Standard Operating Procedure (SOP) – Vulnerability Assessment

To identify, evaluate, prioritize, and recommend remediation for security vulnerabilities across any type of system, network, or application.

Tebogo Matseding
tmatseding@outlook.com

Table of Contents

Standard Operating Procedure (SOP) – Vulnerability Assessment

Purpose:

To identify, evaluate, prioritize, and recommend remediation for security vulnerabilities across any type of system, network, or application.

Applicability:

This SOP applies to web applications, servers, network infrastructure, cloud environments, IoT devices, and other IT assets.

Step 1: Authorization & Rules of Engagement

Obtain written permission from the asset/system owner before testing.

Define rules of engagement:

Testing dates and times.

Systems and components in scope.

Actions that are prohibited (e.g., denial-of-service attacks on production systems).

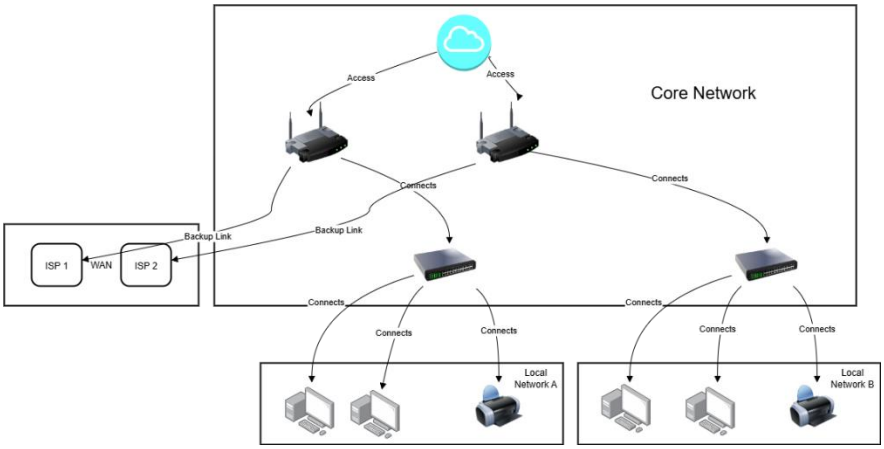
Step 2: Define Scope

Asset Inventory: Identify and list hardware, software, applications, network segments, and data stores to be tested.

Asset ID	Asset Type	Description	Location	Service	Owner	Criticality	Configuration Details	Vulnerability Scope	Testing Methods	Recovery Mechanisms
EDUV-WEB SERVER	WEB-SERVER	handles sensitive client data	In-House	Web Application	Administrator	High	PHP/Node.js MySQL/Postgre SQL TLS (SSL) Enabled	Input validation Session management Access controls Encryption mechanisms	OWASP ZAP for automated scanning Burp Suite for interception and testing OpenVAS for web server vulnerabilities Manual inspection Authentication/auth orization tests	Daily data backups DRP (Disaster Recovery Plan) Web Application Firewall to block attacks

Security Measures: Document existing protections such as firewalls, encryption, access controls, intrusion detection, and patching schedules.

Targets for Testing: Specify components or areas to be assessed (e.g., cloud workloads, endpoints, APIs, database servers, IoT devices).



Example

“For instance, a cloud migration project may include Azure VMs, AWS S3 buckets, and Office 365 accounts as part of the asset inventory.”

Example

“Security measures may include firewall rules that restrict inbound SSH to only the IT team’s IP addresses.”

Step 3: Gather Background Information

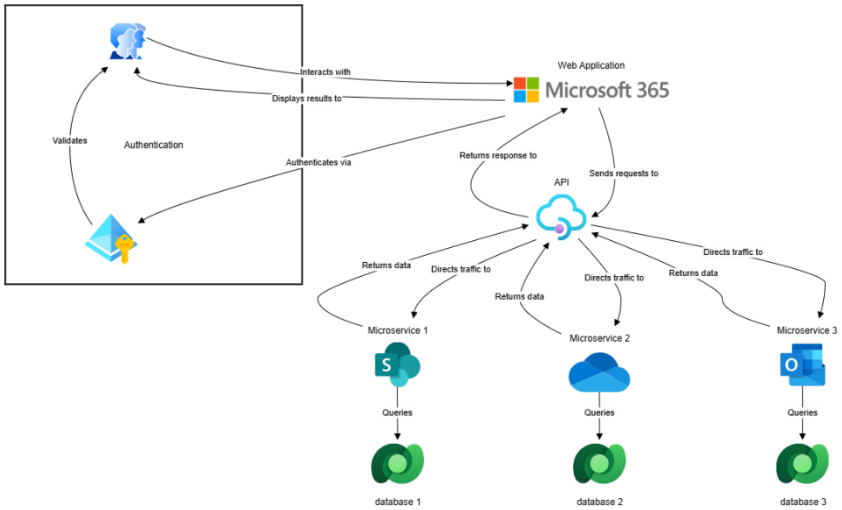
Collect network diagrams, system architecture, and data flow charts.

