Assignment

Monday, August 8, 2022 7:28 PM

Assignment-1:

1. **Types of cyber attacks**
   1. Malware :
      Malware is a term used to describe malicious software that a user installs when he clicks suspicious links. Ex: spyware, rans omware, viruses, worms.
   2. Phishing:
      Getting confidential information by making the user believe like you are an official. Ex: credit card frauds.
   3. Man-in-the-middle attack:
      Or eavesdropping attack. This happens when an attacker inserts himself into a two-party transaction. Ex: Attacker can insert himself between a user and network in a unsecure public Wi-Fi.
   4. Denial-of-service attack:
      A denial of service attack floods system with traffic so that all the resources are exhausted. Mainly done on switches. By fl ooding the switches, you stop the traffic from/to all the servers connected to this switch.
   5. SQL injection:
      When attacker provides false/malicious information to get extra information from server if it uses SQL tables. Attacker explo its the fact that the server needs parameters or some information from the user to run some queries.
   6. Zero-day Exploit:
      Attackers hit after a network vulnerability is detected but not resolved yet. Attackers exploit the vulnerability and damage the servers before a solution is deployed for the vulnerability.
   7. DNS Tunneling:
      Uses DNS protocol to communicate non-DNS traffic through port 53. It sends Http and other protocol traffic over DNS.

Assignment-2:

1. **Define monolithic and microservice architecture and differentiate between REST and SOAP.**

Monolithic - All the program components or functionalities are tightly coupled and every component must be present for the code to be compiled. If one function gets down. All the other functions will get down too.

Microservices - The application is done as a collection of services/features which handles discrete tasks. These tasks communicate with each other for complete working of the application. The communication is done with the help of API's.

2. **Differentiate between REST and SOAP**

REST API (Representational State Transfer):
Confirms to the constraints of REST architectural style and allows interaction between RESTful microservices.

Constraints of REST architecture:

1. Uniform interface.
2. Stateless
3. Cacheable
4. Client-Server
5. Layered System
6. Code on Demand

Main differences of SOAP and REST:

1. SOAP is a protocol whereas REST is a architectural style.
2. SOAP uses only XML for data exchange but REST can use any format for the data like XML,Json.
3. Rest can use SOAP but SOAP cannot use REST.
4. SOAP requires more bandwidth.
5. SOAP has ACID compliance.
6. In REST there are resources to which create, read, update, delete can be done.

Assignment-3:

1. **What are the types of manual testing and prepare small description about each manual testing type.**

Types of manual testing :

1. Acceptance Testing :

User Acceptance Testing (UAT) is performed by the client or end-user, to confirm that the software meets the agreed requirements. Sometimes called pre-production testing, it takes place during the final phase before releasing the product to market.

UAT is an example of functional testing and types of acceptance testing include Alpha (executed within the organization) and Beta (where the application is released to a limited market to generate user feedback).

2. Black box testing:

   Also known as behavioral testing, this method aims to analyze an application's functionality from the end-user's perspective. The internal code structure is not visible during testing (hence the name "Black Box"), so testers are only aware of the inputs and expected outputs of the software.

3. Integration testing:

   Integration Testing is the process of testing an application with two or more integrating components. It is performed once the individual components have been unit-tested, and aims to identify problems with the interfaces and the interactions between them.

   The two main methods are the Bottom-Up Approach (moving steadily from the bottom module to the top module) and Top-Down Approach (the opposite).

4. System testing:

   System Testing means testing the system as a whole, once all its components have been unit-tested and integrated. It checks that the complete application works as intended, by comparing it against the original requirements.

   Also called end-to-end testing, it typically involves installability testing (does the software install correctly?) and recovery testing (can the application recover from hardware crashes and network failures?).

5. Unit testing:

   This is when the individual units or components of an application's source code are tested, to make sure each function performs as expected. It is usually carried out by developers rather than engineers, as it requires detailed knowledge of the internal program design and code.

6. White box testing:

   Sometimes called transparent box testing or structural testing, this is a method of testing the internal structures or workings of an application. It is performed by the developer, who checks the software's internal codes before passing it to a test engineer.

   The main focus of White Box Testing is on strengthening security and improving the software's design and usability. A combination of Black Box and White Box testing is known as Gray Box Testing.