



**DigiSuraksha**

— Parhari Foundation —

# INTERODUCTION TO MALWARE ANLAYSIS WITH AI

By Digisuraksha Parhari Foundation



[HTTPS://DIGISURAKSHA.ORG/](https://digisuraksha.org/)







# Devil In the Depth



# Malware Analysis: Techniques & Tools

Understanding & Investigating Malicious Software

# Basic Overview Of Malware

## How malware works





# Introduction to Malware Analysis

**Definition:** The process of understanding malware's origin, functionality, and impact.

**Importance:** Helps in forensic investigations, threat intelligence, and system hardening.

**Types of Analysis:** Static, Dynamic, Behavioral,.



# Introduction to Malware Analysis

**Definition:** The process of understanding malware's origin, functionality, and impact.

**Importance:** Helps in forensic investigations, threat intelligence, and system hardening.

**Types of Analysis:** Static, Dynamic, Behavioral, and Memory Analysis. pcap volit



# Malware Analysis Approaches

- **Static Analysis:** Inspecting malware without execution.

**Tools:** Strings, IDA Pro, Ghidra.

- **Dynamic Analysis:** Running malware in a controlled environment.

**Tools:** Process Monitor, Wireshark, Cuckoo Sandbox.

- **Memory Analysis:** Analysing RAM for malware footprints.

**Tools:** Volatility, Rekall.

- **Behavioural Analysis:** Monitoring malware actions.



## **Static Malware Analysis Techniques**

- **Extracting Strings – Identifying embedded text.**
- **Checking Hashes – Comparing against malware databases.**
- **Analyzing PE Headers – Checking file structure.**
- **Disassembling Code – Using IDA Pro/Ghidra to read assembly.**

## **Dynamic Malware Analysis Techniques**

- **Running Malware in VM/Sandbox**
- **Monitoring Process Behavior with ProcMon**
- **Network Activity Analysis using Wireshark**
- **Registry & File System Monitoring**



## Memory Forensics & Volatility

- **Why Memory Analysis?** Extracts malware running in RAM.
- **Volatility Framework:**
- **pslist** – Lists running processes.
- **malfind** – Detects hidden injections.
- **netscan** – Captures network activity.

## Behavioral Analysis & Evasion Techniques

- **Key Indicators of Malware Behavior:**
- **Persistence mechanisms** (Registry, Services).
- **Code Injection techniques.**
- **API Hooking detection.**

## Evasion Techniques Used by Malware:

- **Anti-VM detection.**
- **Obfuscation & Packing.**
- **Time delays to avoid sandboxes.**



# Static Analysis

## Classifications

Ransomware

Threat Names

CryptoLocker

## Sample Information

MD5 :- 279fa384aafbf96cb853e98608cda9fc

SHA1 :- 32ba8feee40212a44a581fd8468fd61ae02ae136

SHA256 :- 861c3cfce77888bb07b0269a976040c04c7e249ebbc038344706444eb108efb

SSDeep :-768:b3o/2n1TCraU6GD1gdcKX4WcO+wMVm+slAMphNuhX3HdAxmIkMl1fEPtI4Y:b4/y2M1oFO+BeghXuMUlt

ImpHash :-5e2ad6a5dfbbe82f8269a72cbbfb3895

File Name :- 2xDUktv0aj437mge.exe

File Size :- 55.13 KB

Sample Type :- Windows Exe (x86-32)



