Phase 4

Threat Assessment

Phase 4: Threat Assessment

Scenario 1: Unpatched RDP is exposed to the internet

Description: RDP is an abbreviation of Remote Desktop Protocol which allows a remote computer to get access and take control of another computer in the network. RDP is used by many companies with employees working from home. The port that RDP uses is 3389. When RDP is unpatched and exposed to the internet, it can be a very dangerous attack vector for hackers to exploit.

Operating system/version affected: RDP is built into the Windows operating system and can be installed on macOS, Linux, and Android operating systems.

- BlueKeep attack: Windows XP, Windows Vista, Windows 7, and Windows Servers 2003 and 2008. (CVE-2019-0708) <https://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE-2019-0708>
- DejaBlue: Windows 7, Windows 8.1, Windows 10, Windows Server 2008, 2012, 2016, and 2019, and Microsoft Remote Desktop for Mac, IoS, and Andriod. (CVE-2019-1181 and CVE-2019-1182) <https://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE-2019-1181 or 1882>
- DejaBlue: Windows 10, Windows Server 2016 and 2019 (CVE-2019-1222 and CVE-2019-1226) <https://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE-2019-1222 or 1226>

Risks of attempting to exploit: Attackers could execute arbitrary code on the target machine if they successfully exploited the remote code execution vulnerability. They would remotely gain access to the target machine and could then install malicious programs that can view, change or delete any data or create a full privilege account within the machine.

Risk: RDP Brute force Attack to gain access to an exposed RDP on the network. After gaining access to an exposed RDP, the attacker can data breach the victim's machine database. Also, the attacker can download ransomware on the targeted machine and lock the victim out of the machine until the victim pays the attacker.

Tool: Nmap on Kali Linux for reconnaissance and Crowbar on Kali Linux for brute force attack

- Step 1: Use nmap to search for exposed RDP servers in the network for the specific port of 3389. -p = Scanning IP range.
    - Commandline: nmap <IP Address range> -p 3389
- Step 2: Do reconnaissance on the exposed target to see what operating system is on. -sV = Detection of Services
    - Commandline: nmap -sV <Targeted IP address>
- Step 3: Use a wordlist to perform a brute-force attack with Crowbar to find the login password. -b = target service, -u = Static name to login with, -C = multiple passwords to login with, stored in a file.. <https://www.kali.org/tools/crowbar/>

    - Commandline: ./crowbar.py –server <Target IP address> -b rdp -u Administrator -C <directory of wordlist>
- Step 4: verify the information found from the brute-force attack with xfreerdp. /u for the username, /p for the password, and /v for the IP address to which the attacker will connect.
    - Commandline: /u:Administrator /p:<found password> /v:<IP address>
- Step 5: Enter and gain access to the RDP

Remediation action:
- Disable RDP if it is not needed.
- Use a stronger password for the machine.
- Close down port 3389 with a firewall
- Have a better antivirus system that features Remote Access Shield, such as the Avast Premium Security.

CVSS Score:
- CVE-2019-0708
    - CVSS v3.0 Base Score - 9.8 CRITICAL
    - CVSS Vector - CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:P/RL:O/RC:C
-
- CVE-2019-1181, CVE-2019-1182, CVE-2019-1222, and CVE-2019-1226:
    - CVSS:3.0 Base Score - 9.8 CRITICAL
    - CVSS Vector - CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:P/RL:O/RC:C

Scenario 2: Web application is vulnerable to SQL Injection

Description: SQL injection (SQLi) is a web security vulnerability that lets the attacker interfere with the queries of an application to its database. It is an injection attack that executes malicious SQL statements that control the database server behind a web application. The attacker can bypass the web application's authentication and authorization with SQLi and gain access to the entire SQL database. After gaining access to the database, they can add, modify, or delete any record.

Operating system/version affected: Any web application or website that uses an SQL database, such as Oracle, SQL Server, or MySQL, is vulnerable to the SQL injection attack.

Risks of attempting to exploit: There are tracks that the attacker left behind when they are using SQL injection attack.
- Several invalid queries are coming from a suspicious client.
- Constant conditions in queries that always return TRUE or FALSE
- OR and UNION block in the query coding.

