AUGUST 3, 2026

HACKER ACADEMY LTD

PENETRATION TEST AND VULNERABILITY ASSESSMENT REPORT

FOR EXAMPLE COMPANY LTD

HACKER ACADEMY LTD
https://hackeracademy.uk

1 TABLE OF CONTENTS

This page intentionally left blank

1 EXECUTIVE SUMMARY

EXAMPLE COMPANY LTD engaged Hacker Academy Ltd, a CREST-accredited Penetration Testing Provider, to perform penetration testing of the web applications, the APIs, Internet-facing servers and internal infrastructure. The primary goal of this Grey and Black box penetration testing project was to identify any potential areas of concern associated with the system in its current state and determine the extent to which the system may be breached by an attacker possessing a particular skill and motivation.

The project has been meticulously planned and executed based on the guidelines set forth by CREST's pentest standard. The testing methodology, procedures, and techniques used were carefully selected to ensure that all vulnerabilities were thoroughly assessed and reported. The final report was prepared in compliance with CREST's requirements, providing detailed findings and recommendations for remediation.

Apart from A guide for running an effective Penetration Testing programme by CREST, the assessment was performed in accordance with the best-in-class practices as defined by ISECOM's Open-Source Security Testing Methodology Manual (OSSTMM), Open Web Application Security Project (OWASP) and Penetration Test Execution Standard (PTES).

Penetration Test and Security Assessment, which was conducted over a **10 workdays** period between **April 20th and May 4th, 2026**, was based on the scope of work detailed in the ***Scope of Engagement*** section listed below.

Multiple tests and audits were conducted against the targeted systems involving both automated scanning and manual testing, in order to identify a range of possible vulnerabilities and any potential misconfiguration of the systems.

The assessment identified **11** findings across the three testing areas, including **2 Critical** and **5 High severity issues**. The overall security posture of the in-scope environment is assessed as **CRITICAL RISK**, driven primarily by two findings that, individually, could each result in a complete compromise of a critical system.

Key Highlights:

Web Application: A critical SQL Injection vulnerability was identified on example.com, potentially allowing unauthorised access to the underlying database. Additionally, the web server was running an outdated, exploitable version of Nginx, and supported deprecated SSL/TLS protocols.

External Infrastructure: Unnecessary ports (LDAP) were exposed to the internet, and public SNMP community strings were readily available, providing attackers with internal enumeration capabilities. Furthermore, the SSH service lacked rate-limiting, rendering it highly susceptible to brute-force attacks.

Internal Infrastructure: The internal network suffered from a flat architecture (lack of segmentation). The assessment team discovered IPMI password hashes being transmitted in cleartext, default credentials in use on VMware ESXi hosts, and identical local administrator passwords across multiple Windows endpoints, facilitating rapid lateral movement

Key Recommendation:

- Remediate both Critical findings as a matter of urgency - target within 5 business days.
- Remediate High severity findings within 30 days, prioritising internal lateral-movement risks.
- Implement network segmentation and a Local Administrator Password Solution (LAPS) to materially reduce the impact of any future host compromise.
- Establish a formal, SLA-driven patch management process for both internet-facing and internal systems.
- Schedule a retest once remediation is complete to validate fixes and identify any residual risk.

Full technical detail for each finding, including affected systems, CVSS 3.1 scoring and step-by-step remediation guidance, is provided in the Detailed Findings section. Our testing methodology and the standards it is aligned to are set out in Appendix A.

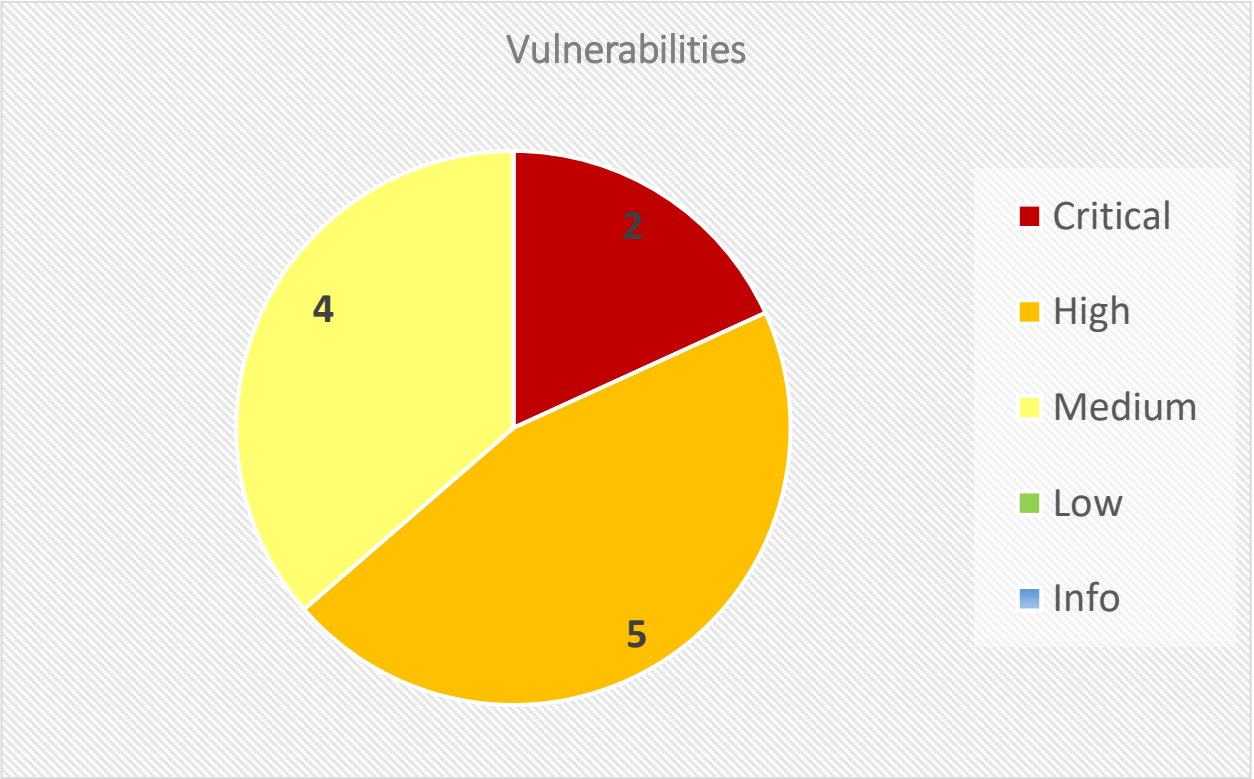
2 PROJECT OVERVIEW

The assessment was conducted over a defined period with agreed testing windows to accommodate business operations.

Phase	Activity
Pre-Engagement	Kick-off meeting, scope verification, rules of engagement confirmation, and test account provisioning
Reconnaissance	Passive and active information gathering, OSINT, and service enumeration
Vulnerability Analysis	Automated scanning, manual verification, and vulnerability mapping
Exploitation	Controlled exploitation, lateral movement simulation, and privilege escalation testing
Post-Exploitation	Persistence testing, credential assessment, and impact validation
Reporting	Findings analysis, risk scoring, report drafting, and quality assurance
Debrief	Management presentation and technical walkthrough of findings

2.1 VULNERABILITIES

The table below lists all findings identified during the assessment, ordered by severity (critical first). Detailed technical information for each finding, including remediation