# Malware Analysis





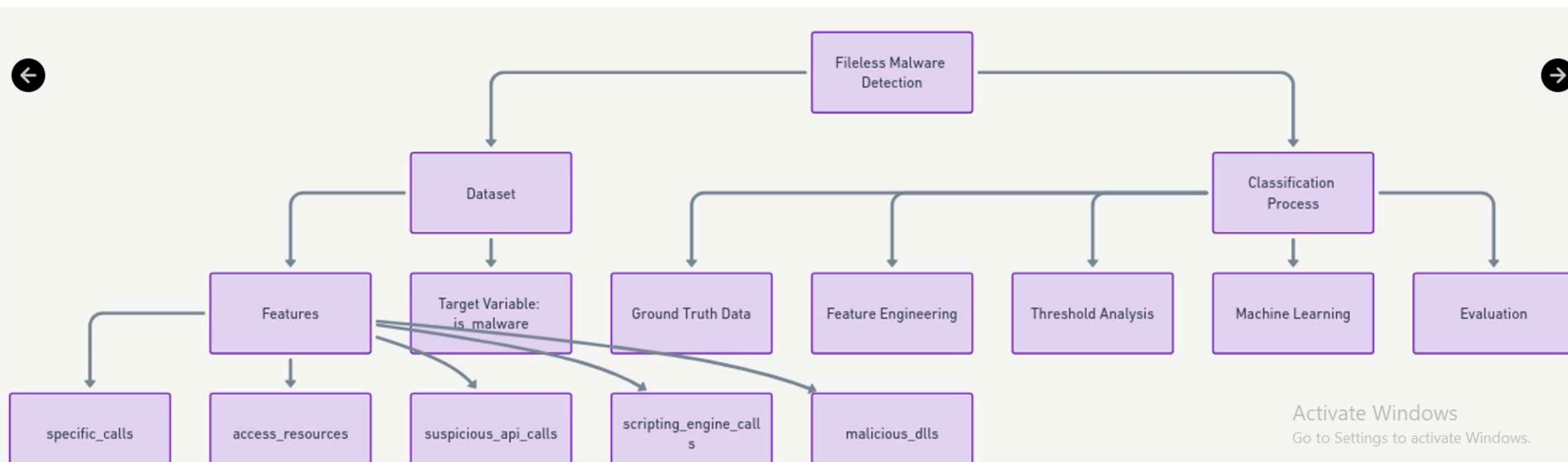
Indicator	Description
Specific Call	Suspicious API or system calls indicative of malicious activity, such as direct memory access or privilege escalation.
Access Resources	Attempts to access protected or restricted system resources without proper authorization.
Suspicious API Calls	Use of APIs often associated with malware behaviors, such as CreateRemoteThread, VirtualAlloc, or WriteProcessMemory.
Malicious DLL	Detection of maliciously crafted Dynamic Link Libraries (DLLs) loaded into legitimate processes.
Scripting Engine Calls	Abnormal use of scripting engines like PowerShell, WScript, or JavaScript to execute commands.
DLL Loading into Processes	Unusual loading of DLLs into processes, potentially indicating process injection or exploitation.
Hijacked Process	Legitimate processes hijacked by malware for execution of malicious payloads.
Unexpected Child Process	Processes spawned unexpectedly, often a sign of command execution or exploitation.
C2 Communication	Evidence of communication with Command and Control (C2) servers for instructions or data exfiltration.
Data Exfiltration	Detection of sensitive data being extracted and transmitted to external systems.
Unusual Memory Access	Anomalies in memory usage, such as code injected into legitimate processes.
Malicious Payload in Memory	Detection of malicious code or payloads residing entirely in memory without writing to disk.
Registry Persistence Changes	Changes to Windows registry entries designed to maintain persistence or evade detection.
Malicious Registry Task	Registry-based tasks configured to execute malicious actions.
PowerShell Activity	Abnormal or obfuscated PowerShell scripts executing commands.
WMI Activity	Suspicious use of Windows Management Instrumentation (WMI) for remote execution or reconnaissance.
JavaScript Activity	Malicious JavaScript executing unauthorized actions in the browser or system.
Unusual Logon Patterns	Login attempts or successful logins from unexpected locations or times.
Privilege Escalation	Actions indicative of attempts to gain higher privileges within a system.
Scheduled Task Changes	Creation or modification of scheduled tasks to maintain persistence.
Behavioral Anomalies	General deviations from normal behavior patterns within the system or network.
Is Malware	Confirmation of fileless malware presence based on combined indicators and behaviors.



