**MIS 6330: IT Security**

**Individual Homework 6**

1. *SQL Injection* is a type of attack where attacker uses bad input (consisting of malicious SQL commands) to perform unauthorized SQL operations in the victim’s backend system. Since, the system of victim is expecting an input from the user, it takes the bad input and runs the intended SQL command using that input. However, while the intended command runs, after it, the malicious command that the attacker added in the input also runs and performs unauthorized SQL operations.

For example, for input expecting a name to search data based on it,  
the SQL command intended to run in the background would be like  
“SELECT \* FROM ABC WHERE Name = $UserName”

Here, $UserName is the input entered by the user. If the attacker enters value for user name as ‘XYZ; DELETE TABLE ABC’, the system will take the whole value and append it to the intended SQL command.

When the SQL command runs, it will fetch values based on XYZ and then delete all records of table ABC. The delete operation was not intended by the victim and yet it’d run and delete the records.

To prevent these attacks, following steps could be taken:

* All inputs should be sanitized first or checked for anomaly in terms   
  of expected input format.

Only if the checks are fine and the input is in the intended format,   
the command should run, otherwise it should throw some error message   
to the user.

This can be done using inbuilt commands for DBMS like MySQL or with use of Regex while taking the input.

* Also, *Input fuzzing* can be used to test the system for any vulnerabilities   
  against various types of input.
* Use of *the principle of least priviledge* could be done to make sure   
  users of different groups can only perform operations, they are   
  authorized to.

This way, even the attacker enters malicious input, the DBMS   
would block the command in the malicious input to run since   
the attacker is not granted the access to run that type of command.

1. *Input fuzzing* is a testing method for checking vulnerabilities for different kinds of input. Very big set of randomly generated inputs is used to test and see if for any input, the system is giving unintended outputs or not.

It is very cheap and easily available for testing systems to make sure no unintended effects can be reproduced by attackers by using bad input.

*The principle of least privilege* means using different ways of limiting the priviledges available to different types of users based on necessity to prevent any user to perform unauthorized operations.

Forexample,the end users don’t need admin level access on the files of the system and so, by using this principle, the priviledges of end users are limited to only viewing the files.

However, maintenance engineers and developers require priviledge to modify execute files, so their priviledges could be limited to exactly that.

Similarly, different user groups could be given different levels of priviledges, so that no one can perform unauthorized operations on the system.

1. I traced the path to utdallas.edu (using *tracert* *utdallas.edu*). The output I got is shown below.
   1. Based on this output, each IP packet traveling from my computer to utdallas.edu gets framed and reframed for 7 times. 6 times during   
      the process of moving from one router to other and 1 time when   
      it is initially sent from my computer to my network’s router.

Caution: Last one is the destination

* 1. Each packet crosses 6 routers when going from my home to UTD.
  2. The bottleneck is the route between routers with IP addresses 192.168.1.1 and 47.187.192.1, between these two routers a packet traveling from my home to UTD spends most of its time as per the given data.

This is because the timings given here are round trip timings from source to each of the given routers.

This means between 1st router and 2nd router, it’d take approximately 4 seconds (5-1 seconds i.e time of second round trip-time of first round trip)

C:\Users\axl144730>tracert utdallas.edu

Tracing route to utdallas.edu [104.16.43.54]

over a maximum of 30 hops:

1 1 ms 1 ms 1 ms Wireless\_Broadband\_Router.home [192.168.1.1]

2 5 ms 5 ms 5 ms 47.187.192.1

3 6 ms 7 ms 6 ms 172.102.52.84

4 8 ms 7 ms 7 ms ae7---0.scr01.dlls.tx.frontiernet.net [74.40.3.17]

5 9 ms 8 ms 8 ms ae0---0.cbr01.dlls.tx.frontiernet.net [74.40.4.14]

6 8 ms 8 ms 7 ms 13335.dal.equinix.com [206.223.118.145]

7 8 ms 7 ms 7 ms 104.16.43.54

Trace complete.

Physical Address. . . . . . . . . : 44-39-D4-5A-C6-E2

IPv4 Address. . . . . . . . . . . : 192.168.1.10

Subnet Mask . . . . . . . . . . . : 255.255.255.0

Default Gateway . . . . . . . . . : 192.168.1.1

1. Physical Address will not change while, rest of the three values are likely to change.

Physical Address does not change as it is unique for each LAN card   
and the LAN card of the laptop was not changed while going   
from home to school.

However, rest of them are likely to change as,

IPv4 Address is the address given to the laptop by either the DHCP server of the network used by the laptop (in most cases), or can be set manually within a given network.

Therefore, it is dependent on the network in which the laptop is connected and since the network changes while going from home to school, IP address is likely to change too.

Similary, subnet mask is dependent on the network and different networks could have different subnet mask. So, subnet mask is also likely to change while going from home to school.

Default gateway is also dependent on the network as it is the IP address of the router of the network. Since, home and school network would have different routers and their IP addresses will be based on the network they are in, default gateway is also like to change while going from home to school.