IronShield Cyber Defense

Empire Company
Security Assessment Findings Report

Business Confidential

Date: December 20th, 2024
Project: EC-001
Version 1.0

Table of Contents

Confidentiality Statement

This document is the exclusive property of Empire Company (EC) and IronShield Cyber Defense. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or part, in any form, requires consent of both EC and IronShield Cyber Defense.

IronShield Cyber Defense may share this document with auditors under non-disclosure agreements to demonstrate penetration test requirement compliance.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. IronShield Cyber Defense prioritized the assessment to identify the weakest security controls an attacker would exploit. IronShield Cyber Defense recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

Contact Information

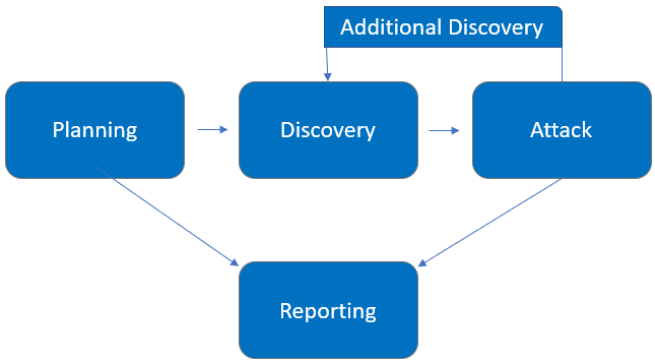
Name	Title	Contact Information
Demo Company		
John Smith	VP, Information Security (CISO)	Office: (555) 555-5555 Email: john.smith@demo.com
Jim Smith	IT Manager	Office: (555) 555-5555 Email: jim.smith@demo.com
Joe Smith	Network Engineer	Office: (555) 555-5555 Email: joe.smith@demo.com
IronShield Cyber Defense		
Bryan Vega	Lead Penetration Tester	Office: (555) 555-5555 Email: b.vega@gmail.com

Assessment Overview

From December 20th, 2024 to December 22nd, 2024, EC engaged IronShield Cyber Defense to evaluate the security posture of its infrastructure compared to current industry best practices that included an external penetration test. All testing performed is based on the NIST *SP 800-115 Technical Guide to Information Security Testing and Assessment, OWASP Testing Guide (v4), and customized testing frameworks*.

Phases of penetration testing activities include the following:

- Planning – Customer goals are gathered and rules of engagement obtained.
- Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.



Assessment Components

External Penetration Test

An external penetration test emulates the role of an attacker attempting to gain access to an internal network without internal resources or inside knowledge. An IronShield Cyber Defense engineer attempts to gather sensitive information through open-source intelligence (OSINT), including employee information, historical breached passwords, and more that can be leveraged against external systems to gain internal network access. The engineer also performs scanning and enumeration to identify potential vulnerabilities in hopes of exploitation.

Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V3 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Moderate	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Scope

Assessment	Details
External Penetration Test	10.21.21.11

Scope Exclusions

Per client request, IronShield did not perform any of the following attacks during testing:

- Denial of Service (DoS)
- Phishing/Social Engineering

All other attacks not specified above were permitted by Empire Company.

Client Allowances

The client provided the IP address of the machine to perform a penetration test.

Executive Summary

IronShield Cyber Defense evaluated EC's external security posture through an external network penetration test from December 20th, 2024, to December 22nd, 2024. By leveraging a series of attacks, IronShield Cyber Defense found critical level vulnerabilities that allowed full internal network access to the EC headquarter office. It is highly recommended that EC address these vulnerabilities as soon as possible as the vulnerabilities are easily found through basic reconnaissance and exploitable without much effort.

Scoping and Time Limitations

Scoping during the engagement did not permit denial of service or social engineering across all testing components.

Time limitations were in place for testing. Internal network penetration testing was permitted for three business days

Testing Summary

IronShield conducted a penetration test on the target machine and identified six critical and high-severity vulnerabilities that led to the system's compromise and pivoting into the internal network hosting the Active Directory environment. The findings are summarized below:

The FTP service on port 21 (Finding EPT-001) allowed anonymous access without requiring authentication. IronShield exploited this configuration using the Metasploit Framework, leveraging the vsftpd backdoor exploit to gain unauthorized access. It exposed the server to unauthorized access, data exfiltration, and potential misuse for further attacks.

