| **Artemis Gas, Inc.**  **Penetration Test Report** |
| --- |
| Matthew Vance  03/27/2025 |

# **Table of Contents**

[**Table of Contents**](#_99znba1g3h11) **1**

[**Scope of Work**](#_k83c7zxsdyjd) **3**

[**Project Objectives**](#_xhl894s6z496) **4**

[**Assumptions**](#_qmmb0teukwyk) **5**

[**Timeline**](#_j5ribc83h8fe) **7**

[**Summary of Findings**](#_8mt3xl6ih4o4) **8**

[Phase 1: Reconnaissance](#_z74r0b47omrg) 8

[Phase 2: Target Identification and Scanning](#_df6kcpmsj6sp) 11

[Phase 3: Vulnerability Assessment](#_dv98hxitadb1) 17

[Phase 4 & 5: Threat Assessment/Reporting and Recommendations](#_bkz45xpfj952) 22

[**References**](#_pffv7a9w7kgi) **31**

# **Scope of Work**

Artemis Gas, Inc. has engaged with our cybersecurity consulting firm to perform an external penetration test of its network infrastructure. This assessment is designed to identify potential security vulnerabilities, evaluate their associated risks, and provide actionable recommendations for remediation. This test follows a structured approach that simulates real-world attack scenarios to measure exposure to external threats. The scope of the penetration test includes reconnaissance, target identification, vulnerability assessment, and a detailed threat analysis. Given Artemis’ global presence and reliance on pipeline monitoring, cloud-based applications, and industrial control systems, evaluating and understanding the external attack surface is crucial. Final deliverables will include a Detailed Technical Report for the IT staff and an Executive Summary for senior management.

# **Project Objectives**

The primary objectives of this penetration test are as follows:

* **Assess Security Posture:** Identify any and all publicly accessible assets, exposed services, and potential attack vectors. Given Artemis’ extensive industrial gas pipeline network and monitoring systems, ensuring secure remote access to critical infrastructure will be essential.
* **Identify and Evaluate Vulnerabilities:** Simulate real-world attack methodologies in order to discover exploitable weaknesses in cloud services, externally available databases, and web applications. Identifying any security gaps can help prevent unauthorized access to trade secrets, customer data, and critical operational technologies.
* **Assess Business Risks:** Evaluate and determine the potential impact of security threats on global operations, including supply chain disruptions, regulatory non-compliance, and reputational damage. This report will offer insight to help prioritize security investments to mitigate risks effectively.
* **Provide Actionable Recommendations:** Develop an enhanced security roadmap that is tailored to Artemis’ unique operational requirements. Recommendations will primarily focus on securing cloud environments, tightening access controls, patching known vulnerabilities, and improving monitoring capabilities.
* **Ensure Compliance and Best Practices:** Align security recommendations with industry standards such as NIST, ISO 27001 and CIS benchmarks. Compliance with regulatory frameworks such as GDPR and U.S. critical infrastructure protection laws will also be assessed.

# **Assumptions**

For this penetration test, the following assumptions have been made:

* **Black Box Testing:** This penetration test will be conducted without prior knowledge of internal configurations beyond information that is publicly available. This approach simulates how an external attacker would attempt to breach Artemis’ network without insider knowledge.
* **Testing is Limited to External Systems:** The scope only includes external-facing systems and services including cloud environments, web applications, and public network infrastructure. Internal networks, employee workstations, and physically secured assets will be excluded.
* **Controlled Testing Environment:** Testing that could cause service disruptions will be noted, and require prior approval of the Artemis’ IT team. Additionally, testing will be performed during agreed-upon maintenance windows in order to minimize any operational impact.
* **No Social Engineering Attacks:** This penetration test will not include phishing, pretexting, or any other social engineering techniques that target Artemis’ employees. The focus will remain purely on technical vulnerabilities pertaining to the external perimeter.
* **Availability of Necessary Permissions (Access Authorization):** Artemis’ IT team will provide written consent for all scanning and exploitation activities to ensure compliance with legal and ethical guidelines. If there are any unexpected findings that require deeper testing they will be reviewed with Artemis before proceeding.
* **Data Sensitivity:** Sensitive data discovered during this penetration test such as credentials or proprietary information, will be handled securely and reported exclusively to designated personnel. No sensitive data will be stored beyond the reporting phase.
* **Cooperation:** It is assumed that the Artemis IT and security teams will actively collaborate throughout the testing process, providing necessary resources and ensuring active communication to avoid misunderstandings and security conflicts.
* **Regulatory Compliance:** Although this penetration test is not a direct compliance audit, findings related to data exposure, access control weaknesses, or cloud misconfigurations may indicate non-compliance risks with regulations such as GDPR, HIPAA, or industry-specific cybersecurity mandates.
* **Mitigation Strategy Support:** This penetration test will provide detailed remediation and recommendations, however implementing security fixes and patches will be the responsibility of Artemis’ internal IT and cybersecurity teams.
* **Risk Acceptance:** Identified vulnerabilities that Artemis team chooses not to remediate will be documented, and the associated risks will be acknowledged by the company as an informed decision.
* **Ethical Conduct:** All activities related to this penetration test will be conducted in accordance with ethical hacking principles and industry best practices. Unauthorized access will not be attempted beyond the agreed-upon scope, and all findings will be reported transparently to Artemis Gas, Inc.

