|  |  |
| --- | --- |
| Penetration Test Report Prepared for Hotel Dorsey | Corporate Logo Haverbrook security lab corporate logo  Name: Jordan Lee  Team Number: Red Team 2  Student Number: Student Number 3 |

**Introduction:**

This report documents the penetration testing performed on Hotel Dorsey's information technology system. It is a follow-up to the system scan requested earlier, which detected/showed critical vulnerabilities in Hotel Dorsey's network. That initial system scan also revealed 20 open ports on their network, leading to multiple critical vulnerabilities. The focus of the penetration test was to gain access to the root through an open port, port 80 HTTP. The hotel system allows customers and staff to access files and functionality through an online booking system. Users create usernames and passwords for easy access, allowing reservations from anywhere and staff to confirm and book rooms before check-in. This penetration test aims to prove that Hotel Dorsey's network can be hacked through port 80 (HTTP) and steal data off the network. To conduct this test, we use Nmap, Metasploit, MySQL, ping, various terminal commands, and John the Ripper.

|  |  |
| --- | --- |
| **Tool** | **Purpose** |
| Nmap | Open-source network scanning tool – used to scan the system for open ports and their related vulnerability. |
| Metasploit | Used to exploit the open ports. |
| MySQL | Used to access target database. |
| Ping Command | Used to test connectivity with victim machine. |
| John the Ripper | Used to decode the password hashes. |

**Table 1: Tools used in the penetration test.**

**Target:**

When we conducted this penetration test, we used Nmap to scan the target system to confirm that those 20 ports were open from the initial vulnerability scan. As shown in Figure 1, you can see the hostname and IP address of the target machine, which is metasploitable and 10.2.3.100. As shown in Figure 2, you can see that the IP address for the attacking machine is 10.2.3.50, operating on Kali Linux. Table 2 shows all 20 open ports on the target/victim machine that can be exploited, but once again, we will be exploiting port 80 HTTP for this penetration test.

A screenshot of a computer

Description automatically generated

**Figure 1: Target machine OS and IP address**



**Figure 2: Attack machine IP address**

A screen shot of a computer

Description automatically generated

**Figure 3: 20 open ports on the target machine**

|  |  |  |
| --- | --- | --- |
| **Machine** | **Open Port Numbers** | **Function** |
| **Attack Machine: Kali Linux**  **IP Address: 10.2.3.50** | No open ports | N/A |
| **Target Machine: Metasploitable**  **IP Address: 10.2.31.00** | **Port 21** | File Transfer Protocol – used to transfer files between client and server. |
|  | **Port 22** | (SSH) Secure Shell, used for secure loin, file transfers, and port forwarding. |
|  | **Port 23** | This port is designated for Telnet, one of the oldest Internet protocols. It is used to |
|  | **Port 25** | This port is used for (STMP) Simple Mail Transfer Protocol. The port is used for email servers, so if the company doesn't host one, it should not be used. |
|  | **Port 53** | (DNS) Domain name service is used for domain name resolution. The hotel should not open this port if it uses a DNS server due to potential attacks. |
|  | **Port 80** | (HTTP) Hyper Text Transfer Protocol. This port should be disabled unless the hotel is running any web services. |
|  | **Port 111** | (RPC) used by the portmapper service to map Remote Procedure Call programs and versions to specific transport port numbers. |
|  | **Port 139** | This port is used for NetBIOS. A protocol used for File and Print Sharing under all current versions of Windows. |
|  | **Port 445** | Is used for direct TCP/IP Microsoft Networking access without the need for a NetBIOS layer. Leaving port 445 open leaves Windows machines vulnerable to trojans and worm attacks |
|  | **Port 512** | Automatic authentication using privileged port numbers takes place on this port. |
|  | **Port 513** | Automatic authentication using privileged port numbers takes place on this port. |
|  | **Port 514** | Interactive shell utilizes this port. |
|  | **Port 1099** | RMIRegistry, a service that establishes and launches a remote object registry, uses this port. |
|  | **Port 1524** | The Ingres database, a SQL database, uses this port; however, hackers primarily use it for Distributed Denial of Service (DDoS) trojans and backdoor viruses. |
|  | **Port 2049** | (NFS) Network File System. This is used for remote filesystem access. |
|  | **Port 3306** | This port is used by MySQL database server connections |
|  | **Port 5432** | PostgreSQL Database Server, an object-relational database management system. |
|  | **Port 6667** | IRC protocol, text-based chat service |
|  | **Port 8009** | Tomcat is a Java HTTP web server environment that enables Java code to execute, and the Apache Tomcat service uses this port. |
|  | **Port 8180** | Unknown – unassigned port |