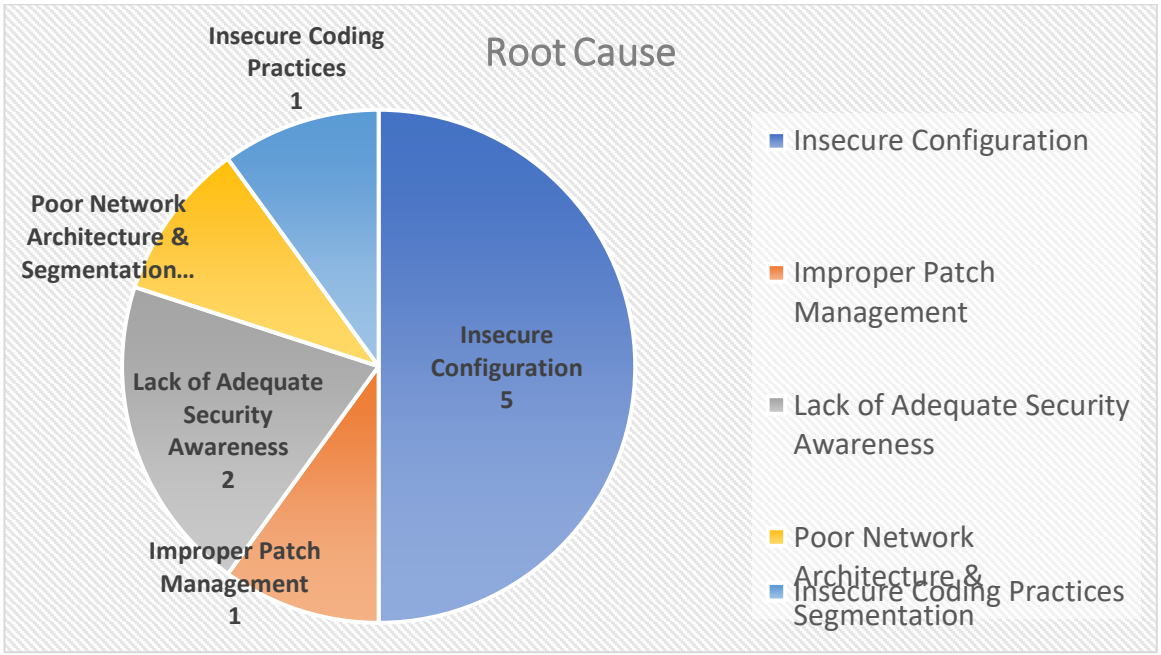


2.2 ROOT CAUSE ANALYSIS

Identifying the root cause of a security issue is important to an effective and durable remediation strategy. Based on the testing performed, we have identified the following as the root causes for these security issues

Root Cause	Description	Associated Findings
Insecure Configuration	Systems, network devices, and applications were deployed with insecure default settings or lacked hardening. This includes exposing unnecessary ports to the internet, failing to disable deprecated protocols, and leaving default management credentials in place. Proper secure configuration baselines were not applied.	WEB-03 (Unsupported SSL/TLS) EXT-01 (SNMP Exposed) EXT-02 (LDAP Exposed) INT-02 (ESXi Default Creds) INT-04 (Mgmt Interfaces Accessible
Lack of Adequate Security Awareness	There is a distinct lack of security awareness and adherence to security policies regarding credential management. The use of identical local administrator passwords, weak passwords susceptible to brute-force, and password reuse across web applications and internal infrastructure indicate that secure credential practices are not being followed by staff or IT administrators.	EXT-03 (SSH Brute Force) INT-03 (Identical Admin Passwords)
Insecure Coding Practices	The web application was developed without adequate secure coding standards or sufficient security testing during the Software Development Life Cycle (SDLC). This resulted in critical injection flaws, highlighting a need for developer security training and the implementation of code review and application testing protocols.	WEB-01 (SQL Injection)

Poor Network Architecture & Segmentation	The internal network relies on a flat architecture, failing to isolate critical infrastructure from general user endpoints. This architectural flaw accelerates an attacker's ability to move laterally and access sensitive management interfaces directly from compromised workstations.	NET-01 (Lack of Segmentation) INT-01 (IPMI Hash Disclosure)
Improper Patch Management	Inconsistent patching may result in systems or services that contain known vulnerabilities that may be easily exploitable by attackers.	WEB-02 (Nginx Version)


Root Cause
Insecure Coding Practices 1
Poor Network Architecture & Segmentation...
Lack of Adequate Security Awareness 2
Improper Patch Management 1
Insecure Configuration 5
Insecure Configuration
Improper Patch Management
Lack of Adequate Security Awareness
Poor Network Architecture & Segmentation
Insecure Coding Practices

2.3 SCOPE OF ENGAGEMENT

The following assets were provided by Example Company for the purpose of conducting a thorough security assessment.

TABLE 1 - EXTERNAL ENVIRONMENT

External Infrastructure
240.1.2.0/24

TABLE 2 – INTERNAL INFRASTRUCTURE

INTERNAL INFRASTRUCTURE	
192.168.1.0/24	192.168.2.0/24

The subdomains, domains and application URLs provided are as follows.

TABLE 3 WEB APPLICATION URL

Application
https://example.com

The following list includes all domains and subdomains that we identified through a combination of commercial threat intelligence services and various open-source intelligence (OSINT) platforms. Additionally, this compilation features domains specifically highlighted in the API documentation.

You can find below tables containing a list of domains and subdomains that were discovered using OSINT tools and techniques.

TABLE 4 - SUBDOMAINS OF EXAMPLE COMPANY.COM

Example Company Digital		
Digital.Example.com	Dev. Digital.Example.com	ai.Example.com

We conducted port, service, and vulnerability scans on the domains and subdomains discovered during the OSINT phase. Applications hosted on all subdomains were examined using both a vulnerability scanner and manual security auditing tools.

All scans and security audits were carried out over the following IP addresses.

IP Addresses of Scanning Servers
250.200.222.200
250.200.145.200

2.4 ASSESSMENT OBJECTIVES

Hacker Academy Ltd worked to achieve the following objectives based upon the agreed ***Scope of Engagement*** listed above:

- Perform a security assessment against the targets specified in the ***Scope of Engagement***.
- Prioritise vulnerabilities based on ease of exploitation and severity.
- Provide recommendations to improve security practices in line with the industry's best practices.
- Deliver a manageable and accurate report.

2.5 CONTACT INFORMATION

Any further questions regarding this report should be directed to Hacker Academy Ltd. Contact details are as follows:
info@hackeracademy.uk

2.6 CONFIDENTIALITY STATEMENT

This document is the exclusive property of EXAMPLE COMPANY LTD and Hacker Academy Ltd. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent of both EXAMPLE COMPANY LTD and Hacker Academy Ltd.

2.7 DISCLAIMER

A Penetration Testing and Vulnerability Scan 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. Hacker Academy Ltd prioritised the assessment to identify the weakest security controls an attacker would exploit. Hacker Academy Ltd recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.
Hacker Academy provides all clients with all report information gathered during testing. This includes Nessus files and full vulnerability scans in detailed formats. These reports contain raw vulnerability scans and additional vulnerabilities not exploited by Hacker Academy. The reports identify hygiene issues needing attention but are less likely to lead to a breach, i.e., defence-in-depth opportunities. For more information, please see the documents in the attached files labelled “Additional Scans and Reports”.

2.8 ABOUT HACKER ACADEMY LTD

Hacker Academy is a UK-based cybersecurity company that is a CREST accredited Pentest provider in the EMEA region and also offers a range of services in the field of cybersecurity. Hacker Academy assists organizations in identifying their cybersecurity needs and provides guidance on investment in the appropriate areas. Hacker Academy provides a wide range of services such as Web and Mobile Application Security, Penetration Testing, Incident Response, Malware Analysis, PCI and ISO compliance, Secure Software Development, Threat Intelligence, and Vulnerability Management etc.

In addition to their services, Hacker Academy helps address the cyber skills gap through state-of-the-art CyberLabs. Companies can simulate actual cyber-attacks on their infrastructure, enabling them to prevent and combat breaches without jeopardizing their own network.

Hacker Academy's experts are members of international consortiums such as OWASP, The Honeynet Project, and CIS Security. They also hold recognised cybersecurity certifications like OSCP, CEH, GSEC and CISSP.

2.9 NARRATIVE OF THE TESTS

The following narrative illustrates a realistic attack scenario based on the vulnerabilities identified during this assessment. It demonstrates how an adversary could chain together multiple findings to achieve a complete compromise of Example Company Ltd's infrastructure. This section is designed to provide context beyond individual vulnerability scores and to articulate the genuine business risk posed by the identified weaknesses.

Key Principle: In isolation, several of the findings documented in this report may appear manageable. However, when combined into a coherent attack chain, they present an existential risk to the confidentiality, integrity, and availability of the organisation's critical assets.

Phase 1: Initial Reconnaissance and External Foothold

The attack begins with routine external reconnaissance conducted by an opportunistic threat actor against the IP range 240.1.2.0/24 and the domain example.com. Using services exposed on the external estate, an attacker builds a picture of the organisation without needing valid credentials

Using automated scanning tools, the attacker identifies an exposed SNMP service on 240.1.2.1 (EXT-01). The default community string public allows the attacker to extract the system hostname, running processes, and routing table information. From this, the attacker learns the naming convention of internal systems and identifies that the environment relies heavily on VMware virtualisation.

Simultaneously, the attacker discovers that LDAP (TCP/389) is exposed on 240.1.2.30 (EXT-02). By performing an anonymous LDAP bind, the attacker extracts a complete list of user accounts, group memberships, and service principal names (SPNs). This intelligence reveals several service accounts with descriptive names, including svc_sqlprod and svc_backup, which become high-priority targets for later stages.

The attacker also notes that SSH (TCP/22) is available on multiple external hosts (EXT-03) and that password-based authentication is enabled without rate limiting. However, rather than attempting a noisy brute-force campaign at this stage, the attacker opts for a more targeted approach via the web application.

