Project Proposal: Malicious Code and Solution Implementation

Course: Introduction to Cybersecurity
Instructor: Prof. PICH Reatrey
Date: November 25, 2025

1. Team Members & Task Division

No.	Name	Role	Specific Task Assignment
1	**Loem Kimhour**	**Team Leader**	**Project Manager & System Architect** • **Core Malware Architecture:** Creating the `main.py` skeleton that calls the attack modules. • **Core Anti-Malware Architecture:** Creating the `defense_engine.py` skeleton that integrates the detection modules. • **Environment:** Setting up the VirtualBox Network.
2	**Lorn Thornpunleu**	**Red Team**	**Delivery Specialist** • Implementing HTML Smuggling & LNK Spoofing.
3	**Chut Homey**	**Red Team**	**Persistence Specialist** • Implementing Registry Keys & Scheduled Tasks.
4	**Ly Kimkheng**	**Red Team**	**Lateral Movement Specialist** • Implementing SMB Worm & USB Replication.
5	**Te Sakura**	**Blue Team**	**Anti-Delivery Specialist** • Developing File Signature Scanner & Script Analyzer.
6	**Panha Viraktitya**	**Blue Team**	**Anti-Persistence Specialist** • Developing Registry Watchdog & Task Auditor.
7	**Penh Sovicheakta**	**Blue Team**	**Anti-Spreading Specialist** • Developing Network Port Monitor & USB Sentinel.

2. Project Introduction

This project aims to simulate a realistic cybersecurity conflict by developing two opposing software artifacts:

1. **The Malware:** A single malicious executable designed to perform three distinct destructive actions (The Attack).
2. **The Antivirus:** A single defensive executable designed to detect, block, and neutralize the malware (The Solution).

The project will be executed in a strictly isolated Virtual Machine environment to ensure safety while allowing us to study the lifecycle of modern "Double Extortion" and "Wiper" malware.

3. The Malicious Code Implementation (Red Team)

Artifact Name: chimera.exe
Concept: A hybrid Ransomware-Wiper that targets the availability and confidentiality of victim data.

A. The 3 Core Attack Methods (Inside the Malicious File)

1. **File Encryption (Ransomware):** The code will locate document files (.docx, .pdf) and encrypt them using AES-256, rendering them unreadable.
2. **System Corruption (Wiper):** The code will attempt to delete or corrupt the Windows hosts file to block access to security websites (e.g., blocking access to antivirus.com).
3. **Data Exfiltration (Spyware):** Before encryption, the code will read the first 100 bytes of every document and send them to a simulated attacker server.

B. The Malicious Pipeline (2 Techniques per Stage)

To ensure the malicious code runs successfully, we will implement the following pipeline:

Pipeline Stage	Technique 1	Technique 2
1. Delivery	**HTML Smuggling:** Embedding the EXE inside a JavaScript blob within an HTML file. When opened, the browser "downloads" the malware locally.	**LNK Masquerading:** Creating a Windows Shortcut (.LNK) with a PDF icon. When clicked, it runs a PowerShell script to fetch the malware.
2. Auto-Executing	**Registry Run Key:** Adding an entry to HKCU\Software\Microsoft\Windows\CurrentVersion\Run for persistence on login.	**Scheduled Task:** Creating a hidden Windows Task that triggers the malware every time the system goes idle.
3. Spreading	**SMB Share Copy:** Scanning the local network (Port 445) for open shared folders and copying the malware to them.	**USB Drive Infection:** Detecting inserted USB drives and creating a hidden copy of the malware with an autorun trigger.

4. The Anti-Malicious Code Implementation (Blue Team)

Artifact Name: aegis_defense.exe
Concept: A host-based intrusion detection system (HIDS) specifically tuned to counter chimera.exe.

A. The 3 Core Anti-Methods (Inside the Anti-Malicious File)

1. **Heuristic Encryption Detection:** The code monitors the file system for rapid file modification (high entropy writes) and kills the process if it modifies more than 3 files in 1 second.
2. **System File Integrity Monitor:** The code creates a hash of the Windows hosts file. If the file is changed, it automatically restores the backup version.
3. **Network Egress Filtering:** The code monitors outbound traffic. If an unknown process tries to send data to an unauthorized IP, the connection is dropped (stopping exfiltration).

B. The Anti-Malicious Pipeline (2 Techniques per Stage)

To stop the malware at every step of its lifecycle:

Pipeline Stage	Solution 1	Solution 2

1. Anti-Delivery	**Magic Number Analysis:** A scanner that checks file headers. It detects if a file claiming to be a PDF is actually an LNK or EXE.	**Script De-obfuscation:** A module that scans HTML files for large Base64 encoded strings (indicative of Smuggling) and blocks them.
2. Anti-Execution	**Registry Watchdog:** A background service that locks the Run key. If a new value is added, it alerts the user and deletes it.	**Task Scheduler Audit:** A script that lists all tasks and highlights any task pointing to a file in the Temp or Downloads folder.
3. Anti-Spreading	**SMB Traffic Blocker:** A firewall rule that temporarily blocks Port 445 if it detects more than 5 connection attempts in 1 second.	**USB Auto-Scan:** A service that automatically scans any new USB drive for hidden files and executable extensions before mounting it.

5. Project Action Plan & Timeline

Based on the 5-week schedule.

Week	Phase	Activity	Responsible
Week 1	**Architecture**	• **Leader:** Create the "Empty" Main Malicious File and Main Anti-Malicious File. • **Members:** Research their specific techniques.	**Leader**
Week 2	**Phase 1 Dev**	• **Red Team:** Send Delivery/Execution code to Leader. • **Leader:** Integrate into Malicious File. • **Blue Team:** Send Anti-Delivery code to Leader.	**All**
Week 3	**Phase 2 Dev**	• **Red Team:** Send Spreading code. • **Blue Team:** Send Anti-Execution code. • **Leader:** Update both Main Files.	**All**
Week 4	**Testing**	• **Leader:** Run the completed Malicious File against the completed Anti-Malicious File. • **Blue Team:** Tune the detection sensitivity.	**Leader + Blue**
Week 5	**Final**	• Final Report & Demo.	**All**

6. Selected Technology Stack

- **Malware Language:** Python (Converted to EXE via PyInstaller) – chosen for rapid development of network and file system modules.
- **Anti-Malware Language:** PowerShell & Python – chosen for deep integration with Windows Management Instrumentation (WMI).
- **Target Environment:** Windows 10 (Victim), Kali Linux (C2 Server).
- **Tools:** Wireshark (Traffic Analysis), Process Hacker (Behavior Analysis).