**FPRT Core CSE**

**Question-1:**

**Ping:** Ping is known to be the time taken by message from to reach from origin to destination. It is usually measured in millisecond. It is the total time taken by data to get transmitted from our device to server and back to our device.

**Trace Route:** It is network related tool basically used in analysing & diagnostics to track the path of data set over an IP network. Traceroute allow us to find precisely how data travelled from our computer to another. It gives us a list of computer and there Ip address which are involved in the whole process of data transmission.

The only similarity between Ping & Trace route is that both are network diagnostics tool which are based on Internet Control Message Protocol.

**Question-2:**

OWASP stand for Open Web Application Security Project, an international organization which deals with Web application security. They provide tools so that anyone who wants to can improve their web application security.

There are 10 vulnerabilities listed by OWASP which can occur in our web application:

**Injection:** Injection occurs when an attacker exploits insecure code to insert their own code into a program. Because the program is unable to determine code inserted in this way from its own code, attackers are able to use injection attacks to access secure areas and confidential information as though they are users.

### **Broken Authentication**: Incorrectly implemented authentication and session management calls can be a huge security risk. If attackers notice these vulnerabilities, they may be able to easily hack user data of authenticated user.

### **Sensitive Data Exposure:** APIs which allow developers to connect their application to third-party services like Google Maps and others but, some APIs rely on insecure data transmission methods, which attackers can exploit to gain access to usernames, passwords, and other sensitive information.

### **Broken Access Control:** If authentication and access restriction are not properly implemented, it's easy for attackers to take whatever they want. With broken access control flaws, unauthenticated or unauthorized users may have access to sensitive files and systems.

### **Cross-Site Scripting:** Hackers take advantage of APIs and DOM manipulation to retrieve data from or send commands to your application.

### **Components with Known Vulnerabilities**: No matter how secure your own code is, attackers can exploit APIs, dependencies and other third-party components if they are not themselves secure. A static analysis accompanied by a [software composition analysis](https://www.veracode.com/products/software-composition-analysis) can locate and help neutralize insecure components in your application.

### **Insufficient Logging and Monitoring:** Failing to log errors or attacks and poor monitoring practices can introduce a human element to security risks. To prevent issues with insufficient logging and monitoring, make sure that all login failures, access control failures, and server-side input validation failures are logged with context so that you can identify suspicious activity. [Penetration testing](https://www.veracode.com/security/penetration-testing) is a great way to find areas of your application with insufficient logging too.

#### XML External Entities: This is an attack against a web application that parses XML\* input. This input can reference an external entity, attempting to exploit a vulnerability in the parser.

### **Insecure Deserialization:** objects that have been written to disks or otherwise saved, can be used to remotely execute code in your application. This flaw occurs when an attacker uses untrusted data to manipulate an application, initiate a denial of service (DoS) attack, or execute unpredictable code to change the behaviour of the application.

### **Security Misconfiguration:** Just like misconfigured access controls, more general security configuration errors are huge risks that give attackers quick, easy access to sensitive data and site areas.

**Question-3:**

**Process:** A process is any program that is in execution phase. In other words, we can say that process is the execution of a program that allows you to perform the appropriate actions specified in a program. The OS helps you to create, schedule, and terminates the processes which is used by CPU. The other processes created by the main process are called child process. A process operation can be easily controlled with the help of PCB (Process Control Block). You can consider it as the brain of the process.

**Thread:** Thread is merely a part of a process which means a process can have multiple threads or group of threads form a process. A thread has 3 states: running, ready, and blocked.

|  |  |
| --- | --- |
| **Process** | **Thread** |
| Process consumes more resources. | Thread consumes less resources. |
| Process takes more time to terminate. | Thread takes less time to terminate. |
| Process has its own Process Control Block, Stack and Address Space. | Thread has Parents’ PCB, its own Thread Control Block and Stack and common Address space. |
|  |  |

**Question-4:**

|  |  |
| --- | --- |
| **FORK** | **EXEC** |
| Fork is a function in UNIX operating system that is used to generate a duplicate of particular process by creating two simultaneous executing processes of a program. | Exec is an operation in a UNIX operating system that creates a process by replacing the previous process. |
| The new process created by a fork system call is identical to the parent process | The new program which replaces the current process is loaded and is run from the entry point. |
| After the fork system call, both the child and parent processes will be executed simultaneously. | After the exec system call the control never get transferred to the original program until and unless an exec () error occurs. |

**Question-5:**

Nohup is short for “No Hangups.” It’s not a command that you run by itself. When exiting the shell of a Linux System, all running processes are usually terminated. It is a command in Linux systems that keep processes running even after exiting the shell or terminal.

Nohup prevents the processes or jobs from receiving the SIGHUP (Signal Hang UP) signal. This is a signal that is sent to a process upon closing or exiting the terminal.

**Question-6:**

A swap file is a [file](https://techterms.com/definition/file) that contains data retrieved from system [memory](https://techterms.com/definition/memory), or [RAM](https://techterms.com/definition/ram). By transferring data from RAM to a [secondary storage device](https://techterms.com/definition/secondarystorage) in the form of a swap file, a computer is able to free up memory for other programs. Swap files are a type of [virtual memory](https://techterms.com/definition/virtualmemory), since they are not stored in physical RAM. They extend the amount of available memory a computer can access by swapping memory used by idle [processes](https://techterms.com/definition/process).