The HTTP service on port 80 was running an outdated PHP version (5.2.4), which is vulnerable to a PHP CGI Argument Injection (Finding EPT-002). IronShield exploited this vulnerability using the Metasploit Framework to gain a Meterpreter session on the target system. Allowed remote code execution, providing full system access.

The SSH service on port 22 (Finding EPT-003) was vulnerable to brute force attacks due to weak credentials. IronShield used the Hydra tool with a wordlist to successfully crack the login credentials and gain access. Unauthorized access to the system via SSH enabled further exploitation.

The Telnet service on port 23 (Finding EPT-004) displayed a banner revealing default credentials. IronShield used these credentials to log in and gain shell access to the system. Enabled unauthorized access due to the use of weak or default authentication mechanisms.

The Samba service on ports 139 and 443 (Finding EPT-005) was exploited using the Metasploit Framework's usermap_script module. IronShield successfully executed the exploit and obtained a shell on the target system. Allowed remote command execution and unauthorized access to the system.

The testing demonstrated significant vulnerabilities in the target system, which were exploited to gain

root access, compromised the machine. These findings highlight the importance of securing services, enforcing strong authentication, and maintaining up-to-date software to mitigate the risks of exploitation and lateral movement. For further information on findings, please review the External Penetration Test Findings section.

Tester Notes and Recommendations

Testing results of the Empire Company network are indicative of an organization undergoing its first penetration test, which is the case here. The findings reveal several critical and high-severity vulnerabilities that pose significant risks to the organization's security posture. These issues highlight the need for a comprehensive approach to strengthening the security of Empire Company's infrastructure. Below are detailed notes and recommendations:

General Observations

1. **Weak Authentication Mechanisms**: Several services were found using default or weak credentials, which enabled unauthorized access.
2. **Outdated Software**: Critical services were running outdated versions, exposing them to known vulnerabilities.
3. **Misconfigured Services**: Services like FTP, Telnet, and Samba were misconfigured, providing attack vectors for exploitation.
4. **Lack of Network Segmentation**: Once the perimeter was breached, pivoting to the internal network and enumerating Active Directory was straightforward, suggesting insufficient segmentation.

Recommendations

1. Authentication and Access Control
    a. **Enforce Strong Password Policies**: Implement and enforce password complexity requirements, including length, character variety, and regular expiration.
    b. **Disable Anonymous and Default Credentials**: Disable anonymous access for FTP and ensure no services are using default credentials.
    c. **Implement Multi-Factor Authentication (MFA)**: Wherever possible, enforce MFA for critical services like SSH and administrative accounts.
2. Software and Patch Management
    a. **Update and Patch Systems**: Ensure all software, operating systems, and services are updated to the latest secure versions. Address vulnerabilities like PHP CGI Argument Injection by updating PHP to a supported version.
    b. **Conduct Regular Vulnerability Scans**: Regularly scan the network for outdated software and vulnerabilities to identify and remediate issues proactively.
3. Service Configuration
    a. **Harden Service Configurations**: Disable unnecessary services like Telnet and replace them with secure alternatives like SSH.

b. **Secure Default Settings**: Configure services with secure defaults, such as limiting banners that disclose sensitive information.

4. **Network Segmentation and Monitoring**
   a. **Implement Network Segmentation**: Separate critical systems, such as Active Directory, from less secure areas of the network to limit lateral movement.
   b. **Deploy Intrusion Detection/Prevention Systems (IDS/IPS)**: Use tools like Snort or Suricata to monitor and block suspicious activities.

Key Strengths and Weaknesses

The following identifies the key strengths identified during the assessment:

1. N/A

The following identifies the key weaknesses identified during the assessment:

1. Password policy found to be insufficient
2. Outdated and Vulnerable Software
3. Lack of Network Segmentation
4. Exposure of Sensitive Information
5. Unsecured Remote Access Services
6. Lack of Monitoring and Detection
7. Misconfigured Services

Vulnerabilities by Impact

The following tables illustrate the vulnerabilities found by impact and recommended remediations:

External Penetration Test Findings

3	3	0	0	1
Critical	High	Moderate	Low	Informational

Finding	Severity	Recommendation
External Penetration Test		
EPT-001: Anonymous FTP Access	Critical	Disable Anonymous FTP Access
EPT-002: PHP CGI Argument Injection (CVE-2007-4562)	Critical	Upgrade PHP, patch the system and deploy a Web Application Firewall (WAF)
EPT-006: Pivot to Active Directory	Critical	Implement network segmentation and restrict access to critical systems. Monitor for lateral movement using endpoint detection and response (EDR) tools.
EPT-003: SSH Service Vulnerable to Brute Force Attack	High	Enforce Strong Authentication and Password Policy
EPT-004: Telnet Service with Default Credentials and Banner Disclosure	High	Disable Telnet
EPT-005: Samba Remote Command Execution	High	Restrict Access to SMB Services, use Strong Authentication, Strong Password Policy and Disable Default Credentials
EPT-006: Steps to Compromise Machine	Informational	Review action and remediation steps.

