TASK-2

K. Naveen Abhiram

21BCE7357

Determine the vulnerabilities in the open ports

Port no's (20,21,22,23,25,53,69,80,110,123,143,443)

1. Port 20 (FTP Data) and 21 (FTP Control):

Vulnerabilities:

Insecure FTP passwords, unrestricted access, and obsolete FTP servers that can be exploited.

Effects:

Attackers can get unauthorized access to server files by using brute force or guessing weak FTP passwords. Anyone can view, potentially edit, or steal data using anonymous access without authentication. Older FTP servers might contain known security flaws that attackers could use to undermine the integrity and confidentiality of the service.

Example:

An FTP server on port 21 utilizes weak passwords and permits anonymous access. This can be taken advantage of by an attacker who gains unauthorized access and then steals confidential information or perhaps uploads harmful files to the server.

2. Port 22 (SSH):

Vulnerabilities:

Weak SSH key pairs, obsolete SSH versions with known security flaws, or SSH services that were made available to the whole internet.

Effects:

Cracking weak SSH key pairs enables attackers to access the server without authorization. The security of the server might be compromised by using known flaws in outdated SSH software versions. Without adequate security measures, exposing SSH services to the public internet raises the danger of brute force attacks and illegal access.

Example:

A server listening on port 22 employs an unpatched version of SSH. By taking advantage of one of these weaknesses, a hacker can access the server without authorization and perhaps jeopardize its security.

3. Port 23 (Telnet):

Vulnerabilities:

Weak passwords, unencrypted communication, and possible system intrusion.

Effects:

Attackers may intercept unencrypted Telnet transmission, revealing login information and perhaps sensitive information. Insecure system access might result from brute-force attempts or easy password guessing. Unauthorized access may lead to resource misuse, system compromise, or data theft.

Example:

Since Telnet communication through Port 23 isn't secured, a hacker might potentially capture confidential data, such login credentials, as it travels across the network.

4. Port 25 (SMTP):

Vulnerabilities:

vulnerabilities related to open relays, email misuse, and email spoofing.

Effects:

Open SMTP relays can be used by spammers to send large amounts of unsolicited email, resulting in email abuse and perhaps resulting in the server's IP address being blacklisted. Phishing attempts and other email-based frauds may result from email spoofing flaws that let attackers seem to be trusted senders.

Example:

Email abuse and possibly blacklisting result from a misconfigured open relay SMTP server on Port 25 that spammers might exploit to send large numbers of unwanted emails.

5. Port 53 (DNS):

Vulnerabilities:

DNS servers with incorrect configurations that enable DNS cache poisoning or amplification attacks.

Effects:

Attackers may take advantage of improperly setup DNS servers to launch DNS amplification attacks, which would overload the server and result in downtime. DNS cache poisoning can cause visitors to be maliciously redirected to bogus websites, aiding phishing and other assaults.

Example:

A misconfigured DNS server running on port 53 is vulnerable to DNS cache poisoning. By poisoning the DNS cache, an attacker takes use of this vulnerability to drive visitors to malicious websites.

6. Port 69 (TFTP):

Vulnerabilities:

sensitive file transmission, a lack of security safeguards, and potential unwanted access.

Effects:

Due to TFTP's lack of security mechanisms, it is susceptible to file transfers and illegal access. Attackers may take advantage of this to get access to private configuration files, which might compromise the system or grant unauthorized access to vital network resources.

Example:

Unauthorized access and file transfer are permitted by the TFTP service on Port 69, which might result in unauthorized access and the exposing of private configuration data.

7. Port 80 (HTTP):

Vulnerabilities:

SQL injection, cross-site scripting (XSS), and server configuration issues are typical online application vulnerabilities.

Effects:

Attackers may use SQL injection to alter databases, steal information, or carry out illegal operations on a web application. By injecting malicious scripts into web pages that other users are seeing, cross-site scripting (XSS) allows attackers to potentially steal data or compromise user accounts. Misconfigured servers might expose confidential data or provide unauthorized users access to system resources.

Example:

A web application running on Port 80 does not verify user input, which enables an attacker to employ a cross-site scripting (XSS) attack to steal user information and jeopardize the site's security.

8. Port 110 (POP3):

Vulnerabilities:

Weak POP3 passwords, communication that is not secured, and possible email theft.

Effects:

Attackers can get unauthorized access to users' email accounts by using brute force against weak POP3 credentials. Email content can be eavesdropped on in communications that are not encrypted. Sensitive information may be compromised or data breaches may occur as a result of unauthorized access or email theft.