# **Timeline**

This penetration test will follow a structured timeline to ensure a comprehensive evaluation of Artemis Gas, Inc.’s security posture. The estimated timeline is as follows:

| Phase | Description | Duration |
| --- | --- | --- |
| 1. Reconnaissance | Gathering intelligence on external presence including cloud based assets and publicly available services. Identify subdomains, exposed databases, and potential entry points. | 2 Days |
| 2. Target Identification and Scanning | Identify live hosts, open ports, and running services using industry-standard tools. The focus during Phase 2 will be on cloud environments and public-facing network infrastructure. | 3 Days |
| 3. Vulnerability Assessment | Conducting in-depth scans to detect known vulnerabilities, misconfigurations, and outdated software versions. Findings will be prioritized based on exploitability and potential impact on business operations. | 3 Days |
| 4. Threat Assessment | Analyze identified vulnerabilities and determine potential attack scenarios. Assess the likelihood of exploitation as well as the consequences of successful attacks on data and operational systems. | 4 Days |
| 5. Reporting and Recommendations | Delivery of a comprehensive penetration test report, including a detailed technical analysis for the IT/security teams and an executive summary for senior management. Recommendations will include an outline of mitigation strategies for each identified risk. | 3 Days |

The total estimated duration for this penetration test is 15 business days. This timeline ensures a swift and thorough assessment, and minimizes operation disruptions. Additionally, this 5 phase approach will allow for the Artemis security team to address high priority vulnerabilities as they are discovered, reducing risk exposure during the testing process.

# **Summary of Findings**

## Phase 1: Reconnaissance

The reconnaissance phase involved gathering any publicly available information on Artemis Gas, Inc. to assess its external security posture. Listed are the various open-source intelligence (OSINT) tools and resources that were used to identify network assets, employee details, potential attack vectors, and any other publicly exposed information.

**Tools and Findings**

**1. WHOIS Lookup:** Conducted a domain lookup on primary web assets to identify registered domain details, subdomains, and associated infrastructure. Identified information about hosting providers, DNS records, and domain expiration dates, which could be used for further enumeration.

**2. Search Engines (Google Dorking):** Used advanced search operators to uncover any related publicly available documents, PDFs, and login portals. Identified misconfigured directories and exposed internal documents, some of which contained metadata with employee email addresses.

**3. LinkedIn Employee Enumeration:** Analyzed LinkedIn profiles to extract employee names, job titles, and technology stacks based on their listed skills. Found profiles of IT administrators and security engineers, which may provide insight into Artemis' security architecture and potential internal weaknesses.

**4. Email Harvesting Tools (Hunter.io, EmailExtractor):** Discovered corporate email patterns (e.g., firstname.lastname@artemisgas.com), allowing for enumeration of employee email addresses. Found publicly exposed emails on job postings and technical support forums, further increasing the risk of phishing or credential stuffing attacks.

**5. DNS Interrogation (MXToolbox, dig, nslookup):** Queried DNS records to identify Artemis' mail servers, web servers, and security policies (SPF, DKIM, DMARC settings). Found subdomains that might host internal applications or development environments, which could be entry points for attackers.

**6. Job Listings & Career Pages:** Reviewed Artemis’ job postings to identify technologies in use (e.g., AWS, Fortinet firewalls, SAP ERP). Found postings mentioning cloud migration efforts, indicating potential security gaps in hybrid cloud environments.

**7. Social Media Analysis (Twitter, Facebook, Employee Posts):** Discovered public discussions by employees referencing internal tools and services. Found corporate event photos revealing physical security controls (badges, office layouts).

**8. Tech Forums & Online Discussions (Reddit, Stack Overflow, Spiceworks):** Identified technical discussions by Artemis IT staff, where they requested help troubleshooting specific technologies (e.g., firewall rules, AWS configurations). Potential leakage of internal security practices and misconfigurations in public posts.