External Penetration Test Findings

Finding EPT-001: Anonymous FTP Access (Critical)

Description:	The FTP service on port 21 is configured to allow anonymous access, enabling anyone to log in using "**anonymous**" as the username with no password required. This configuration exposes the server to unauthorized access, data exfiltration, and potential misuse for malicious purposes. Such a setup significantly increases the risk of sensitive data leakage and can serve as a platform for launching further attacks on the network. IronShield, exploited this vulnerability through the Metasploit Framework, leveraging the vsftpd backdoor exploit to gain unauthorized access and demonstrate potential system compromise.
Impact:	Critical
System:	10.21.21.11
References:	DigitalOcean Guide to Securing FTP: https://www.digitalocean.com/community/tutorials/how-to-secure-an-ftp-server OpenSSH SFTP Documentation: https://www.openssh.com/manual.html

Exploitation Proof of Concept

IronShield demonstrated a proof of concept was conducted using Metasploit Framework to target the vsftpd service running on port 21. By deploying the vsftpd_234_backdoor module, we successfully gained unauthorized shell access on the target system. This demonstrates how an attacker can exploit a known vulnerability in vsftpd to compromise the server and potentially gain control over sensitive data.

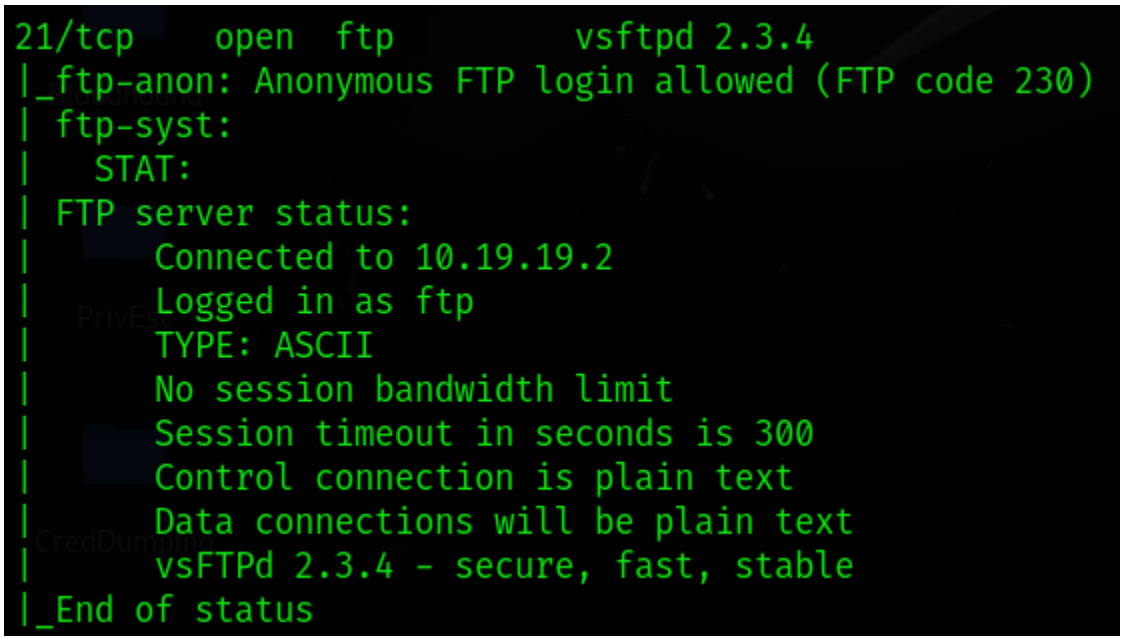


Figure 1: Nmap Scan of FTP showing anonymous login allowed

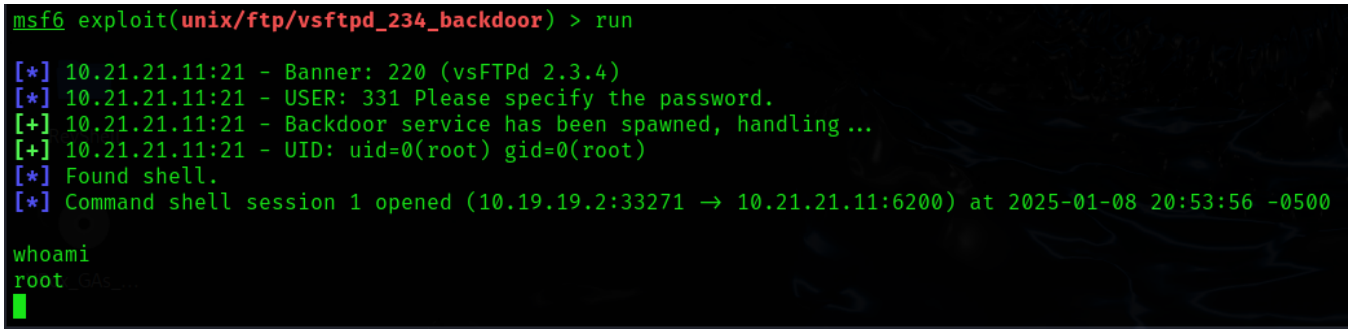


Figure 2: Exploiting FTP through Metasploit Framework

Remediation

Who:	System Administrator or Network Administrator
Vector:	Network-based (FTP Port 21)
