Ex No-10: Use Ghidra to Disassemble and Analyze Malware Code

Aim

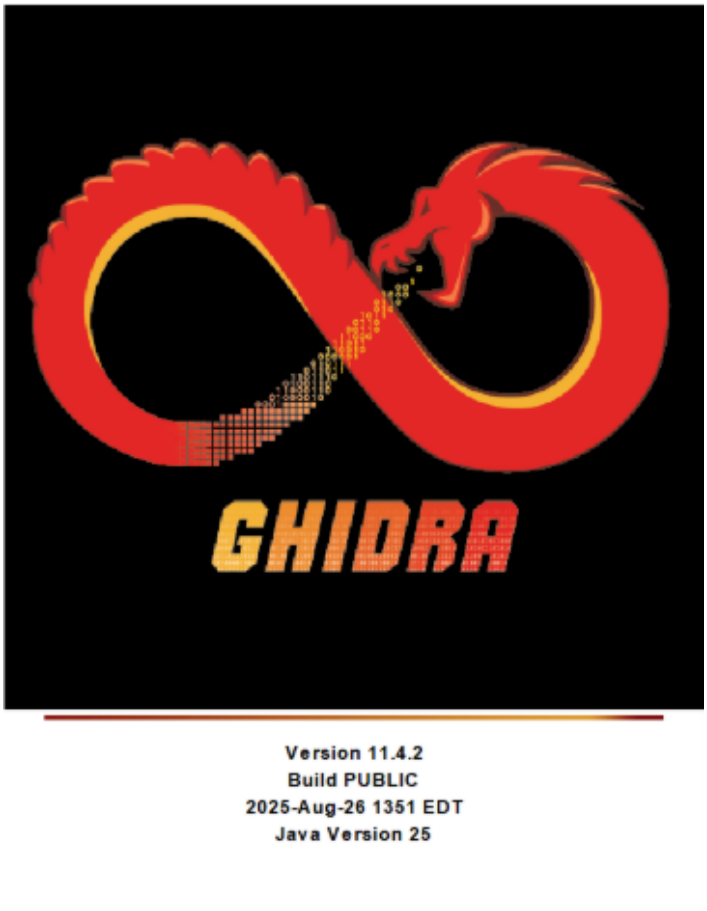
To disassemble and analyze a binary using **Ghidra** to identify potential malicious behavior and understand code functionality.

Description

Ghidra is a free reverse-engineering framework that helps analysts disassemble and decompile binaries to inspect assembly and pseudocode. This experiment covers loading a binary, running auto-analysis, locating key functions and strings, and deriving behavioral indicators.