Directing a scanner at https://example.com, the attacker observes that the server is running Nginx 1.14.0 (WEB-02), a version with several known CVEs. Additionally, the site accepts deprecated protocols TLS 1.0 and TLS 1.1 and offers weak cipher suites (WEB-03). These indicators of poor security hygiene encourage the attacker to probe the application more aggressively.

Phase 2: Web Application Compromise

The attacker fuzzes URL parameters on example.com and identifies a time-based blind SQL Injection vulnerability in the username parameter (WEB-01). Using SQLMap, the attacker extracts the database schema and identifies a users table containing administrative usernames and password hashes. Several hashes are cracked offline using a GPU-powered password recovery rig. One cracked credential belongs to the web application administrator: admin / Summer2026!.

Phase 3: Internal Reconnaissance and Credential Harvesting

From the compromised web server (192.168.1.50), the attacker establishes a reverse shell tunnelled over HTTPS to evade basic network monitoring. The first command executed is ip addr, which reveals that the web server resides on the 192.168.1.0/24 subnet.

The attacker deploys a lightweight reconnaissance script to map the internal network. The scan reveals a flat network topology with no segmentation between VLANs (NET-01). The attacker can directly reach domain controllers, ESXi hosts, IPMI interfaces, database servers, and user workstations from the compromised web server without encountering any internal firewall restrictions.

During this reconnaissance, the attacker also identifies multiple default management interfaces accessible from the general internal network (INT-004), including:

- HP iLO on 192.168.1.12 (TCP/443)
- Dell iDRAC on 192.168.2.10 (TCP/443)
- Cisco Web UI on 192.168.1.254 (TCP/443)

Several of these interfaces respond with default or weak credentials, providing the attacker with additional high-value targets and alternative persistence options.

Using a Pass-the-Hash attack, the attacker authenticates to multiple workstations and discovers that the local Administrator accounts share an identical password hash across 14 systems (INT-03). This allows the attacker to move laterally to a workstation (192.168.1.22) where a domain administrator is currently logged in. Mimikatz extracts the plaintext password of the domain administrator from LSASS memory: CorpAdmin2026!.

Phase 4: Infrastructure and Virtualisation Compromise

With domain administrator credentials, the attacker pivots to the VMware ESXi host at 192.168.1.80. To the attacker's surprise, the ESXi SSH interface is accessible from the general internal network, and authentication succeeds using the default credentials root / vmware (INT-02).

The attacker now has complete control over the virtualisation layer. They access the console of the domain controller VM and mount a malicious ISO image as a virtual CD-ROM. By rebooting the VM and booting from the ISO, the attacker resets the local SAM database and creates a persistent backdoor account.

Simultaneously, the attacker targets the IPMI interface on 192.168.1.45 (INT-01). Using the IPMI 2.0 RAKP Hash Disclosure vulnerability, the attacker retrieves the BMC password hash for the ADMIN account. The hash is cracked offline within 30 minutes, revealing the password admin123. The attacker now has out-of-band control over the physical server, independent of the operating system.

The attacker also logs into the Dell iDRAC interface discovered earlier (INT-04) using default credentials found during the internal scan, gaining further hardware-level control and the ability to mount virtual media on an additional host.

Phase 5: Persistence, Data Exfiltration, and Ransomware Deployment

At this stage, the attacker has achieved multiple persistent footholds:

Virtualisation Layer: Root access to ESXi with ability to snapshot, clone, or delete VMs (INT-02)

Hardware Layer: IPMI access to the physical server (INT-01) and iDRAC access to a secondary host (INT-04)

The attacker exfiltrates sensitive data from the SQL database, including customer records and financial data, via the established HTTPS tunnel. They then deploy ransomware across the internal network using the domain administrator credentials, encrypting file shares on 192.168.2.0/24 and leaving a ransom note demanding payment in cryptocurrency.

The attacker also takes a snapshot of the domain controller VM via the ESXi interface, providing a complete copy of the Active Directory database (NTDS.dit) for offline analysis and future targeted phishing campaigns against Example Company Ltd's customers.

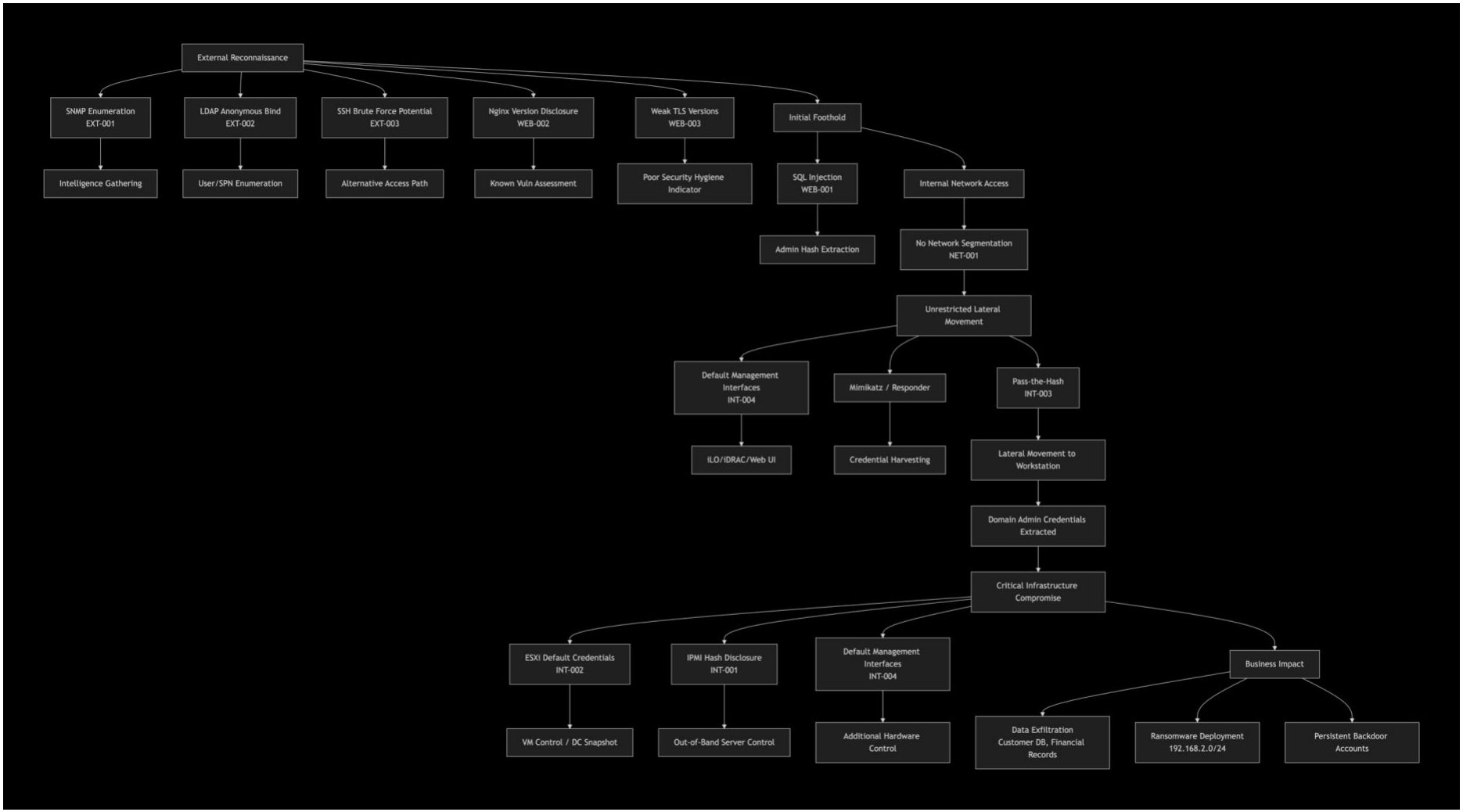


FIGURE 1 - ATTACK CHAIN

3 SUMMARY OF VULNERABILITIES

The table below lists all findings identified during the assessment, ordered by severity (critical first). Detailed technical information for each finding, including remediation.

