Homework 3

Question 1

a) A possible root cause for someone to try and exploit SQL injection is to run undesired SQL queries on a database. For example, a malicious user could extract data from a database that should not be accessed, create data, or/and remove data. A great way to prevent SQL injection is to use prepared statements. Prepare statements in SQL allows the parameters of a query to be sent separately from the command to the database rather than being combined into a SQL string. Doing this will prevent SQL injections to happen.

b) A double-free vulnerability is a memory error that makes programs access memory beyond allocated heap space. Double-free often happens when you try to free up a pointer multiple time. Doing this will cause the blocks of the heap space to become corrupted. A way to prevent double-free is by after freeing a pointer to set it to NULL. This means that if you try to free it again nothing will happen.

c) Format string vulnerability happens when an attacker can input an argument to a for example a printf function. By doing this the malicious user can perform arbitrary memory write for example by inputting more parameters into the argument. An attacker can use the %n parameter type to write to arbitrary memory. Because the %n tells the formatter to write the number of bytes up to an address specified. By formatting a string in a specific way using %n the attacker can write to values on the stack.

Question 2

a) Data flow is the process of data being modified from the original input. For example, if a key is inputted by a function and the function updates/changes the input value the data flow will have propagated to the function return value. Compared to information flow the return value is not propagated of the input value. Information flow is preffered in security analysis because it is extremely important to observe the information being exchanged from two different processes and how that impacts the program/leakage of information.

b) Yes, by tainting each byte of input we can see which input influences the different behavior of the program. Meaning that instead of trying every single mutated input to the program it can understand with the help of taint analysis which input it needs to try/change. Because that specific input influences a certain part of the program, and where the tainted input ends up in memory.

c) Yes, taint analysis can be used to find format string vulnerabilities. In that situation, the taint source will be the untrusted input of a user which can affect the area where the format string vulnerability exists. The taint sink is where the information/data ends up which in this case will be on the stack.

Question 3

a) The three key components that form a side channel attack are secret depends on the program information flow which affects the physical environment. The physical environment can be exploited by a attacker to find secret behavior.

b) Yes, it would. If the piece of software is built so that the attacker and user are sharing the same cache the attacker could perform a cache-side channel attack. If the private user images are being accessed the attacker will be able to distinctly tell that they are different due to the different cache accesses. If you can distinctly identify one image from others, you

can recover one bit from the private image. By doing this over several iterations the attacker will be able to recover several bits.

c) Yes, I would say that it is doable to use taint analysis to detect software timing-based side-channel vulnerabilities in certain situations. If certain inputs take different amount of time for the program to process, there is a chance that depending on the input it can have leakage in information. Meaning that if an input takes more time to process it could end up where an attacker could access it. But this is a specific case and in a more general case taint analysis wouldn't yield anything.

Question 4

a) The problem with saving password in plaintext format is that if an attacker gets their hands on the database or file that the passwords are stored in the attacker now has access to that sensitive data.

b) The advantage of hashing password instead of encrypting them is because a hashing function is a one-way function. Meaning that with just the hash value the password can never be extracted (excluding random input guessing). While encrypting password means that the exists a key which can decrypt the cipher text again. Meaning that it becomes extremely harder for an attacker to extract the original passwords from a hashed value rather than a cipher value.

c) A problem with hashed passwords is that the same password produces the same hashed value. This means that by just randomly guessing passwords a hacker could identify hashed passwords if they have access to the hash function. But this also means that by just looking at the hashed values we can identify which users have the same passwords. By adding salt when hashing a password, two exactly the same passwords would end up as two different hashed values and therefore it becomes a lot more secure when storing them in a database.