Example:

Weak credentials are used to access a Port 110 email server, which provides unencrypted access. Login information can be intercepted, and emails might be accessed or stolen.

9. Port 123 (NTP):

Vulnerabilities:

Use as an amplifier in DDoS attacks or, if misconfigured, vulnerability to NTP reflection attacks.

Effects:

NTP servers can be used as a tool in Distributed Denial of Service (DDoS) attacks, which can impair service for others while magnifying the attack flow. NTP reflection attacks that use misconfigured NTP servers can potentially direct attack traffic toward the goals of the victims and overtax their resources.

Example:

A misconfigured NTP server on Port 123 will react to requests for NTP reflection from unauthorized sources, enabling a hacker to use the server's resources to intensify a DDoS assault.

10. Port 143 (IMAP):

Vulnerabilities:

IMAP credentials that are insecure, communications that are not encrypted, and a risk for email breaches.

Effects:

Users' email accounts may be hacked if their IMAP credentials are weak, enabling illegal access. Attackers may be able to intercept email content if communication is not encrypted. Email hacks or the abuse of private data can result from unauthorized access to emails.

Example:

An attacker might access users' email accounts by intercepting login information from an IMAP server on Port 143 that accepts unencrypted connections and makes use of lax authentication.

11. Port 443 (HTTPS):

Vulnerabilities:

if using web services, typical web application vulnerabilities, SSL/TLS flaws, and out-of-date encryption techniques.

Effects:

Vulnerabilities in SSL/TLS can be used to alter or intercept encrypted data, jeopardizing confidentiality. The danger of data disclosure may be increased by outdated encryption techniques' lack of security measures. If present, common online application vulnerabilities can be used to compromise web servers and allow unwanted access or data breaches.

Example:

The SSL/TLS protocol used by a web server on port 443 is out-of-date, rendering it vulnerable to known security flaws like Heartbleed that might let an attacker steal sensitive data from the server.

The unique risks and services that use those ports determine the best **Mitigation techniques** for the vulnerabilities brought on by open ports. The following are typical mitigating tactics:

- ### Strong Authentication & Access Control:

  To prevent illegal access, use robust authentication techniques. Use complicated, one-time passwords, or think about using MFA (multi-factor authentication).
  Utilize access control strategies to limit access to just valid IP addresses or people.

- ### Encryption:

  For the transfer of sensitive data, enable encryption. Use secure variations of existing protocols like SSH (Port 22) and IMAPS (Port 143) as well as SSL/TLS for online services (Port 443).

- ### Regular Patching and Updates:

  Maintain the most recent security updates on all software, including operating systems and apps, to fix known vulnerabilities.

- ### Firewalls and Intrusion Detection/Prevention Systems (IDPS):

Use firewalls to filter incoming and outgoing traffic, allowing only services that are absolutely essential and preventing any possible risks. Use IDPS to find and stop hostile activity, such as attempted intrusions.

- **Network Segmentation:**

  To lessen the effects of a security compromise, isolate important systems and networks. Segmenting networks can stop attackers from moving laterally.

- **Security Audits and Vulnerability Scanning:**

  Conduct security audits and vulnerability assessments on a regular basis to find and fix flaws in system settings and software.

- **Security Best Practices:**

  Follow security guidelines for all services using open ports. For instance, setup DNS servers (Port 53) to avoid cache poisoning and open recursion.

- **Application Security:**

  For web applications running on port 80, use safe coding standards to guard against typical vulnerabilities like SQL injection and cross-site scripting (XSS).

- **Logging and Monitoring:**

  To quickly identify and react to suspicious actions on open ports, set up logging and monitoring systems.

- **Access Controls for File Transfers (TFTP - Port 69):**

  Only allow authorised users or reliable hosts to access TFTP. Limit file transfer permissions and make sure that TFTP cannot access sensitive files.

- **DDoS Mitigation (NTP - Port 123):**

  To stop NTP reflection attacks, use DDoS mitigation techniques like rate limitation and traffic filtering.

- **Regular Backup and Recovery:**

  Keep frequent backups of important configurations and data to make recovery easier in the event of a security problem.

- **Penetration Testing and Red Teaming:**

  Red team exercises and penetration testing should be used to mimic assaults and proactively find weaknesses.

The particular services and vulnerabilities linked to each open port should be taken into account while developing mitigation techniques. To minimize risks and provide a strong defence against possible attacks, regular security assessments, monitoring, and a proactive approach to security are essential.