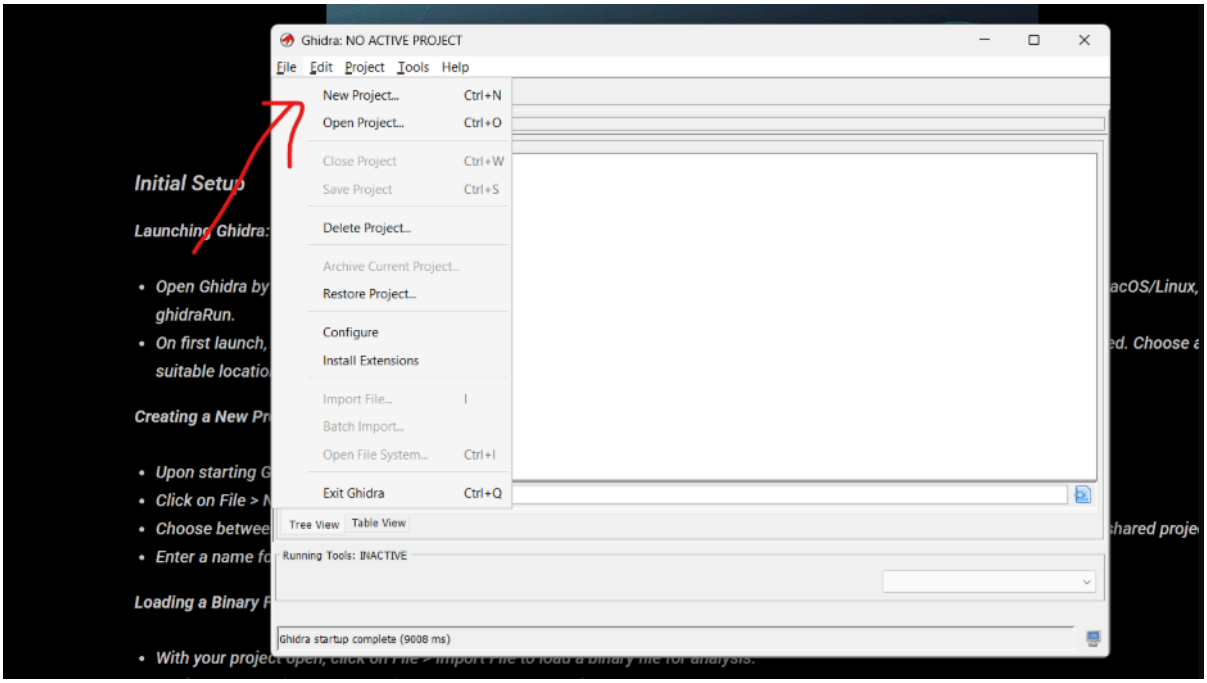
Steps

1. **Prepare Environment**
   - Use an isolated VM (VirtualBox/VMware) with a snapshot.
   - Install Ghidra and required dependencies.



---

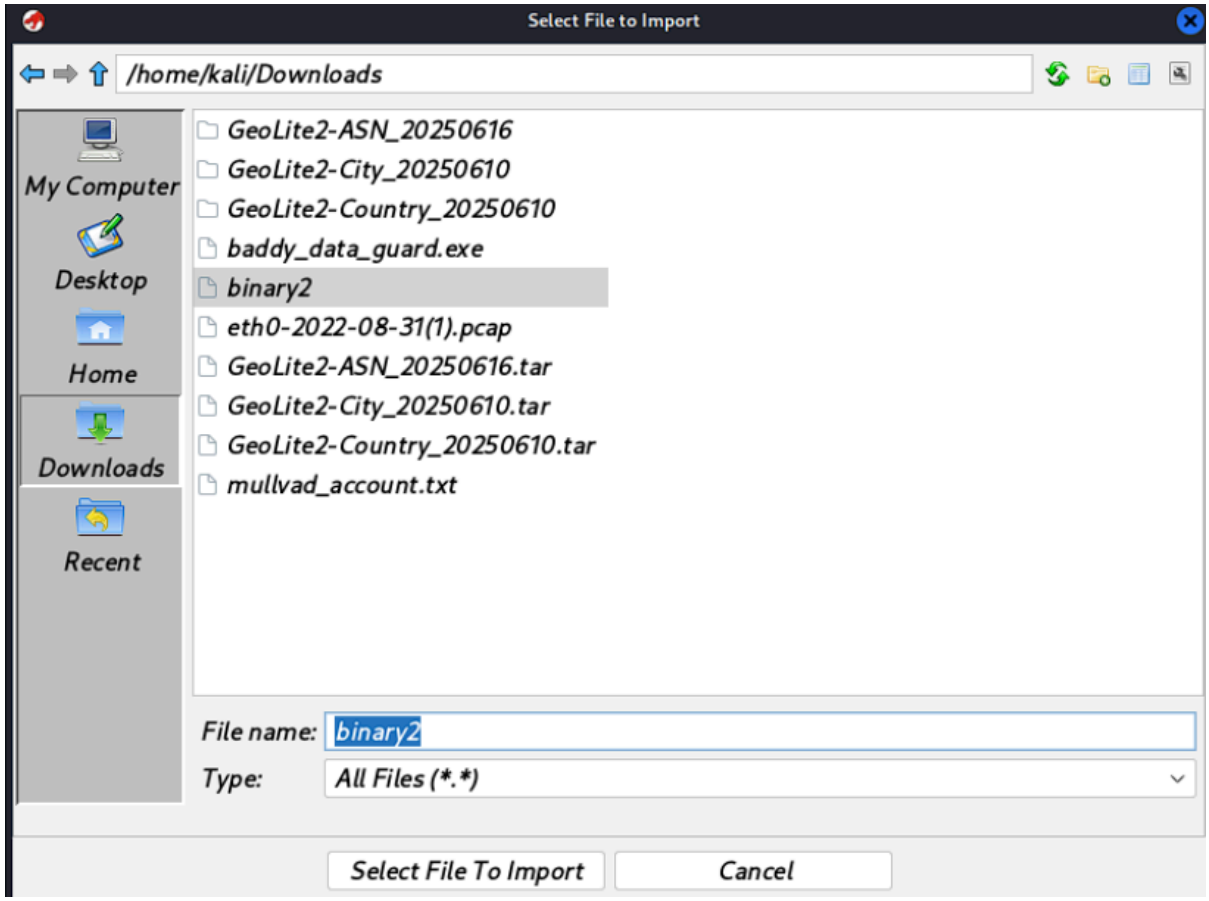
2. **Load Binary**
    - Open Ghidra and create a new project.
    - Import the target binary into the project.