Question 5

a) When using biometrics there are two important keywords: fraud and insult. Fraud points towards the direction that an unauthorized person gets access, and insult is the authorized person gets access. The equal error rate is when the fraud rate is equal to the insult rate. The equal error rate tells us that if the sensitivity is too low the fraud is high and if the sensitivity is high the insult is low. Meaning that the best sensitivity level is where both are equal.

b) The different characteristics of reliable biometric-based voice recognition would be that it is fast and reliable. Meaning that both the insult should be high, and the fraud level should be low. It is also important that it should be quick and simple for the user to enter/speak in this case. Meaning that there shouldn't be an extremely long and difficult sentence that the user needs to say.

c) A possible attack on biometric-based voice recognition could be if an attacker gets access to a recording of the authorized voice. If the biometric system just wants the person trying to access the system say a phrase an unauthorized person could get access to the system. But a way to deal with this could be to have the system require the person to say a specific given phrase (random each time accessing the system). Doing this would make it a lot harder for the attacker to get access to the system. Generally speaking, it is extremely hard to forge biometrics which in this case is voice meaning that it would be hard for an attacker in this case to forge the victims' voice.

Question 6

a) Anomaly-based IDS is used for detecting unknown threats that don't align with the normal behavior of a user. An anomaly-based IDS needs to be dynamic to try and understand exactly what a user uses the system on a standard base and then from there can detect any abnormal activities on the system. Compared to signature-based IDS which is used for identifying known threats by checking the content of programs for specific byte sequences, domains, and/or specific hashes. But there is a lot an attacker that try to hide/change the information and structure of the attack which makes it nearly impossible to use signature-based IDS to detect threats.

b) Other sensible statistics that could be consider for an anomaly-based IDS could be:
   a. End location of the traffic (location where the data is being sent/accessed from)
   b. Time based (outside of work hours)
   c. New unauthorized devices being added to the network
   d. Accessing unpopular domains (Accessing domains that have never been recorded by the network)

c) It might not always be a good idea to combine several statistics due to the more statistics that an IDS goes off on, the easier it might be for an attacker to convince the IDS that unauthorized traffic to use the system.

d) By combining a lot of different types of statistics an IDS could come to a good conclusion if certain activities are abnormal or not. This means that there will be a lot fewer fake alarms for authorized traffic.

Question 7

a) The type of attack is Ping flood. The attack works by abusing the ping command. The ping command works by sending an ICMP echo request to a target host. This means that ping flood is simply a DoS attack to overwhelm the victim which in this case is the company that is selling a product. The best way to both detects and deal with ping flooding is by monitoring the target device. If there is a peak in requests, it might be that an attack is ongoing during that time. Firewalls can help block harmful requests and/or limit the number of pings (ICMP) requests.

b) This type of attack is SYN flooding. Because of the TCP 3-way handshake after an SYN package has been received the server/host will remember the connection as half opened. When it is in this state it consumes resources because TCP is a stateful protocol where it remembers each state of connection. But having too many halves open connections will lead to all the resources being consumed which means that normal requests to any other IP addresses can't be accessed. The attack can be prevented by identifying anomaly traffic using an IPS (Intrusion Prevention System). But the attack can also be prevented/detected by analyzing the traffic on the network. If a lot of TCP connections are created that are left half opened on the network an attack could be ongoing.

Question 8

a) **Packet filter** existing on the network layer. An advantage of packet filtering is that it is fast, but it doesn't have a concept of state and can't handle/see TCP connections. **Stateful packet** works at the transport layer. Adds a state to the packet filter which means that it can remember TCP connections. The disadvantage of stateful packet is that it is slower than packet filtering. **Application proxy** works at the application layer. The goal of an application

proxy is to make sure that the data is safe before letting it in. It can filter out bad data from the application layer like viruses but at the cost of speed.

b) One of the reasons for the application proxy to scan the incoming application data for viruses is because it can prevent some scans that stateful packet filtering cannot which each host wouldn't be able to do. Another thing is also that if the application proxy will only let in the data if it safe. That means that if the hosts would scan the application data themselves the viruses will have to be let further into the network.

c) The Firewalk port scanning tool the TTL (time to live) is crucial because it uses it to determine the hops to the firewall. This means that if the packet filter resets the TTL field to 255 an attacker will never be able to tell exactly how many hops/routers the packet takes. Doing this does prevent a Firewalk to be performed by an attacker and therefore the attack cannot find out the internal network topological structure.