**Table 2: Open ports on target machine with associated services**

**Vulnerability:**

A screenshot of a computer

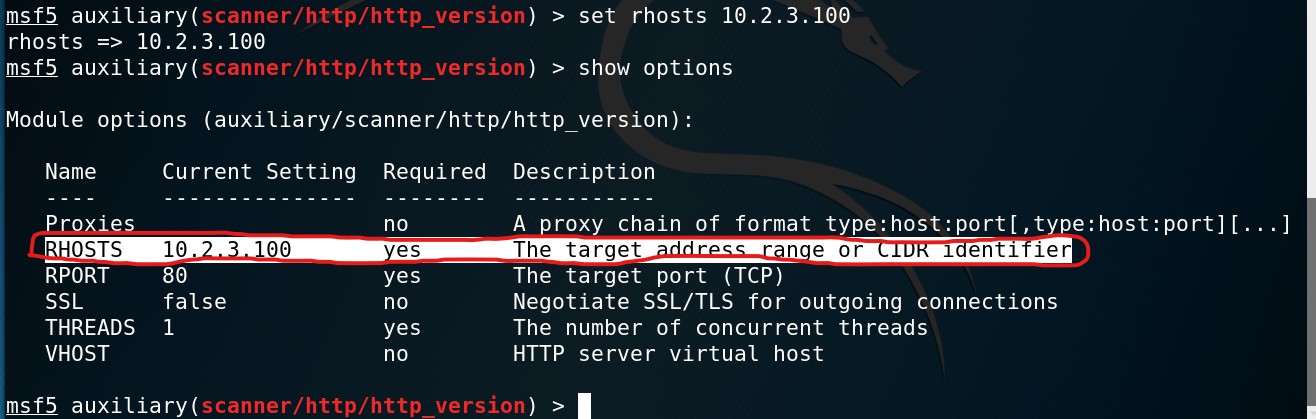
Description automatically generated

**Figure 4: Port 80 open, utilizing Apache httpd version 2.2.8 on Ubuntu**

A computer screen shot of a road

Description automatically generated

**Figure 5: HTTP version that is being used**



**Figure 6: Remote host was created in the module**

**A computer screen shot of a computer

Description automatically generated**

**Figure 7: Victim machine IP address was imputed into the exploit.**

**A computer screen with white text

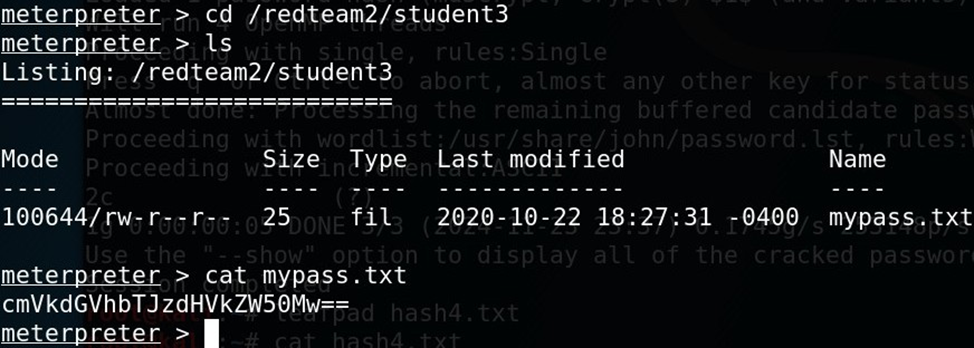
Description automatically generated**

**Figure 8: Obtained access to the target machine through Meterpreter**

I successfully exploited Port 80 (HTTP) to gain unauthorized access to the remote system and retrieve files from the root directory. Port 80, typically used for web server communications, hosted a vulnerable service that allowed me to execute this attack. The vulnerability likely stemmed from an outdated or misconfigured web server or web application, which exposed the system to potential exploitation. By leveraging this weakness, I was able to escalate my privileges and achieve root-level access, granting complete control over the victim machine. This highlights the critical importance of maintaining up-to-date server software and implementing robust security configurations to mitigate such risks.

I was able to gain access to the target machine via meterpreter, which is a post-exploit tool from the Metasploit framework. I used it to navigate and interact with the compromised target's file system. I used the "cd" command to change the directory and navigated to the "/redteam2/student 3" file. I then used the "ls" command to see the contents of the file. The file name "mypass.txt" was in the directory, and after using the "cat" command, I was able to read its contents. There was a base-64 encoded hash, so I used John the Ripper to crack that encoded hash. The use of meterpreter allowed me to extract sensitive information from the compromised target system, starting with exploiting Port 80 HTTP.