Review system documentation, configurations, and security policies.

Identify any third-party dependencies that could introduce vulnerabilities.

Example

"If assessing a university system, third-party dependencies might include payment gateways (PayFast, PayPal) or cloud email providers (Google Workspace)."



Step 4: Set Objectives

Compliance Goals – Determine if assessment must meet specific regulations (e.g., PCI-DSS, GDPR, POPIA, HIPAA).

Security Goals – Decide whether focus is on internal threats, external threats, or both.

Risk Tolerance – Understand the acceptable level of risk for the organization.

Example

Compliance Goal: A hospital in South Africa may require POPIA and HIPAA compliance.

Example

Risk Tolerance: A financial institution may set a 'zero tolerance' policy for critical vulnerabilities.

Step 5: Select Tools & Methodologies

Choose tools based on asset type:

Networks – Nmap, Nessus, OpenVAS.

Web Apps – OWASP ZAP, Burp Suite, Nikto.

Servers/Endpoints – Lynis, Microsoft Baseline Security Analyzer, Qualys.

Cloud – ScoutSuite, Prowler, vendor-native security scanners.

Include manual testing to complement automated scans.

Follow frameworks such as OWASP, MITRE ATT&CK, or NIST SP 800-115.

> Example: Nmap can be used to discover that port 3306 (MySQL) is exposed on a production database.

> Example: Burp Suite can intercept and manipulate login requests to test if the app allows SQL injection.

Step 6: Conduct Assessment

Perform Reconnaissance – Gather public and internal information on the target.

Scanning – Identify open ports, services, and exposed assets.

Vulnerability Detection – Use automated and manual techniques to find weaknesses.

Verification – Confirm vulnerabilities to reduce false positives.

Step 7 – Analyze & Prioritize

Rank findings by:

Severity (Critical, High, Medium, Low).

Impact on confidentiality, integrity, and availability.

Likelihood of exploitation.

Use CVSS scoring or a risk matrix to standardize ratings.

(Example for a new web application)

Severity	**Example Vulnerability**	**Impact**	**Likelihood**	**Priority**
Critical	SQL injection exposing student/customer data	Full database compromise	High	1
High	Insecure admin panel without MFA	Unauthorized admin access	Medium	2
Low	Outdated JavaScript library with known exploits	Potential XSS or data leak	Medium	3
Medium	Missing security headers on non-sensitive pages	Minor security hardening required	Low	4

Step 8 – Reporting

Executive Summary – For management, highlighting major risks and potential business impact.

Technical Details – For IT/security teams, including vulnerability descriptions, proof of concept, affected assets, and screenshots.

Remediation Guidance – Clear, actionable steps to fix each issue.

Example: “An Executive Summary might say: ‘During testing, 3 high-risk vulnerabilities were identified that could allow unauthorized access to sensitive customer data. Immediate remediation is recommended.’”

	Negligible	Minor	Moderate	Significant	Severe
Very likely	Low - Medium	Medium	Medium - High	High	High
Likely	Low	Low - Medium	Medium	Medium - High	High
Possible	Low	Low - Medium	Medium	Medium - High	Medium - High
Unlikely	Low	Low - Medium	Low - Medium	Medium	Medium - High
Very unlikely	Low	Low	Low - Medium	Medium	Medium