**9. GitHub & Bitbucket Repositories:** Discovered code repositories associated with Artemis and its employees. Found exposed API keys, hardcoded credentials, and configuration files, which could be used for privilege escalation or service exploitation.

**10. Security Databases (ExploitDB, CVE Mitre, Shodan):** Cross-referenced Artemis’ technology stack with known vulnerabilities in security databases. Found publicly exposed servers with outdated Apache and Microsoft Exchange versions, both of which have known critical vulnerabilities.

**11. Public Records & Government Databases:** Examined SEC filings, patent submissions, and industry regulations related to Artemis. Found patent filings for proprietary gas pipeline technologies, which could be a target for corporate espionage.

**12. Pastebin & Data Leak Sites:** Checked for previously leaked data related to Artemis on Pastebin, dark web forums, and public breach databases. Identified credentials from third-party breaches, which could be tested in credential-stuffing attacks.

**13. Shodan (Internet-Connected Device Search):** Scanned Artemis’ public IP ranges and found exposed industrial control systems (ICS) that may be accessible from the internet. Detected open RDP ports and unsecured web services, increasing the risk of unauthorized access.

**14. OSINT Tools (Maltego, Recon-ng):** Mapped organizational relationships, email domains, and potential business partners. Found third-party vendors and contractors that could be exploited in a supply chain attack.

**15. SSL/TLS Certificate Analysis (crt.sh, Censys.io):** Analyzed SSL certificates to identify Artemis’ subdomains and internal services. Found evidence of expired or weak encryption ciphers, which could be exploited for man-in-the-middle (MITM) attacks.

**Key Takeaways from Phase 1**

* Multiple publicly exposed services including cloud-based applications, legacy Cisco network devices, and industrial control systems (ICS).
* Email and job descriptions being readily available online increases the risk of phishing and social engineering attacks.
* Unpatched software (e.g., Apache, Microsoft Exchange) has known vulnerabilities that could be exploited by attackers.
* Source code and credential can be found in public repositories and pose a serious risk if exploited.
* Misconfigured cloud storage with exposed RDP ports could serve as initial access points for cyberattacks.

## Phase 2: Target Identification and Scanning

The goal of Phase 2 is to conduct active scans to identify live hosts, open services, operating systems, and any potential vulnerabilities within Artemis Gas, Inc.’s externally facing infrastructure. This phase includes the use of multiple scanning tools to map Artemis’ attack surface, and pinpoint security weaknesses.

**Tools and Findings**

**1. Nmap - Network Mapping & Service Enumeration**

#### **Purpose:** Nmap was used to scan Artemis' public IP ranges, identifying live hosts, open ports, and running services. It also provided OS fingerprinting and banner grabbing, allowing us to determine software versions that could be vulnerable to exploits.

#### **Commands Used & Findings:**

* nmap -sn <Artemis\_IP\_Range> → Identified active hosts within Artemis’ external network.
* nmap -sV -p- <Target\_IP> → Found open ports on multiple web servers, including HTTPS (443), SSH (22), RDP (3389), and SMTP (25).
* nmap -O <Target\_IP> → OS fingerprinting revealed several outdated Linux and Windows servers, some running legacy Cisco network devices.

##### **Key Takeaways:**

* Artemis' external servers expose multiple high-risk services, including SSH, RDP, and database ports, which are commonly targeted by attackers.
* Some servers run outdated operating systems, increasing the risk of known exploits.

### **2. Metasploit - Advanced Service Enumeration**

#### **Purpose:** Metasploit was used for deeper enumeration of network services, specifically targeting SMB shares, databases, and authentication mechanisms.

#### **Commands Used & Findings:**

* use auxiliary/scanner/smb/smb\_enumshares; set RHOSTS <Target\_IP\_Range>; run → Enumerated publicly accessible SMB shares containing internal documentation.
* use auxiliary/scanner/postgres/postgres\_login; set RHOSTS <Target\_IP>; run → Found PostgreSQL database login interface exposed to the internet, which increases SQL injection risks.

##### **Key Takeaways:**

* Artemis has publicly accessible file shares, which could leak confidential documents to attackers.
* Databases with open authentication mechanisms were found, which could be exploited for credential-based attacks.

### **3. Masscan - High-Speed Port Scanning**

#### **Purpose:** Masscan was used to quickly scan Artemis’ entire public IP range to detect open ports across multiple hosts.

#### **Commands Used & Findings:**

* masscan <Artemis\_IP\_Range> -p1-65535 --rate=100000 → Detected open high-risk ports, including exposed Telnet (23), RDP (3389), and outdated FTP services (21).