**Data Exfiltration:**



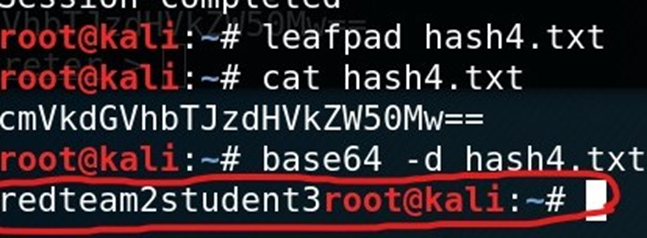
**Figure 9: Directory view of cd /redteam2/student3 file**



**Figure 10: Password cracked with John The Ripper; 2C**



**Figure 11: The password hash assigned to me.**



**Figure 12: Password has been cracked**

When it comes to data exfiltration, it is critical to understand where the data is stored, the file structure, and the commands to use. I was able to access the shadow file that holds the encrypted password that I needed to obtain. The shadow file was located in the "redteamlookhere" directory. I conducted the penetration test by exploiting port 80 HTTP, giving me direct access to the root. Losing proprietary data due to exploiting vulnerabilities, such as an HTTP service on port 80, can have severe implications for a company. Proprietary data often includes trade secrets, customer information, product designs, and intellectual property that provide a competitive edge. If an attacker gains unauthorized access through an exposed HTTP port, they could exfiltrate sensitive data, leading to financial losses, reputational damage, and loss of customer trust.

Costs associated with such a breach can include legal fees, regulatory fines, customer compensation, and investments in remediation and cybersecurity enhancements. Additionally, exploiting a web service vulnerability may highlight inadequate security controls, increasing scrutiny from stakeholders and regulatory bodies. This underscores the critical need for rigorous vulnerability management, including secure configurations, regular patching, and monitoring exposed services.

**Recommendations:**

To remediate the vulnerability exploited on Port 80 (HTTP), several security controls and measures should be implemented to prevent future exploitation and improve the client's overall security posture. The client should ensure that all web servers and web application software are updated to the latest versions, as outdated or unpatched software often contains known vulnerabilities that attackers can exploit. Regular patch management should be enforced to close security gaps promptly. Hotel Dorsey should implement secure configuration practices. This includes disabling unnecessary services and ports, restricting directory access permissions, and enforcing least privilege principles for accounts used by web servers.

I recommend that Hotel Dorsey implement a Web Application Firewall (WAF) to help mitigate threats by filtering and monitoring HTTP traffic and blocking malicious requests before they reach the server. Additionally, secure coding practices should be adopted to reduce vulnerabilities in the application layer, such as injection attacks or mismanagement of authentication mechanisms. The client should also conduct regular vulnerability scans and penetration tests to identify potential weaknesses before exploiting them.

Finally, training employees and developers on cybersecurity best practices and ensuring adherence to these standards can significantly enhance the organization's ability to prevent and mitigate threats. By addressing these recommendations, the organization can eliminate the exploited vulnerability on Port 80 and reduce the likelihood of similar incidents in the future.

**References:**

[1] Chapple, M., & Seidl, D. (2023). Chapter 7 Analyzing Vulnerability Scans. In CompTIA CySA+ Study Guide Exam CS0-003 (3rd ed., pp. 245–284). essay, John Wiley & Sons, Inc.

[2] Chapple, M., & Seidl, D. (2023). Chapter 8 Responding to Vulnerabilities. In CompTIA CySA+ Study Guide Exam CS0-003 (3rd ed., pp. 293–333). essay, John Wiley & Sons, Inc.

[3] Chapple, M., & Seidl, D. (2023). Chapter 8 Responding to Vulnerabilities. In CompTIA CySA+ Study Guide Exam CS0-003 (3rd ed., pp. 293–333). essay, John Wiley & Sons, Inc.

[4] IT Convergence. (2023, March 9). 5 security issues from using an unsupported operating system. <https://www.itconvergence.com/blog/how-safe-are-your-unsupported-legacy-operating-systems-and-software/>

[5] "Chapter 5: Implementing an Information Security Vulnerability Management Process", Pearson CompTIA Cybersecurity Analyst (CySA+), 2020. [Online]. Available: <https://www.ucertify.com/>.

[6] Nmap. (n.d). Zenmap – Official cross-platform Nmap Security Scanner GUI. Nmap.Org

Retrieved November 26, 2024, from <https://nmap/org/zenmap/>

[7] Metasploitable 2 exploitability guide. Metasploitable 2 Exploitability Guide | Metasploit Documentation. (n.d.). <https://docs.rapid7.com/metasploit/metasploitable-2-exploitability-guide/>