Advanced Persistent Threat Simulation & Detection Framework

Alignment: Offensive and Defensive Security (Red + Blue Team)

---

1. Objective

To simulate a full-cycle Advanced Persistent Threat (APT) in a controlled lab environment and detect its behavior using SIEM tools, log analysis, and custom Sigma detection rules. This project integrates payload development, post-exploitation persistence, endpoint logging (Sysmon), log shipping (Winlogbeat), threat detection via ELK Stack (ElasticSearch + Kibana), and alert automation.

---

2. Lab Architecture

2.1 Virtual Environment

- **Attacker Machine:** Kali Linux (2024.1 Rolling)
- **Victim Machine:** Windows 10 (x64, fully updated)
- **Hypervisor:** VMware Workstation
- **Networking Mode:** Host-Only Adapter
- **Static IP Assignment:**
  - Kali: `192.168.56.101`
  - Windows 10: `192.168.56.102`

2.2 Tools Used

Tool	Purpose
`msfvenom`	Payload generation
`Metasploit (msfconsole)`	Remote exploitation, persistence
`Sysmon`	Advanced endpoint logging
`Winlogbeat`	Log shipper for Windows logs
`ElasticSearch`	Storage and indexing of logs
`Kibana`	Log visualization and alerting
`Sigma`	Detection rule framework
`Uncoder.io`	Sigma → JSON rule conversion
`PowerShell`	Registry persistence validation

3. Execution Phases

Phase 1: Environment Setup and Target Discovery

Objective: Establish a controlled, isolated test environment for simulating an APT (Advanced Persistent Threat) scenario.

Steps Taken:

- Configured VMs using VMware Workstation:
  - **Attacker:** Kali Linux
  - **Victim:** Windows 10
- Configured static IPs for both machines in Host-Only network mode.
- Verified connectivity using ipconfig, ping, and netstat -an commands.
- Installed required tools:
  - On Kali: msfconsole, unicorn, Sigma, ElasticSearch, Kibana
  - On Windows 10: Sysmon, Winlogbeat

Validation:

- Successful bidirectional ping between Kali and Windows 10
- Verified system info, user identity, and open ports

Phase 2: Payload Creation and Delivery (APT Simulation)

Objective: Simulate an APT-style attack using Metasploit to gain remote access to the Windows 10 target.

Steps Taken:

- Created a malicious executable (evil.exe) using msfvenom with a reverse TCP payload.
- Transferred the payload to the victim machine via USB.
- Started Metasploit handler:
  - use exploit/multi/handler
  - Set payload, LHOST, LPORT accordingly
- Executed the payload on Windows 10.

Outcome:

- Gained a Meterpreter session with SYSTEM privileges
- Captured system information, user identity, running processes
- Maintained persistence using startup.exe scheduled via registry autoload

Evidence Collected:

- Screenshots of payload execution, handler session, sysinfo, getuid, process list

---

Phase 3: Persistence and Privilege Validation

Objective: Ensure the payload survives reboots and maintains access for further exploitation.

Steps Taken:

- Generated a persistent payload (`startup.exe`) using Metasploit persistence module
- Uploaded the executable to the startup directory
- Verified successful callback after reboot

Post-Exploitation Commands:

- `getuid` → confirms NT AUTHORITY\SYSTEM
- `getenv` → environment variable listing
- `shell` → dropped into cmd.exe shell

Phase 4: Windows Logging Configuration

Objective: Enable in-depth monitoring and log forwarding for detection.

Sysmon:

- Installed Sysmon with `sysmon -accepteula -i sysmonconfig.xml`
- Enabled process creation, network connection, file creation, and registry event logging

Winlogbeat:

- Configured `winlogbeat.yml` to forward logs to ELK (Kali)
- Configured event log sources: Security, Sysmon, Application, System
- Forwarded logs to port 5044 (Logstash)

ELK Stack (Kali):

- Set up Elasticsearch and Kibana
- Applied custom dashboards for event monitoring

Validation:

- Confirmed Sysmon logs captured payload execution
- Kibana dashboard displayed relevant alerts

Phase 5: Log Analysis and Detection Rule Development

Objective: Analyze logs and write custom Sigma rules to detect the APT behavior.

Steps Taken:

- Extracted Sysmon logs from Kibana
- Noted suspicious behaviors:
  - Execution from unusual directories
  - Creation of persistence payload
  - Network connections to Kali attacker IP
- Wrote Sigma detection rule (startup-exe-creation.yml):
  - Detects registry or file changes in startup directories
  - Mapped to MITRE ATT&CK T1059: Command and Scripting Interpreter
- Converted YAML to JSON using Uncoder.io
- Uploaded rule to winlogbeat/sigma_rules/
- Restarted Winlogbeat to apply rule

Validation:

- Custom rule triggered when test payload was executed again
- Alert shown in Kibana with appropriate timestamp and host details

---

Phase 6: SIEM Alerting & Threat Intelligence Enrichment

Objective: Implement basic alerting and threat enrichment capabilities in the detection system.

Steps Taken:

- Enabled alerting in Kibana for detection rule hits
- Configured basic email notifications for high severity triggers
- Explored integration with:
  - MITRE ATT&CK heatmaps
  - Future MISP/TAXII threat intelligence feed plans
- Documented Indicators of Compromise (IOCs):
  - File paths: C:\Users\<user>\AppData\Roaming\startup.exe
  - Hashes: (if collected)
  - IP: Attacker machine's IP

Outcome:

- Demonstrated a full detection-to-alert loop
- Alert details available on dashboard

---

4. MITRE ATT&CK Mapping

Technique ID	Name
T1059	Command & Scripting Interpreter
T1053.005	Scheduled Task/Startup Persistence
T1105	Ingress Tool Transfer
T1027	Obfuscated Files or Information
T1055	Process Injection (potential if payloads modified)

---

5. Outcome Summary

This project successfully simulated an APT attack, achieved persistence, and executed real-time detection via ELK Stack. It bridges offensive security (red team payload delivery) and defensive capabilities (SIEM integration, detection engineering, threat intel enrichment). The simulation replicates modern attacker behavior and provides a full "attack-to-alert" visibility pipeline.

---

6. Skills Demonstrated

Exploitation – Payload creation, reverse shells, registry persistence
Blue Teaming – Sysmon configuration, Winlogbeat setup, ELK dashboarding
Detection Engineering – Sigma rule creation, log forensics
Threat Intelligence – IOC tracking, MITRE ATT&CK mapping
Documentation – Structured multi-phase incident report