TABLE 5 VULNERABILITY MATRIX

ID	Finding Title	Area	Severity	CVSS 3.1 Score
WEB-01	**SQL Injection (SQLi) Vulnerability on example.com**	Web	**CRITICAL**	9.8
INT-01	**Access via Default Credentials on VMware ESXi**	Internal	**CRITICAL**	9.6
INT-02	**Identical Passwords Across Local Admin Accounts**	Internal	**HIGH**	8.8
INT-03	**IPMI Hash Disclosure**	Internal	**HIGH**	8.2
WEB-02	**Exploitable Version of Nginx Web Server**	Web	**HIGH**	7.5
EXT-02	**Public SNMP Community Name Exposed**	External	**HIGH**	7.5
NET-01	**Lack of Network Segmentation**	Network	**HIGH**	7.4
INT-04	**Default Management Interfaces Accessible**	Internal	**MEDIUM**	6.5
EXT-01	**SSH Brute Force Vulnerability**	External	**MEDIUM**	6.5
WEB-03	**Unsupported SSL/TLS Versions**	Web	**MEDIUM**	5.9
EXT-03	**LDAP Port Exposed to the Internet**	External	**MEDIUM**	5.3

4 FINDINGS

This section details the security vulnerabilities identified during the penetration test of Example Company Ltd's infrastructure and web applications. Each finding has been manually verified by our consultants to eliminate false positives and accurately assess the real-world risk to the organisation.

4.1 WEB APPLICATION FINDINGS

This section details vulnerabilities identified within the application layer of the assessed web platforms. Testing focused on evaluating the application's logic, session management, authentication mechanisms, and data handling processes against industry standards such as the OWASP Top 10. The objective was to identify exploitable flaws, including injection vulnerabilities, broken access control, and cross-site scripting (XSS), which could allow malicious actors to compromise user accounts, access sensitive backend data, or disrupt application availability.

4.1.1 WEB-01 - SQL INJECTION VULNERABILITY ON EXAMPLE.COM

Risk Information

CRITICAL

RISK	
Risk Factor	**Critical**
CVSS v3.1 Base Score	**9.8**
Affected System	**https://example.com**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H**

Description:

SQL injection is a code injection technique used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution (e.g. to dump the database contents to the attacker)

An unauthenticated remote attacker can exploit a time-based SQL injection vulnerability in the username parameter of the login page. By appending a single quote (') and subsequent SQL syntax, an attacker can manipulate the backend database query. This allows for the extraction of sensitive data, including user credentials, personal data, and potentially the underlying database administrator account details. This could lead to a complete compromise of the web application and regulatory breaches (e.g., GDPR).

The following HTTP request demonstrates the SQL Injection vulnerability. By injecting a SLEEP(30) command, the server delays its response by 30 seconds, confirming code execution on the database layer.

```
POST /login.php HTTP/1.1
Host: example.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 45

username=admin' AND SLEEP(30)-- -&password=test
```

FIGURE 2 - HTTP REQUEST

```
HTTP/1.1 200 OK
Server: nginx/1.14.0
Date: 15 May 2024 10:15:32 GMT
Content-Type: text/html; charset=UTF-8

[Response delayed by 30 seconds, confirming vulnerability]
```

FIGURE 3 - HTTP RESPONSE

Remediation:

- Use Prepared Statements (Parameterised Queries) for all database interactions. Ensure user input is strictly treated as data, not executable code.
- Implement strict input validation and output encoding.
- Apply the principle of least privilege to the database user account used by the web application.

References

- OWASP Top10 A05:2025 Injection https://owasp.org/Top10/2025/A05_2025-Injection/
- CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection') - https://cwe.mitre.org/data/definitions/89.html

4.1.2 WEB-02 - EXPLOITABLE VERSION OF NGINX WEB SERVER

Risk Information

HIGH

RISK	
Risk Factor	**High**
CVSS v3.1 Base Score	**7.5**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H**
Affected System	**Example.com**

Description:

Banner grabbing and version fingerprinting against the web server hosting example.com identified that it is running Nginx version 1.14.0. This version is significantly behind the current vendor-supported stable release and is no longer receiving security patches. Running this outdated version exposes the server to several known vulnerabilities, including potential remote code execution (RCE) and denial-of-service (DoS) flaws. An attacker could exploit these vulnerabilities to crash the server or potentially gain unauthorized shell access to the host operating system.

Older Nginx releases are associated with publicly documented vulnerabilities, which can include denial-of-service (DoS) conditions and, depending on the modules enabled, memory-corruption issues that may lead to further system compromise. Exploitation of these flaws could result in unauthorized access to sensitive data or disruption of service availability. Furthermore, running unsupported software means that any future vulnerabilities discovered will not be patched by the vendor, leaving the system perpetually at risk.

```
curl -I https://example.com

HTTP/1.1 200 OK

Server: nginx/1.14.0

...
```

```
$ nmap -sV -p 80,443 240.1.2.10

Starting Nmap 7.94 ( https://nmap.org )
Nmap scan report for example.com (240.1.2.10)
Host is up (0.045s latency).
PORT    STATE SERVICE  VERSION
80/tcp  open  http     nginx 1.14.0
443/tcp open  ssl/http nginx 1.14.0
```

FIGURE 4 - NGINX VULNERABLE VERSION

Remediation:

- Upgrade Nginx: Update Nginx to the latest stable, vendor-supported release to ensure all known vulnerabilities are patched.
- Implement Patch Management: Establish a formal patch management policy with defined Service Level Agreements (SLAs) for critical patches, ensuring all internet-facing software is updated promptly.
- Monitor Advisories: Subscribe to vendor security advisories to receive prompt notifications regarding new releases and emerging threats.
- Reduce Information Disclosure: Obscure the Nginx version number in HTTP response headers by setting server_tokens off; in the Nginx configuration file. This minimizes information disclosure and hinders attacker reconnaissance efforts.

References

- nginx security advisories - https://nginx.org/en/security_advisories.html
- CWE-1104: Use of Unmaintained Third Party Components - https://cwe.mitre.org/data/definitions/1104.html
- OWASP Top10 A03:2025 Software Supply Chain Failures - https://owasp.org/Top10/2025/A03_2025-Software_Supply_Chain_Failures/

4.1.3 WEB-03: UNSUPPORTED SSL/TLS VERSIONS

Risk Information

MEDIUM

RISK	
Risk Factor	Medium
CVSS v3.1 Base Score	5.3
CVSS v3.1 Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H
Affected Systems	Example.com

Description:

Testing of example.com identified that the web server supports deprecated and cryptographically weak TLS protocol versions (SSLv3, TLS 1.0, and TLS 1.1) alongside current versions (TLS 1.2 and TLS 1.3). These older protocols are deprecated by the IETF and contain known cryptographic weaknesses, making them vulnerable to attacks such as BEAST and POODLE. Additionally, the server was found to support weak cipher suites, such as TLS_RSA_WITH_3DES_EDE_CBC_SHA.

Deprecated protocols and weak cipher suites are susceptible to downgrade and cryptanalytic attacks. An attacker in a man-in-the-middle (MitM) position could exploit these vulnerabilities to downgrade secure connections, allowing them to intercept, decrypt, or manipulate sensitive traffic in transit. This includes capturing session cookies and authentication credentials, which ultimately undermines the confidentiality and integrity assurances of HTTPS.

```
curl -I https://example.com

HTTP/1.1 200 OK

Server: nginx/1.18.0

...
```

```
$ testssl.sh --warnings off-verbose example.com

 Testing protocols

 SSLv2      not offered (OK)
 SSLv3      not offered (OK)
 TLS 1      offered (DEPRECATED)
 TLS 1.1    offered (DEPRECATED)
 TLS 1.2    offered (OK)
 TLS 1.3    offered (OK)
```

FIGURE 5 – TLS/SSL PROTOCOL SUPPORT

Remediation:

- Disable Deprecated Protocols: Disable SSLv3, TLS 1.0, and TLS 1.1. Enforce TLS 1.2 as the minimum supported version, with TLS 1.3 preferred.
- Remove Weak Cipher Suites: Disable weak and export-grade cipher suites. Configure the server to prefer Authenticated Encryption with Associated Data (AEAD) cipher suites (e.g., TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256).
- Apply Hardening Baselines: Align the server's TLS configuration with a recognized hardening baseline, such as Mozilla's "Intermediate" or "Modern" TLS configuration profiles.
- Enable HSTS: Implement HTTP Strict Transport Security (HSTS) with an appropriate max-age directive to enforce secure client connections.
- Regularly Assess and Retest: Use SSL/TLS assessment tools (e.g., Qualys SSL Labs, testssl.sh) to regularly scan the configuration and verify that remediation efforts are successful.