---

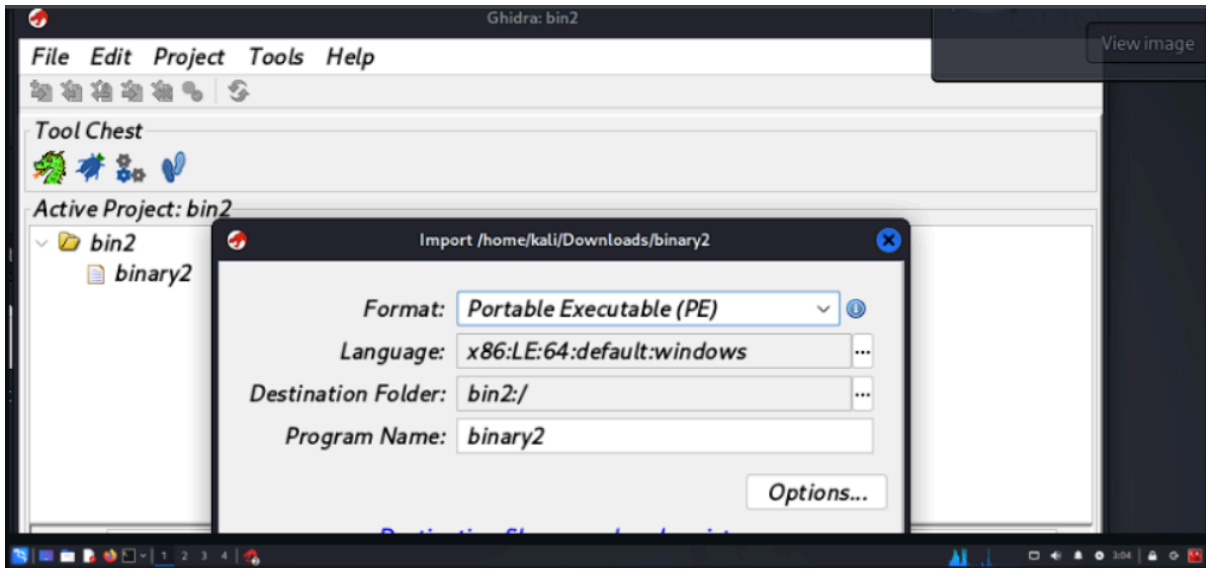
3. **Run Auto-Analysis**
    - Allow Ghidra to perform auto-analysis with default options.

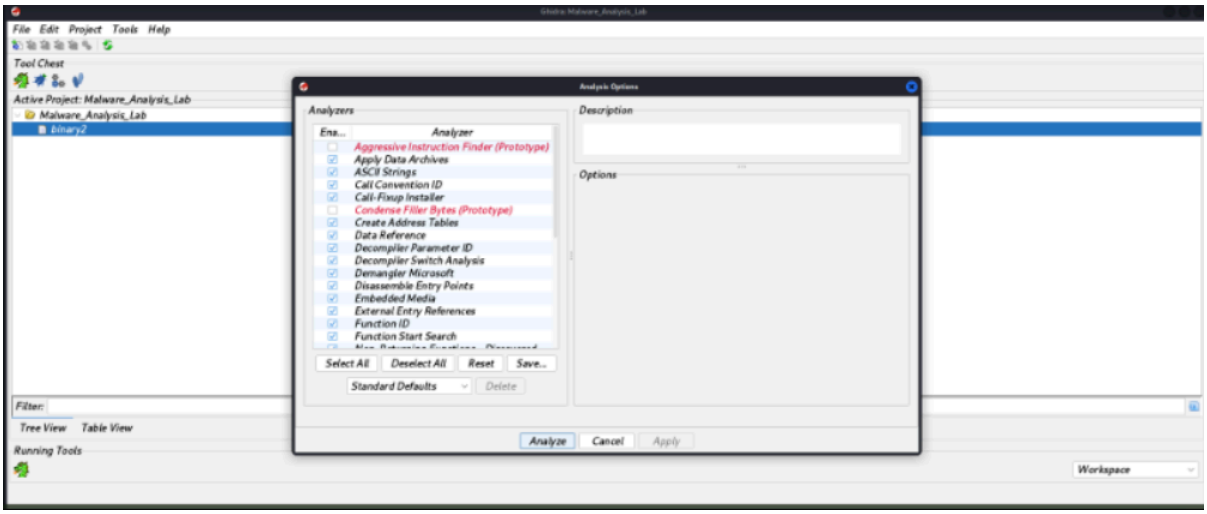
- Review the analysis log for any warnings.