##### **Key Takeaways:**

* Exposed Telnet services indicate the use of insecure protocols, which should be disabled.
* RDP access from public networks increases the risk of brute-force attacks or exploitation via known vulnerabilities (e.g., BlueKeep - CVE-2019-0708).

### **4. Nessus - Vulnerability Scanning**

#### **Purpose:** Nessus was used to identify known vulnerabilities, misconfigurations, and outdated software versions across Artemis’ external-facing infrastructure.

#### **Commands Used & Findings:**

* nessuscli scan run --target <Artemis\_IP\_Range> → Found multiple critical vulnerabilities, including:
  + CVE-2021-26855 (ProxyLogon) → Affects Microsoft Exchange servers, allowing remote attackers to execute arbitrary code.
  + CVE-2019-0211 → Apache Web Server privilege escalation vulnerability detected on several hosts.
  + Weak TLS configurations detected on Artemis’ HTTPS servers, increasing the risk of man-in-the-middle (MITM) attacks.

##### **Key Takeaways:**

* Artemis’ email infrastructure is vulnerable to remote code execution due to unpatched Exchange servers.
* Apache servers running outdated versions pose a privilege escalation risk.
* Weak encryption on SSL/TLS configurations makes Artemis susceptible to data interception attacks.

### **5. Wireshark - Network Traffic Analysis**

#### **Purpose:** Wireshark was used to passively capture network traffic, analyze communication patterns, and detect potential security misconfigurations.

#### **Findings:**

* Captured authentication attempts using outdated encryption, indicating potential credential exposure.
* Observed unencrypted traffic between Artemis’ web applications and cloud services, increasing the risk of session hijacking.

##### **Key Takeaways:**

* Insecure communication protocols should be replaced with stronger encryption (e.g., forcing HTTPS, enabling SSH key authentication).
* Credentials transmitted in plaintext should be encrypted or replaced with token-based authentication.

**Challenges and Potential Drawbacks**

* **Security System Detection:** Artemis’ firewalls, IDS/IPS, and WAFs detected some scanning attempts, which may limit the ability to enumerate services completely.
* **False Positives in Nessus Scans:** Some findings required manual validation, as automated scanners may misidentify vulnerabilities.
* **Masscan Limitations:** While effective for quick port discovery, it lacks service versioning, requiring follow-up with Nmap or Nessus.
* **Wireshark Packet Capture Challenges:** To effectively capture traffic, network interface cards (NICs) must be placed in promiscuous mode, which may not always be feasible in real-world environments.

**Key Takeaways from Phase 2**

* **Multiple high-risk services are exposed:** Public RDP, Telnet, and FTP access make Artemis vulnerable to brute-force attacks and exploits. Open database ports (PostgreSQL, MySQL) could be targeted for SQL injection or credential theft.
* **Outdated and vulnerable software detected:** Microsoft Exchange Server is unpatched, leaving it open to remote code execution attacks. Apache Web Servers contain known privilege escalation flaws, increasing the risk of full system compromise.
* **Encryption weaknesses in network traffic:** Observed unencrypted credentials and data exchanges, increasing the risk of MITM and session hijacking attacks.
* **Artemis’ intrusion detection systems (IDS) flagged scanning attempts:** While firewalls and monitoring tools detected scanning activity, some weaknesses remained exploitable.

## Phase 3: Vulnerability Assessment

In this phase, a vulnerability assessment was conducted to identify security weaknesses in the Artemis Gas, Inc. infrastructure, including network devices, servers, web applications, and cloud environments. The focus was on detecting any misconfigurations, outdated software, and exploitable vulnerabilities that could be leveraged by attackers.

**Tools and Findings**

### **1. OpenVAS – Network Vulnerability Scanning**

#### **Purpose:** OpenVAS was used to scan Artemis' external network infrastructure for security gaps, outdated firmware, and misconfigurations. Since Artemis relies on a hybrid cloud and on-premise infrastructure, OpenVAS was crucial for assessing firewalls, routers, and SD-WAN devices for security weaknesses.

#### **Findings:**

* Identified unpatched vulnerabilities in legacy Cisco and Fortinet devices still in use.
* Detected outdated firewall firmware, increasing the risk of misconfiguration exploits.
* Found exposed network services running unnecessary ports, which could allow unauthorized access.

##### **Pros and Cons:**

* Generates detailed vulnerability reports with severity ratings and remediation steps.
* Supports scanning legacy network devices, which Artemis still partially relies on.
* False positives may require manual verification.
* Vulnerability definitions must be updated manually, potentially delaying detection of new threats.

