PROJECT PHOENIX REPORT

Team 404notfound
Members: Sai Sindhu Javvaji, Sai Bindu Javvaji, Sadia, Hunter, Toufiq Umar
Introduction to Cybersecurity
Date: November 14 2025

1. Executive Summary

Project Phoenix demonstrates a fully automated cybersecurity risk analysis pipeline that transforms raw vulnerability scanner output into a prioritized, business-aligned risk register. Using Kali Purple (GVM) and Metasploitable 2 inside an isolated VirtualBox lab, our team emulated a realistic enterprise environment and executed a “Full & Fast” vulnerability assessment.

We then engineered a Python-based processing engine that:

- Parses GVM XML output
- Maps each finding to assets from a structured inventory
- Converts CVSS scores into likelihood
- Calculates risk using a repeatable scoring model
- Produces a sorted, executive-ready risk register

This automation replicates the workflow of modern SOCs and vulnerability management programs, reducing manual labor while increasing accuracy, traceability, and consistency.

2. Team Workflow

Phase 1 — Environment Setup

- Created a host-only VirtualBox network (192.168.56.0/24)
- Deployed:
  - **Kali Purple** as the dedicated scanning appliance
  - **Metasploitable 2** as the vulnerable target host
- Tested network reachability and service discovery

Phase 2 — Scanning & Data Capture

- Configured GVM Targets, Credentials, and Tasks
- Executed “Full and Fast” vulnerability scan
- Exported **XML report** for machine processing

Phase 3 — Automated Risk Engine (Python)

- Parsed XML using ElementTree
- Extracted host, vulnerability, description, CVSS, and port
- Mapped findings to Level-3 enterprise asset list
- Generated **risk_register.csv**, sorted by risk score

Phase 4 — Reporting & Analysis

- Created executive-level briefing
- Summarized top risks and business impacts
- Compared automated vs manual classification

3. CVSS → Likelihood Model (with Professional Justification)

CVSS Score	**Likelihood**	**Rationale**
7.0–10.0	**5 (High)**	Actively exploited in the wild; low complexity; attacker tools readily available
4.0–6.9	**3 (Medium)**	Requires some skill or prereqs; still moderately likely
0.1–3.9	**1 (Low)**	Limited exploit utility; environment-specific

4. Threat Actor Mapping Table (Professional-Grade)

Vulnerability	**Threat Actor**	**TTP Used**	**ATT&CK Mapping**	**Business Impact**
VSFTPD Backdoor	Ransomware gangs	Initial Access → Remote Shell	TA0001 / TA0002	Full system takeover, data theft
OpenSSH RCE	APT actors	Exploitation → Credential Harvesting	TA0006 / TA0003	Privilege escalation, lateral movement

Directory Traversal	Web exploit kits	Discovery → File Exfiltration	TA0007	Exposure of config files, secrets
Weak SMB Credentials	Internal threat	Brute force → Share Access	TA0008	Loss of sensitive documents

5. Risk Matrix (Industry Format)

Likelihood →	**Low (1)**	**Medium (3)**	**High (5)**
Impact ↓			
5 – Critical	**M**	**H**	**H (TOP PRIORITY)**
4 – High	**M**	**M**	**H**
3 – Medium	**L**	**M**	**M**
2 – Low	**L**	**L**	**M**
1 – Minimal	**L**	**L**	**M**

Top 3 risks all fall under **Critical Impact × High Likelihood → "HIGH RISK"**.

6. Top 3 Risks (Deep, Professional Analysis)

1. VSFTPD 2.3.4 Backdoor

Risk Score: 25 (Critical)

- **Impact:** Remote root access → complete server ownership.
- **Exploitation:** Public exploit code, trivial to use.
- **Business Threat:** Breach of customer data, ransomware payload delivery.
- **Recommended Action:**
    - Remove vulnerable package immediately

    - Reimage host
    - Perform credential rotation

---

2. OpenSSH Remote Code Execution

Risk Score: 25 (Critical)

- **Impact:** Attackers can run arbitrary commands remotely.
- **Threat:** Lateral movement across entire network.
- **Recommended Action:**
    - Upgrade OpenSSH
    - Restrict SSH to known jump hosts
    - Enable MFA + logging enhancements

---

3. Apache Directory Traversal

Risk Score: 25 (Critical)

- **Impact:** Unauthorized file disclosure; config file exposure.
- **Threat:** Attackers steal credentials → pivot deeper.
- **Recommended Action:**
    - Patch Apache
    - Harden filesystem permissions
    - Conduct web application input validation review

---

9. Automated vs Manual Analysis

Manual Day Zero Work:

- We only found a few obvious vulnerabilities.
- No scoring system or priority order.
- Hard to repeat the same results again.

Automated Pipeline:

- Collected *all* vulnerabilities from the GVM scan.
- Automatically matched them to assets and owners.
- Gave each issue a clear risk score.

- Results are consistent, repeatable, and faster.
- Reduced the work from hours to a few seconds.

Conclusion:
Manual analysis helps us understand concepts, but the automated system is much better for accuracy, speed, and real-world security operations.

10. Recommendations & Next Steps

High Priority (Next 7 Days):

- Patch all high-risk vulnerabilities immediately.
- Remove insecure services like FTP and upgrade SSH.
- Improve network segmentation to limit attacker movement.

Medium Priority (7–30 Days):

- Enable SMB signing and strong password rules.
- Set up centralized logging for better monitoring.

Long Term (30–90 Days):

- Run vulnerability scans regularly.
- Follow CIS benchmarks for system hardening.
- Create or update the incident response plan.