CS 419 Midterm, Spring 2020

Instructor: Shiqing Ma

Results:

Name: **Himesh Buch**

Date: **3/13/2020**

RU NetID: **hnb17**

Multiple choices, you will get 4 points if you circle all correct answers; 2 points if you circle a subset of all correct answers; 0 points if you circle any wrong answer.

1. Which of the following are the security properties we care about?

   a. Correctness

   [✓ b. Confidentially]

   c. Concurrency

   d. Reliability

   e. Performance

   [✓ f. Integrity]

   [✓ g. Availability]

   h. Usability

   i. Accessibility

   j. Accountability

   k. Assurance

2. From the technique point of view, why do attacks happen?

   a. Unsafe languages like C/C++

   b. Von Neumann architecture

   c. Social engineering

   d. Hackers are motivated for profit

   e. Attackers are smart

   [✓ f. It is hard to build a system]

   g. The least privilege principle

3. Stuxnet can do self-replicating by attaching to Windows processes, hide itself from process listing tools, and it can also execute as a stand-alone process. Based on this description only, Stuxnet is a ?

    a. Virus

    [✓ b. Worm]

    c. Trojan

    d. Ransomware

    e. Scary ware

    f. APT

    g. Logic bomb

4. What is designed for helping operating system security?

    a. Separating CPU instruction sets to privileged and unprivileged sets

    b. Allow multiple users to use OS at the same time

    c. Logging software failure and kernel crashes

    d. Logging user logins and their activities

    e. CPU supports memory management by paging and segments

    [✓ f. Access control permissions for individual users]

    g. Using ID cars to allow building and computer lab access

5. Which of the following attacks can be prevented by input authentication and formatting?

    a. ARP spoofing

    b. DNS spoofing

    [✓ c. SQL injection]

    [✓ d. XSS]

    [✓ e. XSRF]

    f. Packet sniffing

    g. Backdoor malware

    [✓ h. DDoS]

    i. Scareware

I. (Deffie-Hellamn) For Deffie-Hellman algorithm, suppose the shared secret p=23 and g=5, Alice chooses a=5 and Bob choses b=19. A) what is the value Alice sends to Bob? B) what is the value Bob sends to Alice? C) what is the key? (20 points)

A. Here, p = 23 and g = 5. Since Alice chooses a = 5, it will send **5^5 mod 23** to Bob.

B. Now, Bob chooses b = 19, it will send **5^{19} mod 23** to Alice

C. Now, Bob will take Alice's result and raise it to its private number's power and take the mod, that is,
$(5^5)^{19}$ mod 23, and Alice will take the result sent by Bob and raise it to the power of its private number and take the mod, that is $(5^{19})^5$ mod 23. Hence, both the results will be the same (changing exponent will not change the result). Hence,

$$(5^5)^{19} \bmod 23 = (5^{19})^5 \bmod 23$$

$$(5^5)^{19} \bmod 23 = (5^5)^{19} \bmod 23 \text{ (changing the exponent)}$$

Hence, the key is **$(5^5)^{19}$ mod 23**

II. (Birthday attack) A) What is birthday paradox? B) Birthday attacks can be effective for what types of systems and why? (20 points)

A. In probability theory, the birthday paradox concerns the likelihood that, in a lot of n randomly picked individuals, some pair of them will have a similar birthday. For example, it is said that in a room of 75 individuals there's a 99.9% possibility of any two persons having the same birthday.

B. Birthday attacks can be used to malfunction digital signatures. Suppose some signature m is signed by using some cryptographic function and secret key. Suppose Alice want to trick Bob into signing a fraudulent contract. Alice can prepare a fair contract m and fraudulent one m' and finds a number of positions where m can be changed without changing the meaning of the sentences in the contract. In the end, it can have a contract looking exactly like the original one, but as a matter of fact, it will be fraudulent. It can also be made that the fraudulent contract looks exactly alike the original one, which will be a fair version that Bob can sign and fall right into the trap. This is one of the examples where birthday attack can be effective. Just like the birthday paradox, Alice here, creates the fraudulent contract by assuming both the contract and malicious contract being same, which is the same as predicting the birthdays in a room full of people

III. (Network Security) DNS resolving and attacks. A) Please explain how DNS resolving works (2 ways). B) What is Kaminsky-Style attack and why it can success, please use an example to demonstrate. (4 + 16 = 20 points)