### **2. Burp Suite – Web Application Security Testing**

#### **Purpose:** Burp Suite was used to assess Artemis’ web applications, including SAP, PARS, and APOLLO, which store critical business and intellectual property data. This tool helped identify vulnerabilities such as SQL injection, cross-site scripting (XSS), and authentication flaws.

#### **Findings:**

* Detected potential SQL injection risks in APOLLO’s authentication portal, which could allow attackers to access confidential trade secrets.
* Found cross-site scripting (XSS) vulnerabilities in SAP, potentially enabling attackers to steal session tokens.
* Identified misconfigured authentication policies in Artemis’ internal web applications.

##### **Pros and Cons:**

* Comprehensive scanning capabilities, detecting both manual and automated security flaws.
* Ability to intercept and modify HTTP requests to simulate authentication bypass attacks.
* Advanced scanning configurations can be complex, requiring expertise.
* The free version has limited scanning capabilities compared to the professional edition.

### **3. Tenable Nessus – Server and Compliance Scanning**

#### **Purpose:** Nessus was used to perform security assessments on Artemis' cloud-based (AWS) and on-premise servers, particularly those running Linux, Oracle 12c, and Microsoft Exchange. Since Artemis is transitioning to SD-WAN, Nessus was also used to scan Fortinet and Palo Alto devices for misconfigurations.

#### **Findings:**

* Identified CVE-2021-26855 (ProxyLogon) vulnerability in Microsoft Exchange servers, posing a risk of remote code execution.
* Found weak SSH configurations on Linux servers, increasing the risk of brute-force attacks.
* Detected unpatched Oracle 12c databases, which could be exploited for SQL injection and privilege escalation.

##### **Pros and Cons:**

* Comprehensive vulnerability database, ensuring Artemis receives up-to-date security insights.
* Supports compliance assessments for PCI DSS and HIPAA, which are critical for industrial operations.
* Requires a paid license for advanced features, making extended scanning costly.
* Can be resource-intensive on large networks like Artemis’.

### **4. Wapiti – Lightweight Web Application Scanner**

#### **Purpose:** Wapiti was used to scan Artemis' external-facing web applications for vulnerabilities. Since some business units operate unauthorized or improperly secured web services, Wapiti helped detect security gaps in these platforms.

#### **Findings:**

* Found exposed API endpoints, increasing the risk of data leakage or unauthorized access.
* Detected SQL injection vulnerabilities on an unsecured internal portal used by employees.
* Identified lack of authentication requirements on a cloud-based service.

##### **Pros and Cons:**

* Fast and efficient, ideal for quick scans of multiple web services.
* Supports authentication scanning, allowing testing of restricted portals.
* Command-line only, requiring familiarity with terminal-based execution.
* No active exploitation capabilities, only reports vulnerabilities.

### **5. w3af – Advanced Web Application Assessment**

#### **Purpose:** w3af was used to analyze Artemis’ critical web applications, including APOLLO, which stores proprietary trade secrets. This tool provided both vulnerability scanning and exploitation capabilities, allowing a more thorough web security assessment.

#### **Findings:**

* Identified weak authentication mechanisms in APOLLO, which could be brute-forced or bypassed.
* Found potential directory traversal vulnerabilities, allowing unauthorized access to sensitive files.
* Detected session management weaknesses, increasing the risk of session hijacking attacks.

##### **Pros and Cons:**

* Powerful and flexible, suitable for in-depth web application security testing.
* Can simulate attacks against weak authentication systems in Artemis' infrastructure.
* Resource-heavy, which slows down assessments on large applications.
* Complex setup, requiring additional time to configure and fine-tune scans.

## Phase 4 & 5: Threat Assessment/Reporting and Recommendations

The goal of this summary is to assess the potential threats and risks associated with vulnerabilities that are likely to be found in the infrastructure of ARTEMIS GAS, INC. Analyzing these threats allows us to determine the impact on Artemis’ operations and propose remediation measures to mitigate said risks effectively.

**1. Unpatched RDP Exposed to the Internet**

**Description:** Remote Desktop Protocol (RDP) services without the proper security controls are exposed to the internet, making them susceptible to brute-force attacks and known exploits such as CVE-2019-0708 (BlueKeep). This vulnerability affects Windows Server 2012, 2016, 2019, and Windows 10/11 systems.

**Detection Tools:** Vulnerability scanners such as Nessus and OpenVAS make it possible to detect open RDP ports and identify outdated or unpatched versions that may pose a security risk.