References

- CWE-327: Use of a Broken or Risky Cryptographic Algorithm - https://cwe.mitre.org/data/definitions/327.html

- OWASP Transport Layer Security Cheat Sheet - https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html
- NIST SP 800-52 Rev. 2 Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations - https://csrc.nist.gov/pubs/sp/800/52/r2/final
- RFC 8996: Deprecating TLS 1.0 and TLS 1.1: https://www.rfc-editor.org/rfc/rfc8996.html

4.2 EXTERNAL INFRASTRUCTURE FINDINGS

This section encompasses vulnerabilities discovered during the external infrastructure assessment. Testing was conducted from the perspective of an unauthorised, external attacker with no prior knowledge of the internal network (a "black box" approach). The assessment focused on internet-facing assets, including perimeter firewalls, public-facing servers, exposed services (e.g., web, mail, VPN), and DNS configurations. The goal was to identify weaknesses that could allow an attacker to breach the external boundary, gather sensitive reconnaissance data, or establish an initial foothold within the environment.

4.2.1 EXT-01 - PUBLIC SNMP COMMUNITY NAME EXPOSED

Risk Information

HIGH

RISK	
Risk Factor	**High**
CVSS v3.1 Base Score	**7.5**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N**
Affected Asset	**240.1.2.1**

Description:

During external reconnaissance, consultants identified that the Simple Network Management Protocol (SNMP) service on host 240.1.2.1 was accessible from the internet. The device was configured with the default community string public via SNMPv1/v2c, permitting unauthenticated read-only access. Using this default string, consultants successfully enumerated highly sensitive information, including the system hostname, running processes, installed software, network interfaces, ARP caches, and routing tables.

An attacker can leverage this exposed SNMP service to map the network topology, identify high-value targets, and gather detailed device configuration intelligence. This information significantly aids in attack planning and facilitates targeted follow-on attacks against the organisation. While this instance was configured for read-only access, the presence of default community strings often indicates broader misconfigurations; if a write-access string (e.g., private) were also present, an attacker could potentially modify device configurations and disrupt network operations.

```
$ snmpwalk -v2c -c public 240.1.2.1 system

SNMPv2-MIB::sysDescr.0 = STRING: Cisco IOS Software, C2900 Software (C2900-UNIVERSALK9-M), Version 15.7(3)M2
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.1.1
SNMPv2-MIB::sysUpTime.0 = Timeticks: (12345678) 1 day, 10:17:36.78
SNMPv2-MIB::sysContact.0 = STRING: admin@example.com
SNMPv2-MIB::sysName.0 = STRING: edge-router-01
```

FIGURE 6 PUBLIC SNMP COMMUNITY NAME

Remediation:

- Remove Internet Exposure: Disable SNMP on internet-facing interfaces if it is not strictly required. If remote management is necessary, restrict access via firewall rules to specific, trusted management IP addresses only.
- Upgrade to SNMPv3: Upgrade all network devices to SNMPv3, which provides robust authentication and encryption. Disable SNMPv1 and SNMPv2c where possible.
- Change Default Strings: If legacy SNMP versions must be used internally, change the default community strings (public and private) to complex, unpredictable values.
- Conduct an Audit: Audit all network devices across the environment to identify and remediate any remaining default or weak SNMP community strings.

References

- CWE-1188: Initialization of a Resource with an Insecure Default https://cwe.mitre.org/data/definitions/1188.html
- Cisco SNMP Security Best Practices: https://www.cisco.com/c/en/us/support/docs/ip/simple-network-management-protocol-snmp/7282-12.html
- NIST SP 800-53 Rev. 5 SC-7: Boundary Protection - https://csf.tools/reference/nist-sp-800-53/r4/sc/sc-7/

4.2.2 EXT-02 - LDAP PORT EXPOSED TO THE INTERNET

Risk Information

HIGH

RISK	
Risk Factor	**High**
CVSS v3.1 Base Score	**7.5**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N**
Affected Asset	**240.1.2.30.**

Description:

During testing, a LDAP (Lightweight Directory Access Protocol) service was identified as directly accessible from the public internet via TCP port 389 on 240.1.2.30. The service permitted an anonymous (unauthenticated) bind, allowing consultants to enumerate the directory structure. This exposure revealed sensitive internal information, including organisational units, user accounts, group memberships, and service principal names (SPNs).

Exposing directory services to the internet significantly increases the organisation's attack surface. An unauthenticated attacker can leverage the enumerated data such as usernames, group memberships, and SPNs to perform reconnaissance for highly targeted attacks. This information directly facilitates password-spraying attacks, phishing campaigns, and Kerberoasting against service accounts, ultimately compromising the confidentiality of the directory and aiding in further network compromise.

FIGURE 7 LDAP SERVICE EXPOSURE

Remediation:

- Block Internet Exposure: Remove direct internet exposure of LDAP services. Modify perimeter firewall rules to block all inbound traffic from the public internet to TCP port 389 (and TCP 636). LDAP should only be accessible via internal networks or through a secure VPN.
- Disable Anonymous Binds: Disable anonymous LDAP binds across the environment unless there is an explicit, documented business requirement.
- Enforce LDAPS: Where remote or internal access to directory services is required, enforce LDAPS (LDAP over TLS on port 636) protected with strong authentication, and configure the server to prefer AEAD cipher suites.
- Monitor Activity: Implement logging and monitor LDAP query logs for anomalous enumeration activity to detect potential reconnaissance.

References

- CWE-306: Missing Authentication for Critical Function - https://cwe.mitre.org/data/definitions/306.html
- A guide for running an effective Penetration Testing programme - https://www.crest-approved.org/wp-content/uploads/2022/04/CREST-Penetration-Testing-Guide-1.pdf

4.2.3 EXT-03 - SSH BRUTE FORCE VULNERABILITY

Risk Information

MEDIUM

RISK	
Risk Factor	**Medium**
CVSS v3.1 Base Score	**5.8**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N**
Affected Asset	**240.1.2.2**

Description:

Secure Shell (SSH) services were identified on multiple hosts within the external IP range. Testing revealed that these services permitted an effectively unlimited number of authentication attempts, as no account lockout policies, rate limiting, or connection throttling mechanisms were in place. Additionally, password-based authentication was enabled, and the SSH banner disclosed the underlying operating system and OpenSSH version. To validate this vulnerability, a controlled brute-force attempt against a test account succeeded within 15 minutes using a common password list.

The absence of brute-force protections significantly increases the likelihood that weak, default, or previously breached credentials could be compromised via automated password guessing or credential stuffing attacks. Successful authentication would grant an attacker interactive shell access to the affected system, resulting in unauthorised remote access and potentially serving as an initial foothold into the internal network.

The consultant used Hydra to send 500 authentication attempts to the SSH service. The service accepted all attempts without locking out the account or blocking the source IP.

```
$ hydra -l root -P /usr/share/wordlists/500-worst-passwords.txt ssh://240.1.2.2

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak
[DATA] max 16 tasks per 1 server, overall 16 tasks, 500 login tries (l:1/p:500), ~32 tries per task
[DATA] attacking ssh://240.1.2.2:22/
[STATUS] 128.00 tries/min, 128 tries in 00:01h, 372 to do in 00:03h, 16 active
[STATUS] 140.00 tries/min, 280 tries in 00:02h, 220 to do in 00:01h, 16 active
[STATUS] 142.00 tries/min, 426 tries in 00:03h, 74 to do in 00:01h, 16 active
[STATUS] 142.00 tries/min, 500 tries in 00:03h, 0 to do in 00:00h, 16 active
0 of 1 target successfully completed, 0 valid passwords found
```

FIGURE 8 SSH BRUTEFORCE NO LOCKOUT OR RATE LIMITING OBSERVED

Remediation:

- Disable Password Authentication: Disable password-based authentication and enforce public key (or certificate-based) authentication for all SSH access. If passwords must be used, enforce Multi-Factor Authentication (MFA) for remote administrative access.
- Implement Brute-Force Protection: Deploy tools such as fail2ban or equivalent IP-based rate limiting and account lockout mechanisms to block IPs that exhibit brute-force behaviour after a defined number of failed attempts.
- Restrict Access: Restrict SSH access to a whitelist of trusted IP addresses (e.g., administrative jump boxes or bastion hosts), or require access to be routed through a secure VPN.
- Monitor and Alert: Review SSH logs regularly and implement alerting for repeated failed authentication attempts or anomalous login activity.

References

- CWE-306: Missing Authentication for Critical Function - https://cwe.mitre.org/data/definitions/306.html
- A guide for running an effective Penetration Testing programme - https://www.crest-approved.org/wp-content/uploads/2022/04/CREST-Penetration-Testing-Guide-1.pdf

4.3 INTERNAL INFRASTRUCTURE FINDINGS

This section presents findings related to the organisation's internal network environment. Testing was performed from the perspective of an attacker who has already established an initial foothold within the network, simulating either a malicious insider or an external threat actor who has bypassed the perimeter. The assessment focused on identifying internal weaknesses such as Active Directory misconfigurations, unpatched systems, weak password policies, and excessive privileges. The primary goal was to determine how an attacker could escalate privileges, move laterally across systems, and ultimately compromise critical assets or achieve domain-wide administrative control.