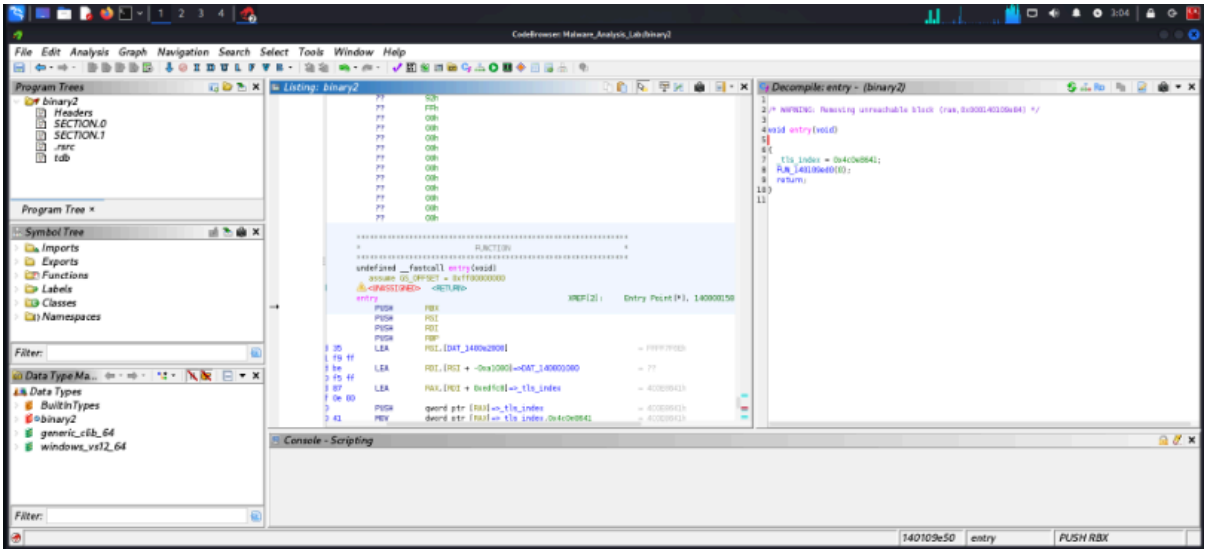
---

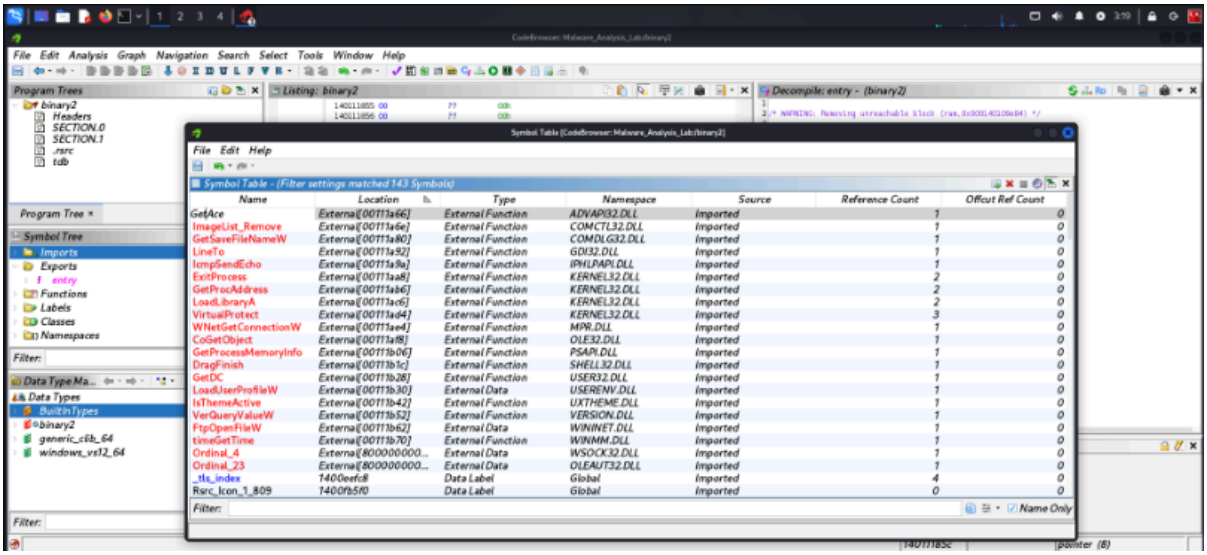
4. **String and Import Inspection**
   - Search for suspicious strings (URLs, file paths, commands).
   - Check imported functions for networking, file, or process APIs.





5. **Function Analysis**
    - Identify interesting functions via cross-references (xrefs).
    - Use the Decompiler to read C-like pseudocode and annotate logic.
    - Rename functions and variables for clarity.





Rubrics

Criteria	Mark Allotted	Mark Awarded
1. GitHub Activity & Submission Regularity	3	
2. Application of Forensic Tools & Practical Execution	3	
3. Documentation & Reporting	2	
4. Engagement, Problem-Solving & Team Collaboration	2	
Total	10	

Result

Loaded the binary into Ghidra, performed static analysis to identify key functions and strings, and produced a summarized report of observed behaviors and indicators of compromise (IoCs).