Risk: The consequences of a successful SQLi attack cause loss of confidentiality since a company's SQL database typically holds sensitive information. Moreover, much information in the database can be modified or deleted, which can cause much damage to a company's integrity. SQL injection attacks can be very simple, as shown below on a website login page:

- Step 1: input admin as username and abc123 as the password and try to login to the website.
    - Query: SELECT * FROM users WHERE username = 'admin' AND password='abc123'
- Step 2: SQL injection attack: input admin' OR '1' = '1 as username and try to login.
    - Query: SELECT * FROM users WHERE username = 'admin' OR '1' = '1' AND password='abc123'
        - (Username = admin and password = abc123) OR (1 = 1). Therefore, putting admin' OR '1' = '1 as username can work.
- Step 3: SQL injection attack: input admin'-- as username and try to login.
    - Query: SELECT * FROM users WHERE username = 'admin'-- AND password='abc123'
        - The query stated that after user = admin, it is all comments. If there is a username = admin, no password will be needed.

Remediation action:

- Use of prepared statements with parameterized queries.
    - All the SQL codes are defined and passed in each parameter. Therefore the attacker will not be able to change the query's intent.
- Use allowlist input validation.
    - Prevents unvalidated user input from being added to a query.
- Escape all user input.
    - A technique that escapes all user input. So the input is not confused with SQL code.
- Use of properly stored procedures.
    - Developers build SQL statements with parameters stored in the database only.

CVSS Score: CVE-2022-1731
CVSS v3.0 Base Score - 9.8 CRITICAL
CVSS Vector - CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Scenario 3: Default password on Cisco admin portal

Description: A default password is a passcode pre-configured for a device or an account when it is first set up. In this scenario, it is Cisco's admin portal. They use their brand name, "cisco", as their default username and password. This type of default login information can usually be found on the product's official website or given manuals. The devices and data under the same Cisco admin portal are at risk in this scenario. Since the default login information is out on the website, hackers can use this information to carry out an attack.

Operating system/version affected: Cisco's enterprise network function virtualization infrastructure.

Risks of attempting to exploit: By attempting to log in to the Cisco admin portal, your action will be recorded on the administrator logins event log, whether a successful or failed login.

Risk: Unwanted access to your devices and data by a cyberattack. They will have full control access to a system with root or administrative privileges. Any company's sensitive information will be leaked. Incidents such as:

- Internet Census 2012 Carna Botnet: A hacker scanned for devices online still using telnet. Since it took too long for the hacker to scan the whole internet, the hacker hacked into the online devices with default credentials without permission and created a botnet to scan for more devices. As a result, the hacker controlled 420,000 devices.
- 2013 EAS Zombie Hoax: An attack that hijacked stations in Montana and Michigan's emergency broadcast system, warning people that bodies of the dead are rising from their graves. The attack was successful because the equipment was using default credentials.

Remediation action:

- Change the default password into a unique password.
- Use alternative authentication mechanisms like multi-factor authentication.
- Restrict network access

CVSS Score: CVE-2020-3446
CVSS v3.0 Base Score - 9.8 CRITICAL
CVSS Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Scenario 4: Apache web server vulnerable to CVE-2019-0211

Description: CVE-2019-0211 is a local privilege escalation bug affecting the Apache HTTP server. This bug allows a worker in the company or anyone with access to change its privileges when the host server resets itself. From that, the privilege can be changed to the root user. Rogue servers can execute arbitrary code with root privilege, allowing the hacker to control the target machine completely.

Operating system/version affected:

- Apache: HTTP Server 2.4.17 to 2.4.38
- Canonical: Ubuntu Linux version 14.04, 16.04, 18.04, and 18.10
- Debian: Debian Linux version 9.0
- Deforaproject: Fedora versions 29 and 30
- Opensuse: Leap version 15 and 42.3

Risks of attempting to exploit: Must attempt during the restart process, where the targeted modules are shut down and restarted. That is the period that allows the privilege elevation to take place. So the attacker will need to attempt a restart for the vulnerability to work.

Risk: Since the hacker can access root privilege to the server. Anything can be changed or stolen. It can negatively impact the brand image and cause a loss of trust from the stakeholders.

Remediation action: Update to Apache 2.4.39 or a newer version.

CVSS Score: CVE-2019-0211
CVSS Base Score: 7.8
CVSS v3.0 Vector AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

Scenario 5: Web server is exposing sensitive data

Description: Data exposure is when sensitive information is lost due to unintentional exposure. It is the result of a company's action or lack of action. Data exposure usually occurs when online data is not protected and encrypted or is unintentionally updated to an incorrect database.