A. The Domain Name System (DNS) is a distributed directory that resolves hostnames into machine-readable IP addresses. It is like a phone book for the internet. It basically contains information about domain names, such as email servers and even SSH fingerprints (SSHFP). Without DNS we wouldn't be able to visit the websites by typing its address, but only by typing the IP address which can be really hard for a lot of reasons. The first time when we ask the computer to look up some hostnames it looks at the local DNS cache, and if it can't find it there it has to perform a DNS query. The DNS basically queries or looks for the hostname recursively on other DNS servers. Those servers are called resolvers, which also has the cache servers. If the hostname is found the process ends there. If not, then it asks the root name servers, these name servers don't know the hostnames, but it points the DNS to the right address . It will be determined then whether to look at TLD by looking at the domain name that the user has passed. For example, if the name contains .com then it's a top-level domain (TLD). These servers then ask the authoritative servers and resolves the domain name.

B. Dan Kaminsky presented a critical Domain Name System (DNS) vulnerability that allowed attackers to send users to malicious sites. This basically allows attackers to exploit a legit website and redirect users to some malicious clone of it. The fundamental flaw here is cache poisoning which affects every DNS server that exists in the entire world. As explained in part A, if the DNS nameserver doesn't know the hostname, it will suggest on which direction to go. Nothing actually prevents any nameserver from hosting any zone, including those it doesn't really own. That means an attacker could set up a nameserver and configure an authoritative zone for any website, but it won't matter because no other nameserver ever delegates to it. This will always work because of the fundamental flaw in DNS structure itself, that is , while looking up unique entries/sibling names to www.google.com - like 1.google.com, and 2.google.com. Attackers can do this and say they're the official server for www.google.com, and they do that by telling the nameserver what www. needs to be, and the nameserver will believe the attacker. Using the same analogy, we can say that when the user ask for some www.somewebsite.com, and if for some reasons the ISP's nameserver isn't able to resolve the IP, it'll give a bunch of nameservers, meaning, point it to the right direction on where to look. In the end, it will present us with a bunch of options of nameservers and there will be no way to delegate which of them are legit and malicious nameservers because the same attack as the sibling server is possible here, meaning attackers can present 1.somewebsite.com and tell the DNS server that www has been replaced with 1 here and the server will believe that. As mentioned earlier, this is possible because of the design structure flaw in the creation of DNS architecture.

IV. (OS Security) Alice and Bob are taking one CS courses. Carl is the TA and David is the instructor. You are asked to maintain user groups and assign proper permissions to individual folders on the Linux server they use. For the folder slides, David and Carl update files (e.g., lec1.pdf, lec2.pdf) and the files are open to all students. For the folder project, Alice and Bob can only upload their own file (e.g, Alice.tar.gz, Bob.tar.gz) but not download any files here. In the meantime, they are allowed to list the files in this folder to confirm if they have submitted or not as well as the timestamp, but they are not allowed to write or create files in this folder. Thus, they will use the provided turnin program to submit. Please lit the proper user groups and users in the group (5 points), permissions for each folder (5 points) and the logic for the turnin program (10 points).

A. Users are all the system authenticated users who have certain access to files and other privileges. Groups are a bunch of users granted same permissions and privileges. Here, the users are Carl, David, Alice, and Bob. We will have two groups, as the users have different permissions, David and Carl in one group and Alice and Bob in second group. (David and Carl have same permissions, so we put them in the same group, while Alice and Bob have same privileges, so we put them in the same group). In Linux environment, Carl and David will have sudo privileges, as they will be admins and will be in the sudo group. Alice and Bob will just be simple users and won't be in sudoer's group

B. Permission for folder with slides: since Alice and Bob can only see these slides but not download it, this folder will have only read permission (in Linux environment, it will have rwxr--r--, meaning only the admin will have the privileges to make changes to the file and others will not have). It is also said that Alice and Bob cannot write or create files in the folder, that means it will have (in Linux environment) rwxr--r--. That means, only admins will have access to read, write, or create files/folders in that folders and others will only have the read permissions so that they can read and list out the files/folders

C. The turnin program will have to check the user's permission. In order to do so, it will have to check, for every user, in what group they belong to. The turnin program will basically help them create the answer files/folders and help them submit. After, checking the permission of the user, it will decide whether they had the read, write, and execute privileges or not, and regardless of that, it should help them create files/folders and help those files/folder submit for grading.