will still not have access to the controller. This would require the email password and controller password to be different and not guessable.

c. The scheme unlocks a controller if the user enters a correct password exclusively within the company's premises.
    i. Although this scheme may seem secure if a malicious actor is attempting to sabotage the company say as a form of revenge or espionage and has access to the building it could still be a vulnerability. Such as the US military when Iran was in the middle of their uranium enrichment program was in progress had pictures of the machinery in an air-gapped system and could exploit the machine after a usb drive was smuggled inside the facilities.
    ii. In this case if the system needed to be as secure as Iranian uranium enrichment facilities then pairing a password as well as some sort of biometric second factor would make the scheme more secure.

5. (4 marks) Identify the type of the following pieces of malware – i.e., whether the malware is a worm, Trojan, Ransomware, and/or Logic Bomb. Give a brief description of how it spreads or how a computer becomes infected, and the resulting effect. (A malware may be classified into more than one type.)
    a. WannaCry
        i. Was a scareware/ransomware worm that exploited windows server message block vulnerability which was discovered by the NSA and left unpatched so that the NSA could exploit it if needed. A group called The Shadow Brokers leaked this and lead to attackers developing the worm because they had the knowledge of how to attack the vulnerability. The attack would encrpyting data in the machine and require a payment of bitcoin to get access to the secret key. The NSA alerted Microsoft to patch the vulnerability before the worm could have any effect but users did not apply the update as soon as the patch was available so many machines were left vulnerable and becake infected.
    b. Code Red
        i. Was a worm that exploited a buffer overflow in microsoft IIS web server. The infected server would then deface its home page and launch attacks on other webservers such as DDOS attacks. Notable it attacked whitehouse.gov. The worm also installed a back door that made patches ineffective as the attacker could still access the machine and reinstall the worm.

Sploit1:

On Line 302 there is a buffer overflow vulnerability that we can exploit due to a snprintf receiving a buffer that is smaller than the buffer size parameter allowing us to overfill the buffer with malicious code.

Pwgen will enter the procedure print_usage when the flag -h (or any string with a '-' appended to the front that is not defined in print_usage). We wish to exploit this by giving the program as an input our sploit along with an argument that will trigger the print_usage function. The exploit is a buffer overflow shellcode injection that has been built with a NOP sled towards the shellcode and we can also include addresses back to the start of the shellcode so that if we overshoot the NOPs or shellcode we can return to the beginning of our payload and follow the sled down. Then when pwgen enters print_usage it will read the payload with snprintf and the shellcode will be launched.

To fix this vulnerability all that needs to be done is to change the buffer size parameter to match the size of the buffer which can be done by replacing

"snprintf(buffer, **BUFF_SZ**, "Usage: %s [options] …)\n" and "buffer[BUFF_SZ - 1] = 0;"

with

"snprintf(buffer, **strlen(buffer)**, "Usage: %s [options] …)" and "buffer[strlen(buffer)- 1] = 0;"

Now it is no longer possible to overflow the buffer by passing in an oversized string for buffer.

Sploit2:

There is a vulnerability within the code for this version of pwgen which we can exploit due to incomplete mediation, which allows the user to masquerade as a root user and run pwgen to print out the password for root while not being root itself. This is due to the get_uid and get_gid functions returning the uid as root (0) if the program is run using “HOME=/root pwgen -w” in the terminal. This gives any user a password that allows it to login as the superuser.

By creating a program that runs pwgen with “HOME=/root -w > /share/StolenPass.txt” we can save the password that is generated and then create a script that attempts to login as superuser, input the correct password and voila!

To remove this vulnerability we should replace the entire body of get_uid() and get_gid() with their respective functions already available in UNIX.

Sploit3:

On line 270 there is a format string vulnerability that we can exploit due to fprintf(stderr, buffer) assuming that buffer is a format string and thus any format specifiers are accepted and parsed which allows specifiers such as %x to print out the stack and %n to replace memory addresses.

The maincode will enter the procedure pwgen_args when the "-e" flag is set and line 270 is accessed by check_perms returning true. This is accomplished by our programming creating the file that check_perms is looking for (/tmp/pwgen_random) to trigger line 268's if(res): {…}. Then we need to have "buffer" contain format specifiers to overwrite a return address with an address to the shellcode that is also contained in buffer.

A fix to this is replacing line 270

```
fprintf(stderr, buffer);
```

With

```
fprintf(stderr, "%s", buffer);
```

Which will turn the buffer value into a constant string and format specifiers can't be abused as they will be read as characters instead of specifiers.

Sploit4:

We can exploit a TOCTOU vulnerability in the program within the function “check_perms” as there is an lstat call before an fopen call which means that there is a period between when we check if we have file permissions and when we open the file. During this race window it is possible to symbolically link /etc/passwd with /tmp/pwgen_random.

To exploit this vulnerability we need to run pwgen with the argument “--seed” and this will make the program ask for user input from stdin and this is where the race window opens, because the race window is created while the program is waiting for the seed. During the race window between lstat-ing and opening the file we can attempt to gain access to /etc/passwd by linking it to the /tmp/pwgen_random and like in sploit2 read the root user’s password and use a script to login as super user.

To fix this vulnerability we can edit the code so that we check to see if the lstat before opening and lstat after opening match the file and restoring the passwd file to what it was before opening.

```
if (lstat(FILENAME, &buf) == 0) { return -1; }

fd = fopen(FILENAME, "w");

fclose(fd);

chown(FILENAME, get_uid(), get_gid());
```

With

```
if (lstat(filename, &lstat_info_before) == -1) {//check errors”}

… do stuff …

if (lstat(filename, &lstat_info) == -1) {//check errors}

If (lstat_infos_before == lstat_infos) {

    fclose(fd);

    chown(FILENAME, get_uid(), get_gd());

}
```