MODULE 7 SECURITY AUDITING & FAULT TOLERANCE

It’s is a systematic, measurable technical assessment of how the organization’s security policy is employed at a specific site.

What is a Security Audit?

You may see the phrase "penetration test" used interchangeably with the phrase "computer security audit". They are not the same thing. A penetration test (also known as a pen-test) is a very narrowly focused attempt to look for security holes in a critical resource, such as a firewall or Web server. Penetration testers may only be looking at one service on a network resource. They usually operate from outside the firewall with minimal inside information in order to more realistically simulate the means by which a hacker would attack the site.

On the other hand, a computer security audit is a systematic, measurable technical assessment of how the organization's security policy is employed at a specific site. Computer security auditors work with the full knowledge of the organization, at times with considerable inside information, in order to understand the resources to be audited. How to manage a successful audit.

How to manage Security Audit?

- Establish a security baseline through annual audits.
- Spell out your objectives.
- Choose auditors with "real" security experience.
- Involve business unit managers early.
- Make sure auditors rely on experience, not just checklists.
- Insist that the auditor's report reflects your organization's risks.

There are a number of key questions that security audits should attempt to answer:

- Are passwords difficult to crack?
- Are there access control lists (ACLs) in place on network devices to control who has access to shared data?
- Are there audit logs to record who accesses data?
- Are the audit logs reviewed?
- Are the security settings for operating systems in accordance with accepted industry security practices?
- Have all unnecessary applications and computer services been eliminated for each system?

- Are these operating systems and commercial applications patched to current levels?
- How is backup media stored? Who has access to it? Is it up-to-date?
- Is there a disaster recovery plan? Have the participants and stakeholders ever rehearsed the disaster recovery plan?
- Are there adequate cryptographic tools in place to govern data encryption, and have these tools been properly configured?
- Have custom-built applications been written with security in mind?
- How have these custom applications been tested for security flaws?
- How are configuration and code changes documented at every level? How are these records reviewed and who conducts the review?

Fault Tolerance

Introduction (Fault Tolerance): - The property that enables a system to continue operates properly in the event of failure or one or more faults. It's a configuration that prevents a computer or network device from falling in the event of an unexpected problem or error.

Steps for Fault Tolerance: -

- **Power Failure: -** Having a power backup for computer or network adapter that can properly make sure that the administrator gets time to turn off the machine.

- **Power Surge: -** If the power backup is not connected to the computer the UPS does not provide surge protection, a surge protector connected to the computer or network device would help prevent the device from falling in the event of a power surge

- **Data Loss:** - Running backups on daily basis or monthly basis can help keep the important information safely stored on the device, also user can create a mirror of the data on an alternate location.

- **Device or Computer Failure:** - Having a second device, computer, or computer or network components available in the event of failure to prevent a long down time.

- **Unauthorized access:** - If the network is connected to an Internet then installing firewalls can be a wise way to avoid network form and infection of virus or intrusions.

- **Frequently Check for Updates:** - Making sure the operating system and any running programs have the latest updates

- **Lock device or Password protect computer:** - When not is use the lock computer and store the machine or network device in a secure area.

- **Overload:** - Setup an alternative machine or network device that can be used as an alternative access point or can be used as an alternative access point or can share the load either through a load balancing or round robin setup.

- **Virus:** - Making sure the antivirus the company or individual is using on computer or for network device its definition is updated.

Techniques of Fault Tolerance: -

- **Hardware Redundancy: -** There are basically two approaches in hardware redundancy: addition of replicated modules and use of extra circuits for fault detection

    1. **Module Replication**: - To avoid wrong results and actions being made, it desirable that the failing modules stop execution when a fault is detected, with a small fault latency

- **Information Redundancy: -** Information redundancy is the addition of extra information to date to allow error detection and correction. This typically error- detection codes, error-correcting codes (ECC) and self-checking circuits.

    1. **Error-Detection (and correction) codes: -** Parity codes are used in most modern computers for memory error detection. This is simple code that does not require much additional hardware. Codes can also be error correcting can contain error but contain enough redundancy to recover data

    2. **Consistency checking: -** This is verification of the results being reasonable. For example range checks address check and arithmetic operation checking.

    3. **Self-Checking logic: -** If the circuit self-detects the existence of fault then the circuit is said to be having self-checking fault tolerance logic

- **Software Redundancy: -** important differences between software and hardware errors. Physical errors (in hardware) will not recur after they have been discovered and corrected. Unfortunately, this is not the case with software errors. In the process of correcting a programming error, new errors are likely to be created. Software development is also a more complex and immature art than hardware design.

    1. **N-Version Programming and software Fault tolerance: -** Writing the program N times the operate all N programs in parallel, and take a majority vote for each answer, also writing the program transaction, using a consistency check at the end and if conditions are not met it should work the second time

    2. **Software Fault Detection: -** Many faults can be detected in much the same way as hardware faults are detected

- **Time Redundancy**: - Hardware- and information- redundancy requires extra hardware. This could be avoided by doing operations several times in the same module and check the results, instead of doing it in parallel on several modules and compare the outputs. This reduces the amount of hardware at the expense of using additional time, and is especially suitable if faults are mostly transient. It could also be used to distinguish between permanent and transient faults.

- **Load Balancing:** - It is a method that acts as a reverse proxy and distributes network or application traffic across a number of servers. Load Balancing is used to increase the capability and reliability of applications

Clusters of Fault Tolerance

- **The HA Clusters**: -This cluster provides high availability of data through replication of data on cluster of machine. This cluster consists of two or more machines interconnected by a high speed bus

- **ClustRa**: - ClustRa is a database engine originally developed for telephony application. In addition to high availability, these applications require high throughput and real time response time.