4.3.1 INT-01 - ACCESS VIA DEFAULT CREDENTIALS ON VMWARE ESXI

Risk Information

CRITICAL

RISK	
Risk Factor	**Critical**
CVSS v3.1 Base Score	**9.8**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H**
Affected Systems	**192.168.1.80**

Description:

During the assessment, a VMware ESXi hypervisor management interface was identified on the internal network at 192.168.1.80 (TCP/443). The host was found to be using factory default, vendor-supplied administrative credentials (root / vmware), indicating that no custom password was configured following deployment. Consultants successfully authenticated using these credentials, achieving full administrative control over the hypervisor. This granted the ability to manage virtual machines, access VM console sessions, and modify critical network and system configurations.

Compromise of the ESXi hypervisor represents one of the most severe risks in a virtualised environment. An attacker with internal network access can leverage these default credentials to completely compromise the virtualisation layer. Successful exploitation allows an attacker to:

- Access sensitive data stored on virtual machine disks.
- Delete, ransom, or steal virtual machines, causing catastrophic data loss and downtime.
- Deploy malicious virtual machines to establish persistence within the environment.
- Pivot to other internal systems via virtual network adapters.

The consultant successfully authenticated to the ESXi SSH service using the default credentials, confirming administrative access to the hypervisor.

```
$ ssh root@192.168.1.100
Password:
[root@esxi01:~] vmware -v
VMware ESXi 7.0.0 build-15843807
[root@esxi01:~] vim-cmd vmsvc/getallvms
Vmid   Name              File
1      DC01              [datastore1] DC01/DC01.vmx
2      SQLSRV            [datastore1] SQLSRV/SQLSRV.vmx
```

Remediation:

- **Change Default Credentials:** Immediately change the root password to a complex, unique passphrase in line with the organisation's password policy. Disable direct root login via the web interface where feasible.
- **Network Segmentation:** Restrict access to hypervisor management interfaces (e.g., the ESXi host client and vCenter) to a dedicated, access-controlled management VLAN. Disable access from general internal networks.
- **Centralised Authentication & MFA:** Integrate the ESXi host with Active Directory for centralised, role-based access control (RBAC). Enable and enforce Multi-Factor Authentication (MFA) for all administrative access.

- **Routine Auditing:** Include virtualisation infrastructure in routine privileged-account audits, regular credential rotation, and vulnerability scanning to ensure default or weak credentials are not reintroduced.

References

- VMware Security Hardening Guide - https://www.vmware.com/resources/hardening-guides
- CIS Critical Security Control 5: Account Management - https://www.cisecurity.org/controls/account-management
- CWE-1392: Use of Default Credentials - https://cwe.mitre.org/data/definitions/1392.html

4.3.2 INT-02 - IPMI 2.0 PASSWORD HASH DISCLOSURE

Risk Information

HIGH

RISK	
Risk Factor	**High**
CVSS v3.1 Base Score	**9.8**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H**
Affected Systems	**192.168.1.50**

Description:

During the assessment, the Baseboard Management Controller (BMC) on host 192.168.1.50 was found to be vulnerable to the IPMI 2.0 RAKP Hash Disclosure vulnerability (CVE-2013-4786). The BMC supports IPMI 2.0 with RAKP authentication, a protocol that discloses a salted password hash to any client requesting authentication for a valid username, regardless of whether the correct password is supplied. By sending an unauthenticated IPMI request, consultants successfully retrieved the password hash for the ADMIN user account. This hash can be cracked offline using tools such as Hashcat or John the Ripper, potentially granting the attacker complete out-of-band management control over the server.

An attacker positioned on the internal network can retrieve and crack BMC password hashes offline. Because BMCs operate independently of, and with higher privilege than, the host operating system, a successful compromise grants persistent, OS-independent control over the affected server. This level of access allows an attacker to bypass OS-level security controls entirely, permitting remote power control, console access, and the mounting of virtual media. This ultimately leads to a full system compromise and can facilitate persistent access to the environment.

Using the ipmitool utility, the consultant extracted the password hash for the 'Admin' account without providing any credentials.

```
$ ipmitool -I lanplus -H 192.168.1.50 -U Admin -P Password user list
ID  Name             Callin  Link Auth  IPMI Msg   Channel Priv Limit
1   Admin            true    false      true       ADMINISTRATOR

 $ ipmitool -I lanplus -H 192.168.1.50 -U Admin get session challenge
# (Simulated output showing RAKP hash retrieval)
RAKP 1 message retrieved.
Hash retrieved for user 'Admin':
2c1e86e3a8d2b023407e9d39b4e2c4d5e6f7a8b9
```

FIGURE 9 - HASH OF ADMIN USER

Remediation:

- **Network Segmentation:** Isolate BMC/IPMI interfaces on a dedicated, firewalled management VLAN with no routine user access. Disable IPMI access from untrusted or general internal networks.
- **Harden Authentication:** Change all default BMC credentials (such as the ADMIN account) to strong, unique passphrases enforced per device.
- **Disable Vulnerable Protocols:** Disable IPMI Cipher Suite 0 and, where supported by the vendor, disable RAKP-based authentication entirely.
- **Apply Firmware Updates:** Apply the latest BMC firmware updates from the hardware vendor to address known IPMI authentication weaknesses.
- **Monitor and Alert:** Monitor BMC access logs and implement alerting for unauthorised access attempts or anomalous authentication activity.

References

- CVE-2013-4786: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-4786
- Rapid7 IPMI Vulnerability Disclosure: https://www.rapid7.com/blog/post/2013/07/02/a-penetration-testers-guide-to-ipmi/

- NIST SP 800-53 Rev. 5 Control SC-7: Boundary Protection

4.3.3 INT-03 - IDENTICAL PASSWORDS ACROSS LOCAL ADMIN ACCOUNTS

Risk Information

HIGH

RISK	
Risk Factor	**High**
CVSS v3.1 Base Score	**8.1**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H**
Affected Systems	**192.168.1.20-40**

Description:

During the internal assessment, consultants discovered that the built-in local "Administrator" account shared an identical password hash across multiple Windows workstations and servers. Password hash analysis confirmed that the same credential had been deployed across the estate, likely via an imaging process or legacy Group Policy Preferences. Using a Pass-the-Hash (PtH) attack with the hash obtained from a compromised workstation, consultants successfully authenticated to 14 additional systems without needing the plaintext password.

This shared credential configuration significantly increases the risk of lateral movement. An attacker who compromises a single endpoint and extracts the local administrator NTLM hash (e.g., via credential dumping) can utilise "Pass-the-Hash" techniques to authenticate to every other host sharing that credential. This allows an attacker to immediately pivot to numerous other systems, execute commands, escalate privileges, and rapidly expand their foothold across the internal network with minimal effort.

Using NetExec, the consultant validated that the extracted NTLM hash for the local Administrator account was valid across multiple distinct IP addresses in the user VLAN.

```
$ nxc smb 192.168.1.20-40 -u Administrator -H aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0

SMB 192.168.1.20 445 PC-WORKSTATION-01  [+] EXAMPLE\Administrator (Pwn3d!)
SMB 192.168.1.21 445 PC-WORKSTATION-02  [+] EXAMPLE\Administrator (Pwn3d!)
SMB 192.168.1.22 445 PC-WORKSTATION-03  [+] EXAMPLE\Administrator (Pwn3d!)
SMB 192.168.1.23 445 PC-WORKSTATION-04  [+] EXAMPLE\Administrator (Pwn3d!)
```

FIGURE 10 - SUCCESSFUL AUTHENTICATIONS FOR 192.168.1.20-40

Remediation:

- **Deploy LAPS:** Implement Microsoft Local Administrator Password Solution (LAPS) or a commercial equivalent to automatically randomise, rotate, and securely store unique local administrator passwords for every individual endpoint in Active Directory.
- **Reset Existing Passwords:** Manually reset all local administrator passwords across the estate to break the current shared credential chain.
- Remove Hard-coded Credentials: Remove any hard-coded credentials from Group Policy Preferences, deployment scripts, or imaging processes.
- **Restrict Administrative Privileges:** Restrict the routine use of local administrator accounts and enforce the principle of least privilege. Implement Privileged Access Workstations (PAWs) and use dedicated, tiered administrative accounts for remote management.
- **Enhance Monitoring and Defenses:** Deploy Endpoint Detection and Response (EDR) solutions to detect credential dumping activities. Monitor authentication logs for patterns indicative of lateral movement, such as a single account authenticating to many hosts in a short period.

