`

T.Y.B.Sc. Computer Science

Semester V

A.Y. 2025 - 2026

Project Proposal

On

Advanced Malware Analysis Tool

Name: Shezan Merajuddin Shaikh

Roll No: 43

`

Title: MalScan – Advanced Malware Analysis Tool

Name: Shezan Shaikh

1. Introduction

Malware threats are evolving rapidly, with attackers employing sophisticated obfuscation and evasion techniques. **MalScan** is a Python-based malware analysis tool designed to automate static and dynamic analysis of suspicious files. It aims to provide cybersecurity professionals and researchers with a lightweight yet powerful solution for dissecting malware behavior, extracting Indicators of Compromise (IOCs), and generating actionable reports.

2. Objectives

Primary Goal:

To develop an automated malware analysis system capable of:

- Performing **static analysis** (file hashing, header inspection, YARA rule matching).
- Conducting **dynamic analysis** (monitoring file, process, and network activity in a sandboxed environment).
- Generating **comprehensive reports** (IOCs, risk scoring, behavioral summaries).

Key Objectives:

1. Achieve **85%+ detection accuracy** for common malware families (e.g., ransomware, trojans).
2. Implement **heuristic analysis** to identify zero-day threats.
3. Ensure **safe execution** via isolated sandboxing.

3. Scope

- **Supported File Types:** PE (Windows), ELF (Linux), scripts (Python, PowerShell).

`

- **Analysis Modes:**
  - **Static:** Structural analysis, entropy checks, string extraction.
  - **Dynamic:** API call tracing, registry monitoring, network traffic capture.
- **Limitations:**
  - No kernel-level analysis (e.g., rootkit detection).
  - Limited to user-mode monitoring.

4. Methodology

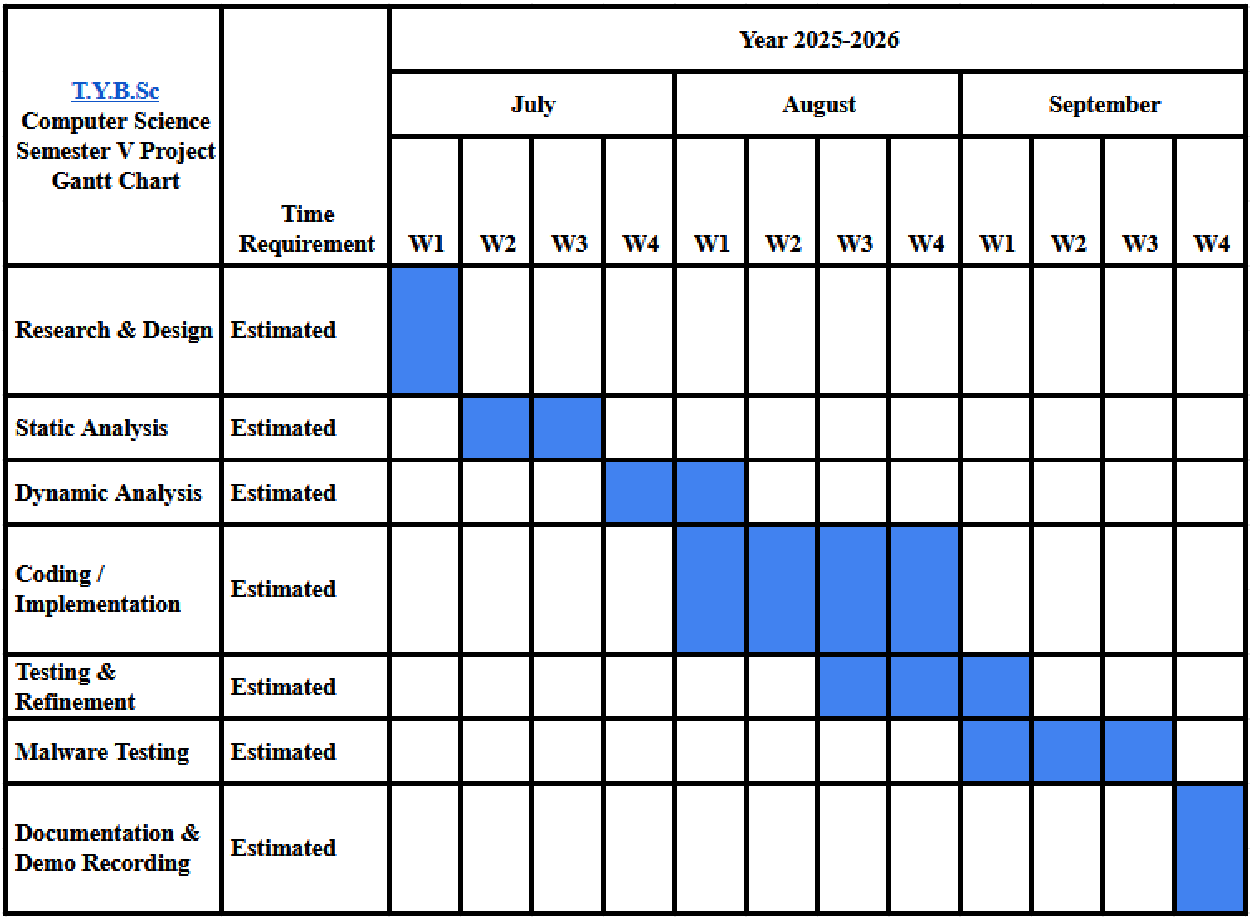
1. **Static Analysis Phase:**
   - File hashing (SHA-256, MD5).
   - PE/ELF parsing (pefile, lief).
   - YARA rule matching (yara-python).
2. **Dynamic Analysis Phase:**
   - Sandboxed execution (Python subprocess + Cuckoo Sandbox integration).
   - Real-time monitoring (ProcMon, psutil, scapy).
3. **Reporting Phase:**
   - JSON/HTML report generation (pandas, Jinja2).