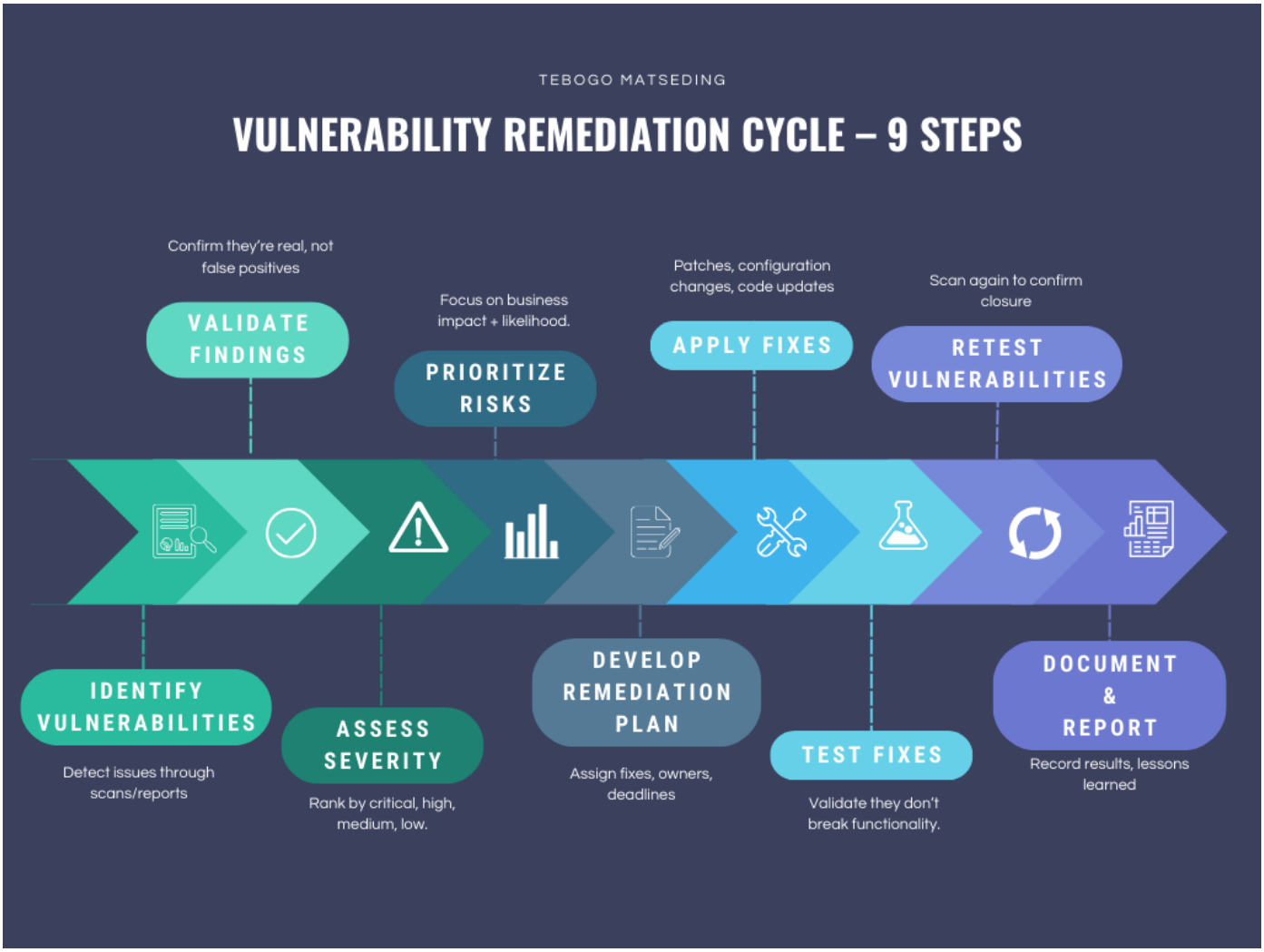
vulnerability assessment executive summary example

Step 9 – Remediation & Retesting

Collaborate with relevant teams to implement fixes.

Retest to confirm issues are resolved.

Document any vulnerabilities that remain (with business justification if not fixed).



Step 10 – Post-Assessment Review

Archive all findings, reports, and scan data.

