CON101: Software and Hardware Vulnerabilities

Apar Ahuja
2019CS10465

November 2020

Software/Hardware vulnerabilities are weaknesses in a system that can be exploited by malicious attackers. In this report, we will be exploring three such system vulnerabilities -

1. **Buffer Overflow Vulnerability** - Buffer overflow occurs when we try to write something in the memory that exceeds(overflows) the memory space available to the program. This overflow can lead to significant issues including software crash, data corruption and execution of malicious code. System memory runs programs using a stack. When a function is called, parameters are created, return address is stored, a buffer memory(say 500 bytes) is allocated and execution takes place. After this, the instruction pointer returns back to the address where it was. To exploit the process, we try to write something(using strcpy as an example) into the buffer that's longer than buffer memory(say 510 bytes) and overflows past it and right onto the return variable. The return address is now modified! (to say 0x323232). Now if the memory address 0x323232 has a malicious code in its place, the system will try to run it. Buffer Overflow is a very high-risk vulnerability and has led to significant internet worms such as the Code Red worm against Microsoft and SQL Slammer worm in Microsoft SQL servers. A protective measure against this is using strongly typed languages like Python and Java instead of C/C++. These languages can help fight against this vulnerability. Further using a pointer and buffer overflow protection, safe libraries, etc. can help protect the system.

2. **OS Command Injection or Shell Injection Vulnerability** - Every website on the internet is hosted on some Operating System(OS). We can execute commands in the shell of any OS (terminal in Linux, command prompt in windows, etc.). In Command Injection the attacker injects OS commands as input to a website which gives the attacker full access to the OS on which the application has been hosted. OS injection is a very critical and high priority vulnerability. Attackers can use the ping command to check if the server is online by sending packets(server is online if the server sends back a response). Ping is an OS-level command. Similarly, the user(attacker) can send other OS-level commands to show directory listings, create or delete files and perform malicious

activities. This is possible because servers have by default admin privileges. If the OS commands are working at the client-side, then the system is said to be vulnerable to injection. Some ways to protect the system from injection are - maintain a white-list of characters users can input and block abnormal user input or use APIs that can filter and sanitize the input data before execution.

3. **SQL Injection** - SQL Injection is an injection technique to execute malicious SQL queries. Assume we have a username-password based login system. Here the queries are sent to the database to retrieve data of all users with username = 'user_input' AND password = 'password_input'. A SQL injection with username = ' 'OR 1=1– ' returns True for all values as 1=1 is a true statement. We can prevent SQL injection attacks by using special functions such that user input is taken as a string, i.e. "OR 1=1–" is taken as a string and not a logical command and hence the standard string matching takes place in the database. Similarly limiting characters available to use for passwords, pattern checking, and form validation are some other ways to prevent attacks.