Action:	• **Disable Anonymous FTP Access**: Immediately reconfigure the FTP service to disable anonymous logins. This can be done by modifying the FTP server configuration file (e.g., /etc/vsftpd.conf for vsftpd) to ensure anonymous_enable=NO. • **Enforce Strong Authentication**: Implement strong authentication mechanisms for FTP access, requiring valid user credentials. • **Review and Limit FTP Access**: Restrict FTP access to necessary users and internal networks only, using firewall rules to limit exposure. • **Monitor and Audit Access**: Regularly monitor and audit FTP server logs for any unauthorized access attempts and ensure compliance with security policies.

Finding EPT-002: PHP CGI Argument Injection (CVE-2007-4562) (Critical)

Description:	IronShield identified an open HTTP service on port 80 and determined through analysis that the server was running PHP version 5.2.4. This version is vulnerable to a PHP CGI Argument Injection vulnerability, which was exploited using Metasploit Framework. By leveraging this exploit, IronShield gained access to the system via a Meterpreter session, demonstrating the risk posed by outdated and vulnerable software. Which leads to performing a privilege escalation on the machine.
Impact:	Critical
System:	10.21.21.11
References:	https://www.rapid7.com/db/modules/exploit/multi/http/php_cgi_arg_injection/

Exploitation Proof of Concept

IronShield discovered that the HTTP service on port 80 was running PHP version 5.2.4, a vulnerable version susceptible to PHP CGI Argument Injection. Using Metasploit Framework, they executed the exploit/multi/http/php_cgi_arg_injection module. Upon exploitation, IronShield successfully gained a Meterpreter session, demonstrating remote code execution on the vulnerable system.