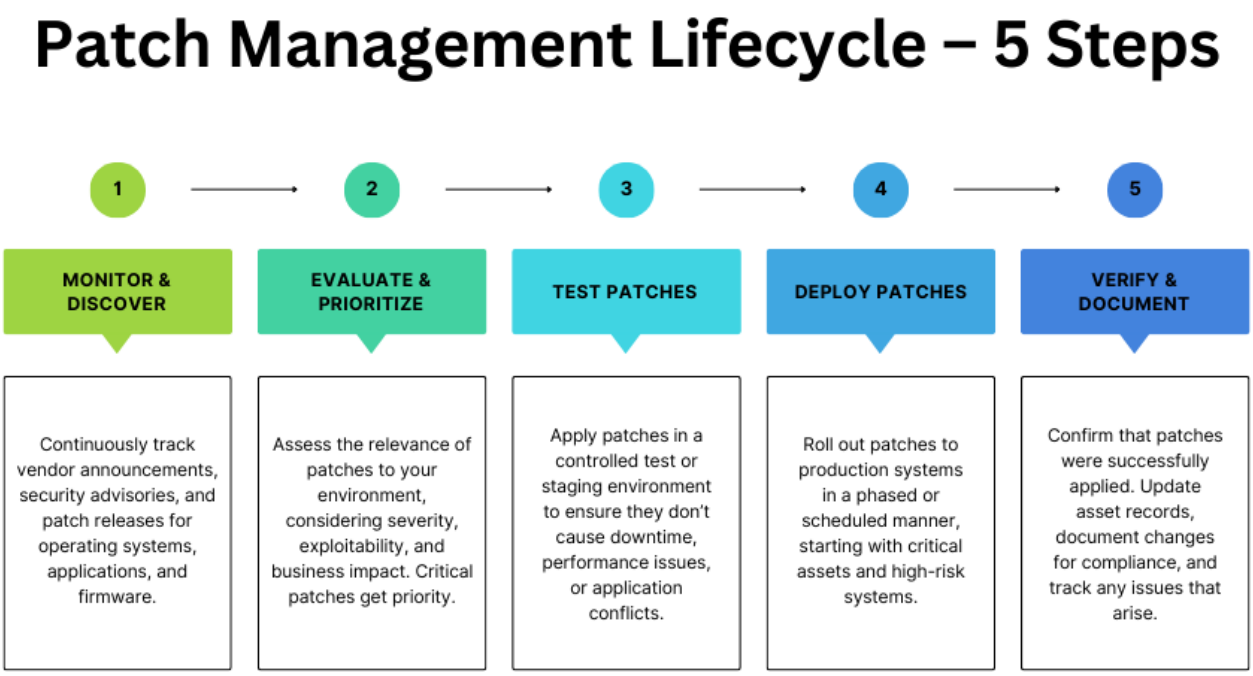
Identify lessons learned to improve the next assessment cycle.

Update SOPs, tools, and methodologies as needed.

Patch Management Lifecycle

Purpose:

To ensure systems remain protected against known vulnerabilities by applying patches in a structured, controlled, and repeatable process.



Conclusion

This SOP outlined a clear process for conducting vulnerability assessments and ensuring secure systems:

Authorization & Scope – Always begin with permission and a well-defined scope to avoid unintended impact.

Background & Objectives – Collect system details and align goals with compliance, security priorities, and risk tolerance.

Tools & Assessment – Use a mix of automated scanners and manual testing to uncover real vulnerabilities.

Analysis & Prioritization – Rank findings by severity, impact, and likelihood, focusing first on critical risks.

Reporting – Provide both executive summaries and technical details to ensure stakeholders understand the risks.

Remediation & Retesting – Apply fixes, verify effectiveness, and document remaining issues with justification.

Patch Management – Regular monitoring, testing, and deployment of patches help prevent vulnerabilities from recurring.

Continuous Improvement – Lessons learned from each cycle should update procedures, strengthen defenses, and improve efficiency.

By following these steps, organizations can build a consistent and repeatable approach to vulnerability management, reducing risk and supporting long-term security resilience.

References

(NIST), National Institute of Standards and Technology, 2024. *The NIST Cybersecurity Framework (CSF).* [Online]
Available at: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf
[Accessed 17 August 2025].

Carnegie Mellon University / CISA, 2016. *Volume 4: Vulnerability Management (CRR Resource Guide).* [Online]
Available at: https://www.cisa.gov/sites/default/files/publications/CRR_Resource_Guide-VM_0.pdf
[Accessed 17 August 2025].

OWASP, n.d.. *OWASP Vulnerability Management Guide.* [Online]
Available at: https://owasp.org/www-project-vulnerability-management-guide/
[Accessed 17 August 2025].

OWASP, n.d.. *Web Security Testing Guide v4.0. Open Web Application Security Project.* [Online]
Available at: https://owasp.org/www-project-web-security-testing-guide/assets/archive/OWASP_Testing_Guide_v4.pdf
[Accessed 17 August 2025].

Secureframe, 2025. *A Step-by-Step Guide to Vulnerability Management.* [Online]
Available at: https://secureframe.com/blog/vulnerability-management
[Accessed 17 August 2025].

Souppaya, M. S. K. &. C. A., 2013. *Guide to Enterprise Patch Management Technologies (SP 800-40 Rev. 3). NIST Special Publication.* [Online]
Available at: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-40r3.pdf
[Accessed 17 August 2025].

Souppaya, M. S. K. &. C. A., 2022. *Guide to Enterprise Patch Management Planning (SP 800-40 Rev. 4). NIST Special Publication.* [Online]
Available at: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-40r4.pdf
[Accessed 17 August 2025].