Project2

AI-Assisted Web Attack Detection and Automated Findings Report System

A hands-on security project using DVWA, Kali Linux, and a Python-based AI rule engine

Course: CTE 561 - Cybersecurity

Project Overview

This project requires you to build a complete web attack detection system from scratch. You will use three real-world attack tools **(Nmap vulnerability scanner, Hydra, Dirb ...)** already installed on Kali Linux to attack DVWA that runs locally on KALI machine. A Python-based AI engine will monitor the attack traffic in real time, classify each attack by type, assign a threat score, and automatically generate a structured Findings Report at the end of the session.

The 'AI' component is a rule-based expert system. When multiple rules fire together on the same IP address, the combined score gives a high-confidence classification of the attack type. The system then formats all findings into one clean report without any manual effort from you.

The AI Detection Engine

The AI engine is a Python script that reads the DVWA Apache access log in real time. For every IP address it sees, it builds a behavioral profile and runs six detection rules against it. Each rule that fires adds points to a threat score between 0 and 100. The combined score determines both the threat classification and the automated response.

1 - The Six Detection Rules

These rules are the 'intelligence' of the system. Each one encodes knowledge that a security analyst would use when manually reviewing logs.

Rule	Attack Type Detected	Trigger Condition	Score Added

Rule 1	Brute-Force Login	More than 5 failed logins in 60 seconds from one IP	+40 points
Rule 2	Vulnerability Scan	More than 30 unique paths visited in 60 seconds	+35 points
Rule 3	Sensitive Path Probing	Requests to /sqli, /exec, /phpmyadmin, /.htaccess, /backup	+20 pts each (max 40)
Rule 4	Directory Scanning	More than 20 HTTP 404 responses in 30 seconds	+30 points
Rule 5	High Request Rate	More than 15 requests in any 10-second window	+25 points
Rule 6	Suspicious User-Agent	User-Agent header contains: Nmap, Hydra, sqlmap, curl, bot	+30 points

2 - Threat Score to Decision

Once the score is computed, the engine makes an automated decision. This graduated response avoids blocking legitimate users while maximizing the cost to attackers.

Score Range	Threat Level	Action Taken
0 – 29	SAFE	Log only. Normal user traffic.
30 – 49	SUSPICIOUS	Print alert to terminal. Continue monitoring.
50 – 74	THREAT	Print alert. Log to findings file. Flag for report.
75 – 100	CRITICAL	Auto-block IP via iptables. Log. Include in report with full detail.

3 - How the AI Classifies Attack Type

When multiple rules fire on the same IP, the engine labels the session with the highest-confidence attack type. For example: if Rule 1 (brute-force) fires for 40 points and Rule 5 (high rate) fires for 25 points, the total score is 65 (THREAT level) and the attack type is labelled 'Brute-Force Login'. If Rule 2 and Rule 6 both fire and the User-Agent contains Nmap, the type is labelled 'Vulnerability Scan' with high confidence. The rule with the highest individual contribution determines the label.

The AI-Generated Findings Report

This is the most important part of the project. After all attacks are complete, you run a second Python script. It reads all the detection logs, aggregates every finding across all three attack tools, and writes one structured Findings Report. You do not write this report manually.

1 - What the Report Contains

The auto-generated report includes the following sections:

1. Executive Summary: total attacks detected, highest threat score, number of IPs blocked.
2. Attack Timeline: every detected event in chronological order with timestamp, IP, score, and type.
3. Per-Attack-Tool Findings: one section for Hydra, and one for Dirb. Each includes: rules fired, peak score, duration of attack, and evidence (sample log lines).

4. Threat Score Distribution: a table showing how scores were distributed (how many SAFE / SUSPICIOUS / THREAT / CRITICAL events).
5. Blocked IPs: list of IPs that reached CRITICAL threshold, with the exact timestamp and rule combination that triggered the block.
6. Recommendations: auto-generated per attack type. Brute-force → recommend account lockout. Scanning → recommend web application firewall. Directory enumeration → recommend 404 normalizations.

Required

1- Video showing the complete solution output
2- A document with the complete solution design and components with screenshots show casing the solution.