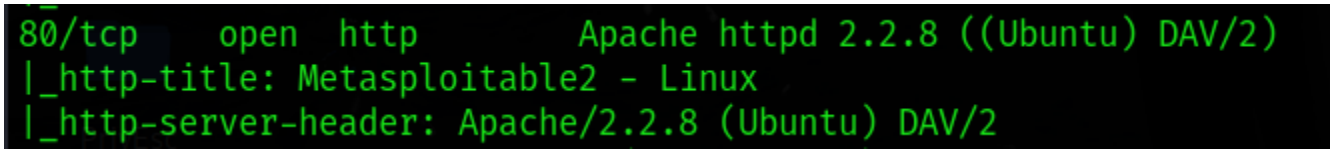


Figure 3: Nmap Scan of Port 80

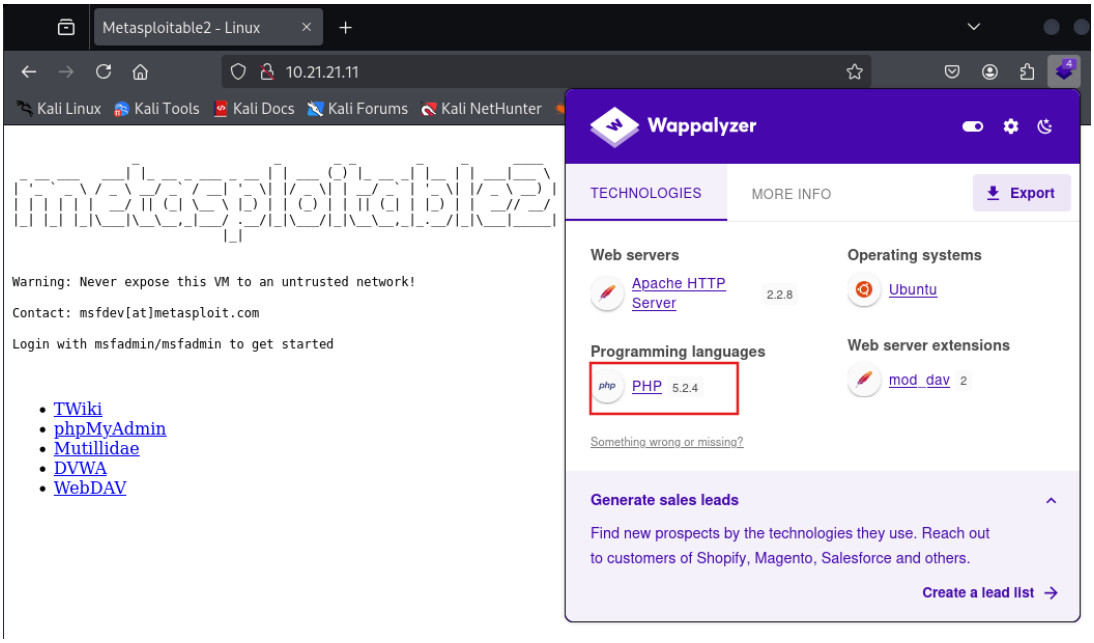


Figure 4: PHP version identified using Wappalyzer

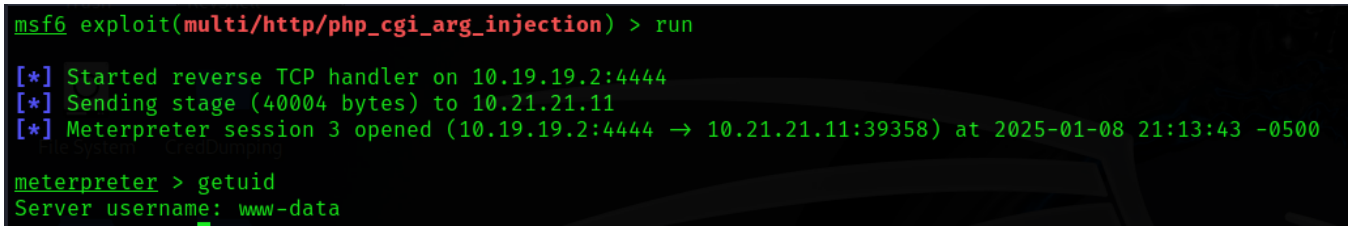


Figure 5: Exploiting PHP through Metasploit Framework

Remediation

Who:	System Administrator
Vector:	The attack vector is an **unpatched PHP CGI Argument Injection vulnerability** in the HTTP service running on port 80, which allows remote attackers to execute arbitrary code on the server.
Action:	• **Upgrade PHP**: Immediately update PHP to a supported and secure version, ideally one that is not vulnerable to CGI argument injection (e.g., PHP 5.3.13 or later). • **Patch the System**: Apply any available security patches for PHP and other web server software to mitigate known vulnerabilities. • **Disable PHP as CGI**: Configure the server to run PHP as a module rather than as a CGI to reduce the attack surface. • **Input Validation**: Implement strict input validation to prevent injection attacks and sanitize user inputs. • **Web Application Firewall (WAF)**: Deploy a WAF to detect and block malicious payloads targeting this vulnerability. • **Monitor and Audit**: Continuously monitor logs for any signs of exploitation and conduct regular vulnerability assessments to identify and mitigate risks.