**Risks:** Exploitation may allow for unauthorized remote access to the Artemis internal network, potentially leading to data exfiltration, lateral movement, or ransomware deployment. Brute-force techniques may be used by attackers to guess weak credentials or leverage existing RDP vulnerabilities to execute malicious code on the system. While IDS/IPS may help monitor and block suspicious login attempts, attackers may evade detection by using slow brute-force attacks with tools like crowbar, or by leveraging stolen credentials from previous breaches. If login credentials are obtained, attackers can dump NTLM password hashes and crack them offline using hashcat, allowing privilege escalation and deeper access into the Artemis network.

**Remediation:** Apply the latest security patches to all RDP-enabled systems and disable public RDP access, requiring VPN connections for remote access. MFA should be enforced for all remote users to prevent unauthorized logins. Login attempts should be monitored for unusual activity, account lockout policies enforced, and RDP access restricted to specific trusted IP addresses to reduce exposure.

**CVSS Score:** 9.8 (Critical)

**2. Web Application Vulnerable to SQL Injection**

**Description:** SQL Injection is a critical vulnerability that allows attackers to manipulate database queries by injecting malicious SQL statements through user input fields. Artemis web applications interact with MySQL, PostgreSQL, and Microsoft SQL Server databases. If the web application does not properly sanitize user input, attackers can directly communicate with the backend database, potentially leading to data theft or administrative control over the system.

**Detection Tools:** Burp Suite, Wapiti, and w3af can scan the Artemis web applications for SQLi vulnerabilities and assess exploitability.

**Risks:** Exploiting SQL injection can extract sensitive customer information, credentials, or proprietary data from Artemis databases. They may also be able to modify or delete critical records, disrupt business operations, or escalate privileges to gain administrative access. Automated tools such as SQLmap can enumerate database structures and extract confidential information. WAFs and input validation can help to mitigate SQLi attacks, but sophisticated obfuscation techniques may allow attackers to bypass. In extreme cases SQLi may lead to remote code execution, giving attackers full control over the application server.

**Remediation:** Implement parameterized queries and stored procedures to ensure user input cannot alter database commands. Enforce input validation to restrict potentially malicious characters. Deploy a WAF such as ModSecurity to help detect and block SQLi attempts in real time. Database servers should follow the principle of least privilege, ensuring web applications have the minimum level of access needed to function. Use automated scanning tools and manual code reviews alongside regular security testing to help identify and remediate SQL injection vulnerabilities before they can be exploited.

**CVSS Score:** 9.0 (Critical)

**3. Default Password on Cisco Admin Portal**

**Description:** Many network devices including Cisco routers, switches, and firewalls come with default administrative credentials that are often left unchanged after deployment. Attackers can use these credentials to gain access to the Artemis network infrastructure. This vulnerability is particularly concerning given Artemis’ reliance on Cisco networking equipment, which plays a critical role in managing data flow across both their industrial and cloud environments.

**Detection Tools:** Nessus and OpenVAS can be used to detect default credentials and misconfigurations in network devices.

**Risks:** Successful login to a Cisco admin portal using default credentials can allow for full control over network configurations, allowing an attacker to modify firewall rules, disable security mechanisms, or create backdoor access for future attacks. Unauthorized access can also be leveraged for network reconnaissance, lateral movement, and data interception. Account lockout policies and monitoring tools can help detect suspicious login attempts, but slow brute-force attacks or credential stuffing using known default passwords may still be successful.

**Remediation:** Immediately change all default credentials on Cisco network devices and enforce strong, unique passwords for administrative accounts. MFA should be enabled wherever possible for an additional layer of security. Unnecessary administrative interfaces should also be disabled to reduce the attack surface. RBAC can also be implemented to ensure that only authorized personnel have access to network management functions. Additionally, regular network audits should be conducted to identify and remediate any misconfigured or weak authentication settings.

**CVSS Score:** 9.0 (Critical)

**4. Apache Web Server Vulnerable to CVE-2019-0211**

**Description:** CVE-2019-0211 is a privilege escalation vulnerability in Apache Web Server versions prior to 2.4.39, which allows an attacker to execute arbitrary code with root-level privileges. This is particularly relevant to Artemis since Apache servers may be used to host internal applications or customer-facing services. Exploitation of this vulnerability can escalate privileges from a low-level web user to a full system administrator.

**Detection Tools:** Nessus and OpenVAS can identify outdated Apache versions and flag vulnerable configurations.

**Risks:** This vulnerability could allow for complete control over the web server, leading to data theft, defacement of company websites, or pivoting into the Artemis internal network. This could also allow attackers to execute arbitrary code, modify web content, or install persistent backdoors for future access. SELinux and hardened Apache configurations can mitigate the risk, but unpatched servers remain vulnerable to privilege escalation attacks.