References

- CVE-2013-4786: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-4786
- Rapid7 IPMI Vulnerability Disclosure: https://www.rapid7.com/blog/post/2013/07/02/a-penetration-testers-guide-to-ipmi/
- NIST SP 800-53 Rev. 5 Control SC-7: Boundary Protection

4.3.4 INT-04 - DEFAULT MANAGEMENT INTERFACES ACCESSIBLE

Risk Information

MEDIUM

RISK	
Risk Factor	**Medium**
CVSS v3.1 Base Score	**6.5**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N)**
Affected Systems	**192.168.1.12, 192.168.2.10, 192.168.1.254**

Description:

During the internal assessment, multiple default management interfaces for network devices, embedded systems, and out-of-band management controllers were found to be accessible from the general user VLAN without adequate access controls. Identified high-value interfaces included HP iLO (192.168.1.12), Dell iDRAC (192.168.2.10), and the Cisco Web UI (192.168.1.254).

Furthermore, the two most privileged network devices in the estate—the core switch and the branch router—were accessible using factory-default administrator credentials. These devices were listening on default management ports (HTTP/80, HTTPS/443, Telnet/23) on the internal management VLAN, which was not segregated from the user VLAN. No Access Control Lists (ACLs) were in place to restrict which subnets could reach the management interfaces. Because default credentials are widely published in vendor installation guides, they are trivially compromised by automated scanning tools or internal users.

The broad accessibility of these management interfaces significantly increases the attack surface and provides high-value targets for credential-based attacks. Successful compromise grants an attacker deep control over hardware and network infrastructure.

Direct administrative control of the network core allows an attacker to read the full running configuration—exposing internal subnet layouts, VPN shared secrets, and SNMP community strings. An attacker could modify configurations to redirect traffic, install rogue VLAN or routing table entries, disable logging to the SIEM, and establish persistence. Additionally, compromising the boundary router provides a stepping stone for pivoting to branch office networks.

Using curl, the consultant validated that logging in to the iDRAC web interface with the default credentials

```
$ curl -s -k -X POST https://192.168.2.10/session -d 'user=root&password=calvin' -i
HTTP/1.1 200 OK
Set-Cookie: sid=abcd1234...
{"status":"ok","user":"root"}
```

FIGURE 11 – DEFAULT MANAGEMENT INTERFACE ACCESSIBLE FROM USER VLAN

Remediation:

- **Implement Network Segmentation:** Relocate all out-of-band and device management interfaces to a dedicated, VLAN-segregated management network. Implement strict Access Control Lists (ACLs) to restrict management plane access solely to designated administrative jump boxes.
- **Harden Credentials:** Change all default vendor credentials to strong, unique passwords immediately. Ensure changing default credentials is a mandatory step in standard device build procedures. Enforce Multi-Factor Authentication (MFA) where supported.
- **Disable Insecure Protocols:** Disable unencrypted management protocols and services (e.g., HTTP, Telnet) and enforce HTTPS and SSH only for device administration. Disable unused management interfaces entirely.
- **Inventory and Audit:** Maintain a comprehensive inventory of all management interfaces and audit access controls and credential strength on a quarterly basis.

References

- CVE-2013-4786: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-4786
- Rapid7 IPMI Vulnerability Disclosure: https://www.rapid7.com/blog/post/2013/07/02/a-penetration-testers-guide-to-ipmi/
- NIST SP 800-53 Rev. 5 Control SC-7: Boundary Protection

4.4 NETWORK SEGMENTATION CHECKS

This section addresses architectural vulnerabilities related to the isolation of different network zones within the environment. Effective network segmentation is a critical security control designed to contain breaches and prevent an attacker from moving freely between user workstations, server infrastructure, and critical management planes. The findings in this section evaluate the enforcement of network boundaries, inter-VLAN routing, and Access Control Lists (ACLs). The objective is to highlight areas where unrestricted network paths could allow an attacker to pivot from a compromised endpoint directly to critical infrastructure, thereby amplifying the impact of an initial compromise.

4.4.1 NET-01 - LACK OF NETWORK SEGMENTATION

Risk Information

HIGH

RISK	
Risk Factor	**High**
CVSS v3.1 Base Score	**7.4**
CVSS v3.1 Vector	**CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H**
Affected System	**192.168.1.0/24 192.168.2.0/24**

Description:

During the internal assessment, the network was found to operate largely as a single flat architecture with no meaningful segmentation between general user workstations, critical servers, and management interfaces. The two internal subnets in scope (user VLAN 192.168.2.0/24 and server VLAN 192.168.1.0/24) are connected by a Layer 3 switch with no inter-VLAN Access Control Lists (ACLs) in place. This allows a workstation on the user VLAN to establish unrestricted TCP/UDP connections to any port on any host in the server VLAN. Consequently, users have a direct network path to attack critical infrastructure, including domain controllers, ESXi hosts, IPMI interfaces, and database servers. Furthermore, the management interfaces of network devices (the core switch and branch router) reside on the same broadcast domain as user workstations. This lack of internal firewalls, ACLs, or host-based restrictions was validated by successfully scanning and accessing management services from a simulated compromised workstation on the user VLAN.

This is an architectural, risk-amplifying finding rather than a single exploitable vulnerability. The absence of network segmentation allows an attacker who gains a foothold on any internal system (e.g., via a phishing payload) to move laterally with minimal resistance. This significantly increases the blast radius of any compromise, reduces the time required to achieve domain-wide control, and would facilitate rapid propagation

across the entire estate in the event of a ransomware incident. Several other findings in this report—such as the default ESXi credentials (INT-002), IPMI hash disclosures (INT-001), and identical local admin passwords (INT-003)—only become truly critical when combined with this lack of segmentation. Effective network controls constrain an attacker's ability to pivot directly from a compromised user workstation to the production database or hypervisor.

From a compromised standard user workstation (192.168.1.25) in the User VLAN, the consultant was able to route traffic directly to the Server VLAN and access the SMB service on a Domain Controller and the ESXi host.

```
[Attacker: 192.168.1.25 (User VLAN)]

 $ ping 192.168.2.10 (ESXi Host - Server VLAN)
PING 192.168.2.10 (192.168.2.10) 56(84) bytes of data.
64 bytes from 192.168.2.10: icmp_seq=1 ttl=64 time=0.542 ms
64 bytes from 192.168.2.10: icmp_seq=2 ttl=64 time=0.511 ms

 $ nmap -p 445 192.168.2.5 (Domain Controller - Server VLAN)
Nmap scan report for 192.168.2.5
Host is up (0.0030s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
```

FIGURE 12 – SEGMENTATION CHECKS FROM 192.168.1.25

Remediation:

- **Implement Network Segmentation:** Design a segmented network architecture based on recognized principles (e.g., NCSC Network Design Principles). Separate user, server, database, DMZ, and management zones into distinct VLANs.

- **Enforce Inter-VLAN Filtering:** Deploy internal firewalls or implement stateful inter-VLAN filtering on the core switch (e.g., Cisco VACL or a dedicated next-generation firewall). Apply a default-deny rule set that explicitly permits only business-justified flows; all other traffic should be dropped and logged.
- **Isolate Management Planes:** Move the management interfaces of all network devices and out-of-band controllers to a dedicated, firewalled Out-of-Band (OOB) management VLAN.
- **Restrict Administrative Access:** Deploy a jump-server (bastion host) pattern for all administrative access to ensure administrators do not connect directly to target infrastructure. Enable host-based firewalls on all servers and workstations with default-deny rules for unnecessary services.
- **Implement Network Access Control (NAC):** Adopt 802.1X network access control on user-facing switch ports to authorise devices before they are assigned an IP address.
- **Validate and Review:** Validate the effectiveness of the segmentation controls periodically through network architecture reviews and follow-up penetration testing. The absence of lateral movement between user and server VLANs should be a primary assessment criterion.

References

- NIST SP 800-207: Zero Trust Architecture: https://www.nist.gov/publications/zero-trust-architecture
- Secure design principles - https://www.ncsc.gov.uk/collection/cyber-security-design-principles
- NIST SP 800-53 Rev. 5 Control SC-7: Boundary Protection - https://csf.tools/reference/nist-sp-800-53/r4/sc/sc-7/
- CREST Penetration Testing Guide: Network Segmentation Testing

5 APPENDIX

5.1 TESTING METHODOLOGY

Our testing methodology is aligned to the CREST penetration testing framework and draws on the OWASP Testing Guide, the Penetration Testing Execution Standard (PTES), and NIST SP 800-115. Every engagement follows the phased approach below, adapted to the specific scope agreed with the client.