Finding EPT-003: SSH Service Vulnerable to Brute Force Attack (High)

Description:	The Secure Shell (SSH) protocol provides secure remote access and data transfer over an unsecured network. However, if weak or easily guessable credentials are used, the service becomes vulnerable to brute force attacks, where an attacker systematically attempts to crack usernames and passwords. IronShield used the Hydra tool to perform a brute force attack. By leveraging a wordlist of potential usernames and passwords, IronShield successfully cracked the credentials and gained unauthorized access to the SSH service, demonstrating the risk posed by weak authentication mechanisms.
Impact:	High
System:	10.21.21.11
References:	NIST Cybersecurity Framework: https://www.nist.gov/cyberframework OpenSSH Hardening Guide: https://www.ssh.com/academy/ssh/hardening

Exploitation Proof of Concept

Using the Hydra tool, IronShield performed a brute force attack against the SSH service running on port 22 of the target system by providing a wordlist of usernames and passwords. This attack successfully identified valid credentials, allowing IronShield to establish an SSH connection and gain unauthorized access to the target system.

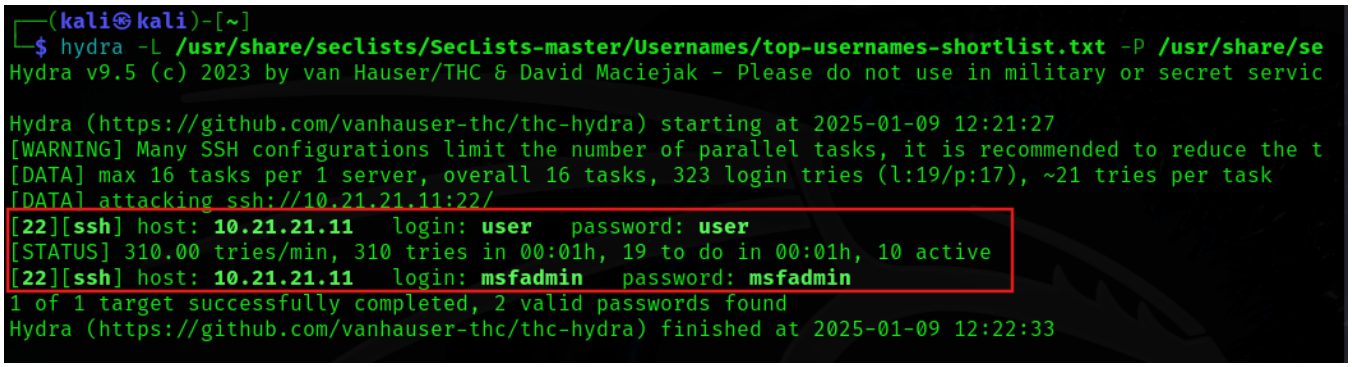


Figure 6: Brute Force with Hydra tool

Remediation

Who:	System Administrator or Network Administrator
Vector:	The attack vector is an **open SSH port (22)** with weak or easily guessable credentials, exploited via brute force.
Action:	• **Enforce Strong Authentication**: Implement strong, complex passwords and consider using key-based authentication instead of password-based login. • **Enable Account Lockout Policies**: Configure the system to lock accounts temporarily after a certain number of failed login attempts. • **Restrict Access**: Limit SSH access to specific IP addresses or networks using firewall rules. • **Use Multi-Factor Authentication (MFA)**: Add an additional layer of security to SSH logins. • **Monitor and Alert**: Continuously monitor login attempts and set up alerts for suspicious activity. • **Change Default Configurations**: Disable root login over SSH and ensure the PermitRootLogin directive is set to "no" in the SSH configuration file. • **Update and Harden Systems**: Regularly update the SSH service and ensure all security patches are applied.

Finding EPT-004: Telnet Service with Default Credentials and Banner Disclosure (High)

Description:	IronShield identified an open Telnet port (23) on the target system and used the Telnet protocol to connect to it. By analyzing the banner information displayed during the connection, they retrieved the default username and password. Using these credentials, IronShield successfully logged into the system and gained shell access, demonstrating the risk of using default or weak authentication with Telnet.
Impact:	High
System:	10.21.21.11
References:	SSH Banner Configuration: https://linux.die.net/man/5/sshd_config NIST Password Guidelines: https://pages.nist.gov/800-63-3/

Exploitation Proof of Concept

Using the Telnet client, IronShield connected to the target system's open Telnet port (23). Upon connecting, the system displayed a banner revealing the default username and password. By entering these credentials, IronShield successfully gained shell access, demonstrating how default credentials can be exploited to compromise the system.