5. Tools & Technologies

Category	**Tools/Libraries**	**Purpose**
Static Analysis	pefile, yara-python, lief	File structure, signature matching
Dynamic Analysis	volatility3, Frida, scapy	Behavior monitoring, memory forensics
Sandboxing	Cuckoo Sandbox, Docker	Safe execution environment
Reporting	Jinja2, pandas	HTML/JSON report generation

`

6. Timeline:



T.Y.B.Sc Computer Science Semester V Project Gantt Chart	Time Requirement	Year 2025-2026											
		July				August				September			
		W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
Research & Design	Estimated	🟦											
Static Analysis	Estimated		🟦	🟦									
Dynamic Analysis	Estimated				🟦	🟦							
Coding / Implementation	Estimated					🟦	🟦	🟦	🟦				
Testing & Refinement	Estimated							🟦	🟦	🟦			
Malware Testing	Estimated									🟦	🟦	🟦	
Documentation & Demo Recording	Estimated												🟦

7. Resources

- **Hardware:** Virtual machines (Windows/Linux), 16GB RAM.
- **Datasets:** Malware samples from MalwareBazaar.
- **References:**
  - "Practical Malware Analysis" by Michael Sikorski.
  - MITRE ATT&CK Framework (attack.mitre.org).

`

8. Expected Outcomes:

1. **Functional CLI Tool**
    - Static analysis (file hashes, headers, YARA rules)
    - Dynamic analysis (processes, registry, network activity)
    - Sandboxed execution
2. **Automated Reports**
    - JSON/HTML outputs with:
        - IOCs (hashes, IPs, C2 domains)
        - Risk score (1-10)
        - Behavior summary
3. **Documentation**
    - Installation guide (Windows/Linux)
    - Sample malware analysis reports
    - API docs for customization
4. **Performance Targets**
    - 85%+ detection rate for common malware
5. **Extensible Design**
    - Supports adding new YARA rules
    - Modular for future upgrades
    -

9. References

1. Ligh, M. H., Adair, S., Hartstein, B., & Richard, M. (2014). *Malware Analyst's Cookbook.* Wiley.
2. Sikorski, M., & Honig, A. (2012). *Practical Malware Analysis.* No Starch Press.
3. Ligh, M. H., Case, A., Levy, J., & Walters, A. (2014). *The Art of Memory Forensics.* Wiley.
4. Microsoft Corporation. (2021). *PE Format Specification.* Microsoft Docs.
5. MITRE Corporation. (2023). *MITRE ATT&CK Framework.* Technical Report.
6. National Institute of Standards and Technology. (2013). *Guide to Malware Incident Prevention and Handling* (NIST SP 800-83 Rev. 1).
7. YARA Project. (2022). *YARA: The Pattern Matching Swiss Knife for Malware Researchers.* Documentation.
8. Cuckoo Foundation. (2021). *Cuckoo Sandbox: Open-Source Automated Malware Analysis.* Technical Whitepaper.