This whole process is called as **PAGING** which is a common type of memory management.

While swap files are a convenient way to increase the amount of available system memory, they can also decrease system performance. For instance, if a computer is using nearly all of its physical memory, the system may need to frequently swap data between RAM and swap files. Since reading data from a secondary is much slower than accessing RAM, repeatedly swapping memory may cause noticeable delays.

**Question-7:**

Unix/Linux command which are used to change the file ownership and permissions are:

1. **ls - l** on terminal gives permissions in Linux.
2. **chmod permissions filename** on terminal us used to change mode and set permission (read, write & execute) on file/directory.
3. **chown user** is used to change the user & group for a file directory.
4. **chgrp** command can change the group ownership**/group filename.**

**Question-9:**

A distributed system is a [network](https://www.ibm.com/cloud/learn/networking-a-complete-guide) that stores data on more than one node (physical or [virtual machines](https://www.ibm.com/cloud/learn/virtual-machines)) at the same time. Cloud applications are distributed systems, it’s essential to understand the CAP theorem when designing a cloud app.   
  
The CAP theorem applies a similar type of logic to distributed systems—namely, that a distributed system can deliver only two of three desired characteristics: **consistency**, **availability**, and **partition tolerance.**

### **Consistency:**

This means the data must be throughout similar for all the users across the platform. For this to take place data must be sent to server instantaneously.

### **Availability**

Availability means that that any client making a request for data gets a response, even if one or more servers are down.

### **Partition tolerance:**

A partition is a communication break within a distributed system a lost or temporarily delayed connection between two server. Partition tolerance means that the cluster must continue to work despite any number of communication breakdowns between servers in the system.

**Question-11: Difference between SQL and NoSQL**

|  |  |
| --- | --- |
| **SQL** | **NO-SQL** |
| SQL databases are primarily called Relational Databases | NoSQL databases are primarily called as Non-relational database |
| Uses SQL syntax and queries to analyse and get the data for further insights | NoSQL database system consists of various kind of database technologies. |
| Structured query language (SQL) | No declarative query language |
| These databases are table-based databases | These databases can be document based, key-value pairs, graph databases |
| SQL databases have a predefined schema | NoSQL databases use dynamic schema for unstructured data. |
| An ideal choice for the complex query intensive environment. | It is not good fit complex queries. |
| SQL databases are not suitable for hierarchical data storage. | More suitable for the hierarchical data store |
| Example: Oracle, Postgres, and MS-SQL. | Example: [MongoDB](https://www.guru99.com/mongodb-tutorials.html), Redis, Neo4j, Cassandra, Hbase. |

**Question-12:**

DDoS is one of the most popular types of denial-of-service attack. Instead of launching the attack from a single computer, attackers use many distributed machines like computers or smart devices in different locations to overwhelm the target.

Typically, attackers generate large volumes of packets or requests ultimately overwhelming the target system. In case of a Distributed Denial of Service (DDoS) attack, and the attacker uses multiple compromised or controlled sources to generate the attack.

**There are several measures that you can use to protect your business from a DDoS attack:**

1. **Have a plan**. To start, set up a DDoS response plan. Putting a plan in place means going through your system and working out any possible security vulnerabilities.
2. **Allocate roles**. You should have a defined response team in place for your data centers as well as in your network administration or IT teams. Make sure everyone knows their role and who to talk to if they need to escalate the situation
3. **Install protection tools**. Make sure you have appropriate protection tools installed for both your networks and your applications. This includes such key tools as firewalls, network monitoring software, anti-virus and anti-malware programs, as well as threat monitoring systems.
4. **Keep everything up to date**. All these systems should be kept up to date, to make sure that any bugs or issues are fixed. Detecting threats as early as possible is the best way to prevent a DDoS attack from taking down important network infrastructures.

**Question-13:**

Hard links and symbolic links are two different methods to refer to a file in the hard drive. These methods are part of the filesystem that organizes what file is what and where.

Hard Links: A hard link is essentially a synced carbon copy of a file that refers directly to the inode of a file.

Sym links: Symbolic links on the other hand refer directly to the file which refers to the inode, a shortcut.

**Question-8:**

A transaction can be defined as a group of tasks. A single task is the minimum processing unit which cannot be divided further. A transaction in a database system must have ACID property.

ACID stands for Atomicity, Consistency, Isolation, and Durability in order to ensure accuracy, completeness, and data integrity.

* Atomicity − This property states that a transaction must be treated as an atomic unit, that is, either all of its operations are executed or none. There must be no state in a database where a transaction is left partially completed.
* Consistency − The database must remain in a consistent state after any transaction. No transaction should have any adverse effect on the data residing in the database. The state of the database before and after the execution must be consistent.
* Durability − The database should be durable enough to hold all its latest updates even if the system fails or restarts. If a transaction updates a chunk of data in a database and commits, then the database will hold the modified data.
* Isolation − In a database system where more than one transaction is being executed simultaneously and in parallel, the property of isolation states that all the transactions will be carried out and executed as if it is the only transaction in the system