Operating system/version affected: Web server.

Risks of attempting to exploit: Intelligent threat detection will alert the company's IT specialist in real time.

Risk:

All data, including sensitive data, can be compromised.

- Password, credit card data, social security number, and other authentication credentials can be stolen.

Attacks that can expose sensitive data:

- SQL injection attack
- Network compromise
- Broken access control attacks
- Ransomware attacks
- Phishing attacks
- Insider threat attacks

Remediation action:

- Encrypt all sensitive data at rest
- Classify all data and apply control to that data
- Store passwords using a strong and salted hashing function.
- Use strong ciphers on the web servers
- Do not store sensitive data after it is used.
- Disable caching for the response that stores sensitive data

CVSS Score:

CVSS v3.0 Score: 9.4
CVSS v3.0 Rating: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:L
According to <https://gaya3-r.medium.com/sensitive-data-exposure-19ac6e3090f4>

Scenario 6: Web application has broken access control

Description: Broken access control is a security vulnerability that allows an unauthorized user to access restricted resources such as sensitive information or system. Broken access control is often the result of weak authentication and authorization mechanisms.

Operating system/version affected: Web application.

- It can be Tableau Server, version: 2020.4.16, 2021.1.13, 2021.2.10, 2021.3.9, 2021.4.4, and earlier

Risks of attempting to exploit: often remain unnoticed and are potentially targeted by hackers. The hacker must create a request for content or function that they cannot access and can be detected. Techniques such as injection flaws and cross-site scripting (XSS) might be needed to exploit, and these techniques can be detected.

Risk:

- Exposure of Unauthorized Content: With a successful attack, the hacker might be able to view, modify or delete sensitive data, perform unauthorized functions or even take control of the administrator account along with many user accounts. The impact of exposure to unauthorized content can hinder system performance, the company's reputation, and availability.
- Privilege Escalation: hackers can easily steal user data or deploy malicious payloads that can damage the application hosting ecosystem.
- Distributed Denial of Service: Hackers can use user accounts to deploy bots, causing the system to crash by sending numerous requests.

Remediation action:

- Enforce trusted server-side code. Attackers can not modify the access control check or metadata.
- Deny by default, deny everything else besides anything that the public should see on the web application.
- Log all failures and alert admins, and rate limit all access.
- Enforce the least privilege to all types of access accounts.
- Discretionary access control (DAC), Role-based access control (RBAC), Managed access control (MAC)

CVSS Score:

CVSS v3.1 Score: 7.7
CVSS v3.1 Rating: CVSS:3.1/AV:N/AC:H/PR:H/UI:N/S:C/C:H/I:H/A:N
According to <https://help.salesforce.com/s/articleView?id=000390544&type=1>

Scenario 7: Oracle WebLogic Server vulnerable to CVE-2020-14882

Description: CVE-2020-14882 is a remote code execution vulnerability in the Oracle WebLogic Server of Oracle Fusion Middleware. Supported versions are listed below. It is a vulnerability that bypasses the authentication and allows unauthorized hackers with HTTP network access to completely compromise the Oracle WebLogic Server. The exploit can be done with a single HTTP GET request.

Operating system/version affected:

- Oracle WebLogic Server versions 10.3.6.0.0, 12.1.3.0.0, 12.2.1.3.0, 12.2.1.4.0, and 14.1.1.0.0.

Risks of attempting to exploit: The HTTP GET request will create a file in the victim's system, located in the /tmp directory, and trigger remote code execution.

Risk: The impact of this vulnerability allows the unauthorized hacker to achieve complete control over the affected application.

Steps to exploit CVE-2020-14882
Step 1: Use tools like Kali Linux to get the victim's IP address. [ifconfig]
Step 2: Use nmap -sV <victim's IP address> to scan the victim's service version of Oracle. Suppose the version of Oracle is the version that is affected.
Step 3: Type in the URL = <victim's IP address>:7001/console/images/%252E%252E%252Fconsole.portal

Remediation action: Install the critical patch update that was released in October 2020

CVSS Score: CVE-2020-14882
CVSS v3.1 Score: 9.8
CVSS v3.1 Rating: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Scenario 8: Misconfigured cloud storage (AWS security group misconfiguration, lack of access restrictions)

Description: Amazon Web Services (AWS) is an on-demand cloud computing platform provided by Amazon. The misconfiguration occurs when a critical mishandling in cloud security leaves your company's data at risk. Making errors while configuring the security control or failing to implement them will also result in misconfiguration. According to data, misconfiguration by the customer is the number 1 cause of successful cloud-based data breaches.