Remediation: Upgrade all Apache servers to version 2.4.39 or later and apply the latest security patches. Unnecessary Apache modules should be disabled to reduce the attack surface, and access controls should be tightened to limit user privileges. Additionally, regular security assessments that use automated vulnerability scanners should be conducted to ensure compliance with industry best practices.

**CVSS Score:** 8.8 (High)

**5. Web Server Exposing Sensitive Data**

**Description:** Misconfigured web servers can expose sensitive data such as confidential documents, user credentials, or proprietary company files. Access controls that are not properly implemented may allow for unauthorized users to view or download this information.

**Detection Tools:** Burp Suite and Wapiti are security tools that can scan web servers for exposed data and misconfigurations.

**Risks:** Sensitive data leaks may result in financial loss, regulatory penalties, and reputational damage to the company. Exposed data can also be exploited to launch further attacks, such as identity theft or social engineering.

**Remediation:** Restrict public access to sensitive files, enforce proper permissions, and regularly audit exposed endpoints in order to prevent unauthorized access.

**CVSS Score:** 8.0 (High)

**6. Web Application Has Broken Access Control**

**Description:** Broken access control occurs when web applications fail to enforce proper restrictions on authenticated users which can allow for unauthorized access to sensitive resources or administrative functions. This is concerning for internal systems such as SAP ERP, PARS patent submission system, and APOLLO trade secrets repository, which contain confidential business data. Exploiting broken access control can allow attackers to escalate privileges, access restricted data, or manipulate system configurations.

**Detection Tools:** Burp Suite and w3af can be used to identify access control weaknesses in web applications.

**Risks:** Allowing attackers to gain unauthorized access to critical business applications and perform actions meant only for privileged users. This includes viewing confidential financial records, modifying intellectual property filings, or disabling security settings. Forced browsing, manipulated session tokens, or parameter tampering may be used to bypass authentication mechanisms.

**Remediation:** Implementation of RBAC to ensure that users only have access to data and functions necessary for their respective job roles. Web applications should enforce strict authorization checks at the server level, rather than relying solely on client-side enforcement. Session management improvements such as regenerating session tokens upon login and implementing timeout policies can also help to prevent unauthorized access. Lastly, conducting regular access control audits and penetration tests will help to identify and remediate any weak authorization mechanisms before they are exploited.

**CVSS Score:** 8.0 (High)

**7. Oracle WebLogic Server Vulnerable to CVE-2020-14882**

**Description:** CVE-2020-14882 is a remote code execution vulnerability in the Oracle WebLogic Server that allows an unauthenticated attacker to execute arbitrary commands with administrative privileges. This is concerning because WebLogic is often used to host business-critical applications. Exploiting this flaw could allow for full control of affected WebLogic servers, disrupting internal operations.

**Detection Tools:** Nessus and OpenVAS can detect vulnerable WebLogic instances and flag outdated versions.

**Risks:** Allowing remote code execution, enabling attackers to install malware, exfiltrate sensitive data, or pivot to other systems within the network. WebLogic servers often handle enterprise applications and sensitive data, meaning a compromise could lead to financial loss, regulatory violations, and reputational damage. This vulnerability is typically exploited using malicious HTTP requests that target administrative endpoints. Network firewalls and IDS may help to mitigate exploitation attempts, but attackers can use obfuscated payloads to bypass signature-based detection mechanisms.

**Remediation:** Immediately apply the latest Oracle security patches to all WebLogic instances and monitor for signs of exploitation attempts. WebLogic administrative interfaces should be restricted to internal IP addresses only, while unnecessary services should be disabled to minimize exposure. WAFs with strict filtering rules can help prevent malicious HTTP requests from reaching vulnerable endpoints. Finally, implementing continuous security monitoring on WebLogic servers to detect anomalous activity and unauthorized access attempts.

**CVSS Score:** 9.8 (Critical)

**8. Misconfigured Cloud Storage (AWS S3 Buckets, Security Groups)**

**Description:** Misconfigurations in cloud storage can expose sensitive data to unauthorized users, which can lead to potential data breaches. If storage containers are publicly accessible due to misconfigured access control policies, attackers could easily retrieve sensitive files. Insecure AWS security groups can also allow for unintended access to cloud-hosted infrastructures, making it vulnerable to exploitation.