Figure 7: Banner Disclosure and Default Credentials on Telnet

Remediation

Who:	System Administrator
Vector:	The attack vector is an **open Telnet port (23)** with default or weak credentials, which allowed unauthorized access to the system.
Action:	• **Disable Telnet**: Replace Telnet with a secure protocol like SSH, which encrypts communication. • **Remove Default Credentials**: Change all default usernames and passwords to strong, unique credentials. • **Restrict Access**: Use a firewall to block Telnet access or restrict it to trusted IPs if absolutely necessary. • **Encrypt Communications**: If Telnet must be used, implement a VPN or other encryption mechanism to secure the connection. • **Monitor and Audit**: Continuously monitor logs for unauthorized Telnet access attempts and perform regular audits of system configurations. • **Update Systems**: Ensure the system and services are patched and up to date to address any known vulnerabilities.

Finding EPT-005: Samba Remote Command Execution (High)

Description:	IronShield identified an open Samba service running on ports 139 and 443, which are commonly used for file and printer sharing via the SMB protocol. They exploited a vulnerability in Samba using the Metasploit Framework with the usermap_script exploit module, gaining unauthorized access to the target system by executing the exploit. After successful exploitation, they obtained a shell on the target system, demonstrating the risk of improperly secured Samba services.
Impact:	High
System:	10.21.21.11
References:	Samba Documentation: https://www.samba.org/samba/docs/ Official Samba Security Updates: https://www.samba.org/samba/security/

Exploitation Proof of Concept

IronShield identified an open Samba service on ports 139 and 443 and used the Metasploit Framework to exploit it. They ran the usermap_script module and executed the exploit. This resulted in successful shell access to the target system, demonstrating vulnerability in the Samba service.

```
msf6 exploit(multi/samba/usermap_script) > run

[*] Started reverse TCP handler on 10.19.19.2:4444
[*] Command shell session 1 opened (10.19.19.2:4444 → 10.21.21.11:36618) at 2025-01-09 12:30:20 -0500

uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
id
uid=0(root) gid=0(root)
```

Figure 8: Exploiting SMB through Metasploit Framework

Remediation

Who:	System Administrator
Vector:	The attack vector is the **Samba service** running on ports 139 and 443, which is vulnerable to exploitation through the usermap_script module in Metasploit, allowing attackers to gain unauthorized access to the system.
Action:	• **Update Samba**: Immediately update Samba to the latest stable version to patch known vulnerabilities and mitigate the risk of exploitation. • **Disable Unnecessary SMB Ports**: If not required, disable SMB on ports 139 and 443 to reduce the attack surface. • **Restrict Access to SMB Services**: Implement firewall rules to restrict access to the Samba service, allowing only trusted IP addresses or networks. • **Use Strong Authentication**: Configure Samba to require strong authentication methods and disable any weak or default credentials. • **Monitor and Audit**: Continuously monitor Samba logs for unusual activity and conduct regular vulnerability assessments to ensure the system remains secure.

Finding IPT-006: Steps to Compromise (Informational)

The steps below describe how the penetration tester obtained domain administrator access. Each step also provides remediation recommendations to help mitigate risk.

Step	Action	Remediation
1	Enumerate Services. Use nmap to identify open ports and services.	Use a firewall to restrict unnecessary ports and services. Regularly scan and monitor the network for open ports.
2	Exploit FTP (Port 21). Log in with anonymous credentials to gain FTP shell access.	Disable anonymous FTP access. Use strong authentication methods and enforce secure file transfer protocols (e.g., SFTP).
3	Exploit SSH (Port 22). Use Hydra to brute force SSH credentials.	Implement account lockout policies, use strong passwords, and enable SSH key-based authentication.
4	Exploit Telnet (Port 23). Log in using default credentials revealed in the Telnet banner.	Disable Telnet and replace it with secure alternatives like SSH. Remove default credentials and banners.
5	Exploit HTTP (Port 80). Use Metasploit to exploit PHP CGI Argument Injection (CVE-2007-4562).	Update PHP to the latest version. Regularly patch and monitor web applications for vulnerabilities.
6	Exploit Samba (Ports 139/443). Use Metasploit to execute commands via the usermap_script exploit.	Update Samba to a secure version. Restrict SMB access to trusted hosts and implement network segmentation.

Remediation

Review action and remediation steps.

IronShield Cyber Defense

Last Page