Operating system/version affected: Amazon Web Services Cloud Storage.

Risks of attempting to exploit: Hackers will need to use tools such as nmap to scan open ports on the cloud IP address to see if any security group allows unrestricted access. Scanning ports with nmap will send a packet of network data to a port to check the port's status, which will leave a trail or alert the company.

Risk: Security groups that allow unrestricted access on ports 22 and 3389. Security groups should not allow port 22 traffic from or to the public internet because it is the secure shell port, a way for hackers to control an instance or a server remotely. Port 3389 is the Remote Desktop Protocol (RDP) port. This port should not allow traffic from or to the public internet either. Doing so might result in allowing hackers to compromise the RDP application, and they might potentially gain access to the organization's cloud-based data.

Remediation action: Only allowing specific IP addresses access to the secure shell port and RDP port.

1. Log in to the AWS Console and navigate to Instances services.
2. Click on the instance you want to modify its security group.
3. Navigate to the security tab and click on the security groups above inbound rules.
4. Click on edit inbound rules and look for SSH and RDP, change the source to "My IP," or input the IP address range you want the security group to have access to.

CVSS Score:

CVSS v3.1 Score: 7.5

CVSS v3.1 Rating: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Scenario 9: Microsoft Exchange Server vulnerable to CVE-2021-26855

Description: A server-side-request-forgery (SSRF) vulnerability in Microsoft Exchange Server known as ProxyLogon. A remote unauthenticated attacker can exploit this vulnerability by sending an HTTP request crafted for this vulnerable Microsoft Exchange Server. The Exchange Server needs to be able to accept untrusted connections over port 443 in order for this vulnerability to work. The attacker could authenticate to the Exchange Server if the exploitation is successful.

Operating system/version affected: Microsoft Exchange Server versions 2013, 2016, and 2019

Risks of attempting to exploit: This exploit can be detected with an exchange HttpProxy logs:

- The following directory is the location of these logs: %PROGRAMFILES%\Microsoft\Exchange Server\V15\Logging\HttpProxy
- The exploitation can be found by searching for log entries where AuthenticatedUser = empty and AnchorMailbox = pattern of ServerInfo~*/*
    - Example: Import-Csv -Path (Get-ChildItem -Recurse -Path "$env:PROGRAMFILES\Microsoft\Exchange Server\V15\Logging\HttpProxy" -Filter '*.log').FullName | Where-Object { $_.**AnchorMailbox -like 'ServerInfo~*/*'** -or $_.BackEndCookie -like 'Server~*/*~*'} | select DateTime, AnchorMailbox, UrlStem, RoutingHint, ErrorCode, TargetServerVersion, BackEndCookie, GenericInfo, GenericErrors, UrlHost, Protocol, Method, RoutingType, AuthenticationType, ServerHostName, HttpStatus, BackEndStatus, UserAgent
- Can use the application logs in the AnchorMailbox path to determine what actions were taken.
    - These logs are located in the %PROGRAMFILES%\Microsoft\Exchange Server\V15\Logging directory.

Source: <https://www.microsoft.com/en-us/security/blog/2021/03/02/hafnium-targeting-exchange-servers/>

Risk: The exploit is an authentication bypass by performing pre-authentication SSRF. This vulnerability can leak sensitive information onto the internal network and allow hackers to download user emails and gain full access to the mail server.

By using the Burp Suite, the hacker can perform the exploitation
Source: <https://www.safe.security/resources/blog/microsoft-exchange-ssrf/>

1. Load the login page and intercept the request.
2. Modify the HTTP request by adding the Burp Collaborator URL in place of the value present in the Cookie Header “X-AnonResource-Backend”, then send the request.
3. Both HTTP and DNS requests receive a callback at the Burp Collaborator. It can be observed that it discloses some internal information related to the Exchange Server.

4. As observed below, an SSRF was executed, and it could communicate with the external server to successfully retrieve the data.

Remediation action:

- Install the mandatory patches for the affected version.
- Place the Exchange Server inside a VPN to separate port 443 from external connection requests.

CVSS Score: CVE-2021-26855
CVSS v3.1 Score: 9.1
CVSS v3.1 Rating: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H