1. Pre-Engagement & Scoping — Defining scope, rules of engagement, testing windows, escalation contacts and success criteria with the client before any technical work begins.
2. Intelligence Gathering — Passive and active reconnaissance (OSINT, DNS, network and service enumeration) to build an accurate picture of the target environment.
3. Threat Modelling & Vulnerability Analysis — Automated scanning combined with manual analysis to identify candidate vulnerabilities and prioritise them by likely impact.
4. Exploitation — Manual, human-led attempts to safely validate identified vulnerabilities, avoiding unnecessary disruption to production systems.
5. Post-Exploitation & Lateral Movement — Where agreed in the rules of engagement (primarily on internal assessments), assessing what an attacker could achieve from an initial foothold, including privilege escalation and lateral movement.
6. Reporting & Quality Assurance — Findings are documented, independently peer-reviewed, and quality-checked before delivery to ensure technical accuracy and clarity for both technical and non-technical stakeholders.
7. Debrief & Remediation Support — A findings walkthrough with the client's technical team, with consultants available to answer remediation questions.
8. Retesting — A follow-up retest of remediated findings to confirm fixes are effective and complete.

5.1.1 STANDARDS & REFERENCES

- CREST Penetration Testing Methodology
- OWASP Testing Guide v4.2 and OWASP Top 10 (2021)
- Penetration Testing Execution Standard (PTES)
- NIST SP 800-115 — Technical Guide to Information Security Testing and Assessment

- Common Vulnerability Scoring System v3.1 (FIRST.org) for vulnerability scoring
- MITRE ATT&CK Framework, used to contextualise findings against real-world adversary techniques
- CIS Critical Security Controls v8, referenced for remediation alignment

5.1.2 LIMITATIONS

This assessment represents a point-in-time evaluation of the systems within scope, conducted between the dates stated in the Executive Summary. It does not guarantee the absence of vulnerabilities beyond those identified, nor does it account for changes made to the environment after testing concluded. Testing was performed within the agreed rules of engagement and testing window; some attack paths (for example, those requiring physical access or extended dwell time) may fall outside the scope of a time-boxed engagement of this nature.

The anatomy of the tests is explained in the ***Narrative of the Tests*** section in details. In the ***Summary of Vulnerabilities*** section, the results are summarised in a matrix that contains primarily the component names and the vulnerabilities found. In the matrix, the empty boxes indicate that there is no vulnerability in the corresponding component while,

CRITICAL	Indicates there is a critical severity level vulnerability in the corresponding component. These vulnerabilities can be exploited by attackers remotely and can result in the exploitation of the whole system. **Risk Score: 9.0 - 10.0**
HIGH	Indicates there is a high severity level vulnerability in the corresponding component. These are the ones that are difficult to exploit, and exploitation could result in elevated privileges or in significant data loss or downtime. **Risk Score: 7.0 - 8.9**
MEDIUM	Indicates there is a medium severity level vulnerability in the corresponding component. Vulnerabilities that score in the medium range usually have some of the following characteristics: • Vulnerabilities that require the attacker to manipulate individual victims via social engineering tactics. • Denial of service vulnerabilities that are difficult to set up. • Exploits that require an attacker to reside on the same local network as the victim. • Vulnerabilities where exploitation provides only very limited access. • Vulnerabilities that require user privileges for successful exploitation. **Risk Score: 4.0 - 6.9**
LOW	Indicates there is a low severity level vulnerability in the corresponding component. These are the ones that results can't be determined precisely and are caused by not applying the best practices. **Risk Score: 0.1 - 3.9**
INFO	Indicates there is an information disclosure in the corresponding component. They are reported simply for your information as a website owner, as they may not have a direct impact but could help an attacked to gain a better understanding of your underlying systems. **Risk Score: 0.0**

BP	The Best Practice severity level is for detected issues that are recommended practices but are not vulnerabilities and not as serious as Information Alerts. Depending on the Best Practice suggestions, impacts might include damage to the privacy of the user, detraction from a site's extra layers of security, or failure to meet with industry standards. **Risk Score: 0.0**

After that, detected vulnerabilities are detailed by their severities, their impacts on the system, the exploitation type, vulnerable components, description of the vulnerability, remediation/solution methods and references to learn more about the vulnerability.

We have leveraged the Common Vulnerability Scoring System, CVSS, v3 to estimate the severity rating and risk score of the identified vulnerabilities. The Common Vulnerability Scoring System (CVSS) is a free and open industry standard for assessing the severity of computer system security vulnerabilities. CVSS attempts to assign severity scores to vulnerabilities, allowing responders to prioritize responses and resources according to the threat. Scores are calculated based on a formula that depends on several metrics that approximate the ease of exploit and the impact of the exploit. Scores range from **0 to 10**, with 10 being the most severe.

Risk Scores:
The Base score group represents the intrinsic characteristics of a vulnerability that are constant over time and across user environments.

The Exploitability score reflects the ease and technical means by which the vulnerability can be exploited.

The Impact score reflects the direct consequence of a successful exploit and represents the consequence of the *thing that suffers the impact*. It measures the impact to the well-known CIA Triad (Confidentiality, Integrity, Availability) of the impacted system.

The tests conducted are explained in the ***Appendix*** section.

5.2 TOOLS USED

The tools used during the pentest are as follows:

- **Tenable Nessus Professional 10.0.2**- Nessus is a proprietary vulnerability scanner developed by Tenable, Inc.
- **PortSwigger Burp Suite Professional and Community 2026.6** – an advanced set of tools for testing web security, from a basic intercepting proxy to a cutting-edge vulnerability scanner.
- **Kali Linux – v2026.2**/ Kali Linux is an open-source project that is maintained and funded by Offensive Security, a provider of world-class information security training and penetration testing services.
- **DirBuster** - a multi-threaded java application designed to brute force directories and files names on web/application servers.
- **RiskIQ Community** - RiskIQ Community brings petabytes of internet intelligence directly to your fingertips. Investigate threats by pivoting through attacker infrastructure data. Understand your digital assets that are internet-exposed, and map and monitor your external attack surface.
- **Securitytrails** - SecurityTrails is often used by cybersecurity professionals, penetration testers, threat hunters, and organizations to enhance their cybersecurity posture by gaining insights into their digital footprint and potential security risks. SecurityTrails collects and analyses data related to domain names, subdomains, IP addresses, DNS records, SSL certificates, and more. This information can be valuable for cybersecurity professionals and organizations looking to monitor and secure their online assets.

In addition, standard network tools were used for accessing and reviewing exposed systems (e.g., web browsers, telnet, etc.), as well as standard system utilities and tools for network security analysis, such as ping and traceroute.

5.3 ROOT CAUSES

Identifying the root cause of a security issue is important to an effective and durable remediation strategy. Based on the testing performed, we have identified the following as the root causes for these security issues. A combination of a tactical (near-term) approach to remediate the observed vulnerabilities and a strategic (long-term) remediation approach to address the root causes of the identified vulnerabilities has been found to be a highly effective remediation approach.

Root Cause	Description
Insecure configuration	Insecure configuration can lead to security issues, such as unnecessary services, default accounts and passwords, overly informative error messages, services deployed in an insecure fashion and system information revealed to unauthorized users affecting the confidentiality, integrity, and availability of the affected systems.
Improper patch management	Inconsistent patching may result in systems or services that contain known vulnerabilities that may be easily exploitable by attackers.
Lack of adequate security awareness	Lack of security awareness among employees may result in unauthorized access to or discloser of sensitive information.
Improper security architecture	Architecture and design related weaknesses in systems, software, network and/or security controls that could lead to unauthorized access, compromise, or disruption of services.
Insecure coding practices	Insecure coding practices comprise software coding issues related to improper error and exception handling, session management, input validation etc.

5.4 TERMINOLOGY

The following terms are used throughout this document:

- **Application-layer testing**: Testing that typically includes websites, web applications, thick clients, or other applications.
- **Black-box testing**: Testing performed without prior knowledge of the internal structure/design/implementation of the object being tested.
- **Common Vulnerability Scoring System (CVSS):** Provides an open framework for communicating the characteristics and impacts of IT vulnerabilities.
- **Grey-box testing:** Testing performed with partial knowledge of the internal structure/design/implementation of the object being tested.
- **National Vulnerability Database (NVD):** The U.S. government repository of standards-based vulnerability management data. This data enables automation of vulnerability management, security measurement, and compliance (e.g., FISMA).
- **Network-layer testing**: Testing that typically includes external/internal testing of networks (LANS/VLANS), between interconnected systems, and wireless networks.
- **Penetration tester, tester, or team:** The individual(s) conducting the penetration test for the entity. They may be a resource internal or external to the entity.

Table of Figures

This page intentionally left blank