**Detection Tools:** Nessus and OpenVAS can scan for exposed cloud storage and insecure access configurations.

**Risks:** Unaddressed cloud storage misconfigurations can allow attackers to exfiltrate sensitive data including customer information, financial records, proprietary research, and internal documents. A publicly exposed S3 bucket may also contain API keys, credentials, or server configurations, which could allow for privilege escalation and the compromise of additional cloud resources. Misconfigured AWS security groups may also permit unauthorized inbound or outbound network traffic, exposing cloud-based servers to attacks such as brute-force login attempts or remote code execution. While AWS Identity, IAM, and RBAC may help manage access, improper configurations can lead to significant gaps in security posture.

**Remediation:** Enforce strict IAM policies and ACLs to restrict public access to cloud storage and mitigate associated risks. Additionally, security teams should conduct regular audits of cloud storage permissions and ensure sensitive files are encrypted both in transit and at rest. Implementing AWS Security Hub or Azure Security Center can help to enforce compliance with best practices and identify misconfigured resources. Features such as AWS CloudTrail should be enabled for logging and monitoring, to track access attempts and detect suspicious activity. Security teams should also ensure the configuration of network segmentation and least privilege access to minimize the risk of lateral movement within the cloud environment.

**CVSS Score:** 8.5 (High)

**9. Microsoft Exchange Server Vulnerable to CVE-2021-26855**

**Description:** CVE-2021-26855, also known as ProxyLogon, is a critical Microsoft Exchange Server vulnerability which allows remote attackers to bypass authentication and execute arbitrary commands on the mail server. This affects Exchange Server 2013, 2016, and 2019. Artemis relies on Microsoft Exchange for both internal and external email communications, meaning an exploit against this service could lead to a severe security breach.

**Detection Tools:** Nessus and OpenVAS to scan for vulnerable Exchange servers and detect outdated versions that require patching.

**Risks:** Granting attackers unauthorized access to the email system, allowing them to exfiltrate sensitive business communications, steal credentials, or escalate privileges within the corporate network. By using ProxyLogon with additional vulnerabilities, attackers can achieve remote code execution, which could potentially lead to a full domain compromise. Unpatched Exchange servers are also a high-priority target for threat actors. Firewall and endpoint security solutions can detect some exploitation attempts, but attackers may use obfuscated payloads to bypass signature-based defenses.

Remediation: Immediately apply Microsoft’s security patches to all vulnerable Exchange servers. MFA should be enforced for email access, and network segmentation should be implemented in order to limit exposure. Additionally, Exchange logs should be monitored for IoCs such as unauthorized access attempts and abnormal administrative activities. Deployment of IDS/IPS can help detect exploitation attempts in real time. To further reduce risk, consider migrating to Exchange Online (Microsoft 365), which receives automatic security updates and provides built-in threat protection.

**CVSS Score:** 9.9 (Critical)

# **References**

CVE Mitre. (n.d.). *Common vulnerabilities and exposures*. Retrieved from <https://cve.mitre.org>

National Institute of Standards and Technology. (n.d.). *NIST special publications*. Retrieved from<https://csrc.nist.gov/publications/sp>

Open Web Application Security Project. (n.d.). *OWASP top 10*. Retrieved from <https://owasp.org/www-project-top-ten/>

Tenable. (n.d.). *Nessus: Vulnerability scanner*. Retrieved from <https://www.tenable.com/products/nessus>

Greenbone Networks. (n.d.). *OpenVAS: Open vulnerability assessment system*. Retrieved from <https://www.greenbone.net/en/community-edition/>

PortSwigger. (n.d.). *Burp Suite: Web security testing*. Retrieved from <https://portswigger.net/burp>

Graham, R. D. (n.d.). *Masscan: TCP port scanner*. Retrieved from<https://github.com/robertdavidgraham/masscan>

Shodan. (n.d.). *Internet connected device search engine*. Retrieved from<https://www.shodan.io/>

DomainTools. (n.d.). *WHOIS lookup: Domain name search*. Retrieved from <https://whois.domaintools.com/>

Wireshark. (n.d.). *Wireshark: Network protocol analyzer*. Retrieved from<https://www.wireshark.org/>

Microsoft. (2021). *CVE-2021-26855 - ProxyLogon vulnerability*. Retrieved from<https://msrc.microsoft.com/update-guide/vulnerability/CVE-2021-26855>

Apache Software Foundation. (2019). *Apache HTTP server CVE-2019-0211*. Retrieved from<https://httpd.apache.org/security/vulnerabilities_24.html>

Oracle. (2020). *CVE-2020-14882 - WebLogic server vulnerability*. Retrieved from<https://www.oracle.com/security-alerts/>

Cisco. (n.d.). *Cisco security advisories*. Retrieved from <https://tools.cisco.com/security/center/publicationListing.x>

Amazon Web Services. (n.d.). *AWS security documentation*. Retrieved from<https://docs.aws.amazon.com/security/>