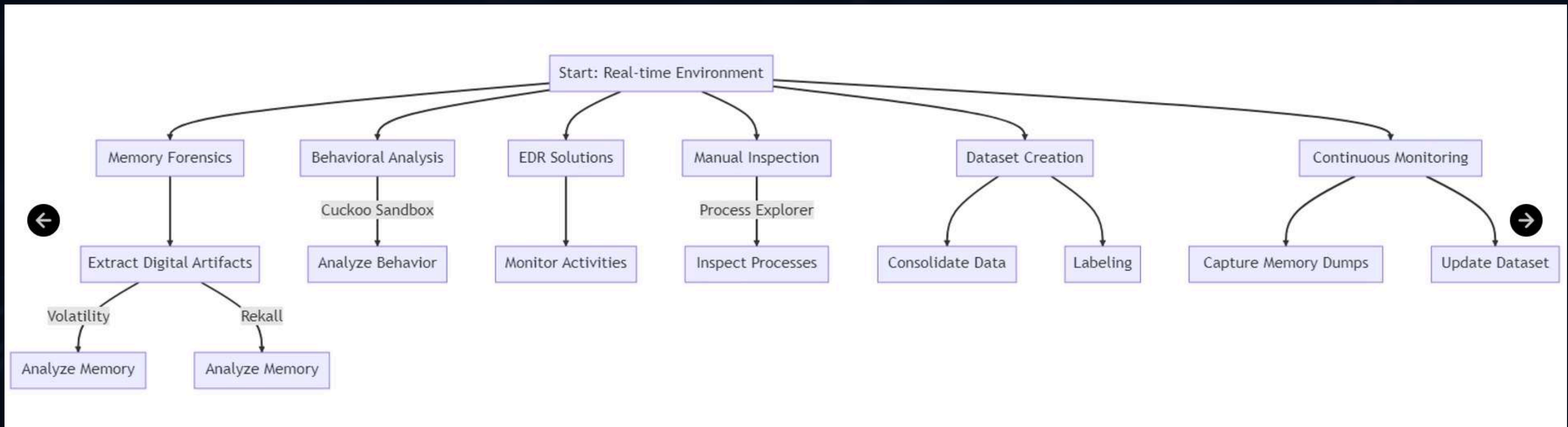
Indicator	Description
Specific Call	Suspicious API or system calls indicative of malicious activity, such as direct memory access or privilege escalation.
Access Resources	Attempts to access protected or restricted system resources without proper authorization.
Suspicious API Calls	Use of APIs often associated with malware behaviors, such as <code>CreateRemoteThread</code> , <code>VirtualAlloc</code> , or <code>WriteProcessMemory</code> .
Malicious File	Detection of files containing malicious code, including executables, scripts, and documents with embedded malware.
Embedded Payloads	Files with embedded payloads that exploit vulnerabilities or deliver malware upon execution.
Code Obfuscation	Files utilizing techniques like encryption or polymorphism to hide malicious intent.
Packaged Malware	Malware packaged within legitimate-looking archives or installers to evade detection.
Suspicious File Extensions	Files with uncommon or misleading extensions that could indicate malicious intent (e.g., .exe disguised as .jpg).
File Dropping Activity	Malware that drops additional files onto the system for further execution or exploitation.
Malicious DLL	Detection of maliciously crafted Dynamic Link Libraries (DLLs) loaded into legitimate processes.
Scripting Engine Calls	Abnormal use of scripting engines like PowerShell, WScript, or JavaScript to execute commands.
DLL Loading into Processes	Unusual loading of DLLs into processes, potentially indicating process injection or exploitation.
Hijacked Process	Legitimate processes hijacked by malware for execution of malicious payloads.
Unexpected Child Process	Processes spawned unexpectedly, often a sign of command execution or exploitation.
C2 Communication	Evidence of communication with Command and Control (C2) servers for instructions or data exfiltration.
Data Exfiltration	Detection of sensitive data being extracted and transmitted to external systems.
Unusual Memory Access	Anomalies in memory usage, such as code injected into legitimate processes.
Malicious Payload in Memory	Detection of malicious code or payloads residing entirely in memory without writing to disk.
Registry Persistence Changes	Changes to Windows registry entries designed to maintain persistence or evade detection.
Malicious Registry Task	Registry-based tasks configured to execute malicious actions.
PowerShell Activity	Abnormal or obfuscated PowerShell scripts executing commands.
WMI Activity	Suspicious use of Windows Management Instrumentation (WMI) for remote execution or reconnaissance.
JavaScript Activity	Malicious JavaScript executing unauthorized actions in the browser or system.
Unusual Logon Patterns	Login attempts or successful logins from unexpected locations or times.
Privilege Escalation	Actions indicative of attempts to gain higher privileges within a system.
Scheduled Task Changes	Creation or modification of scheduled tasks to maintain persistence.
Behavioral Anomalies	General deviations from normal behavior patterns within the system or network.
Unusual Network Connections	Connections to known malicious IP addresses or domains, or unusual data transfer volumes.

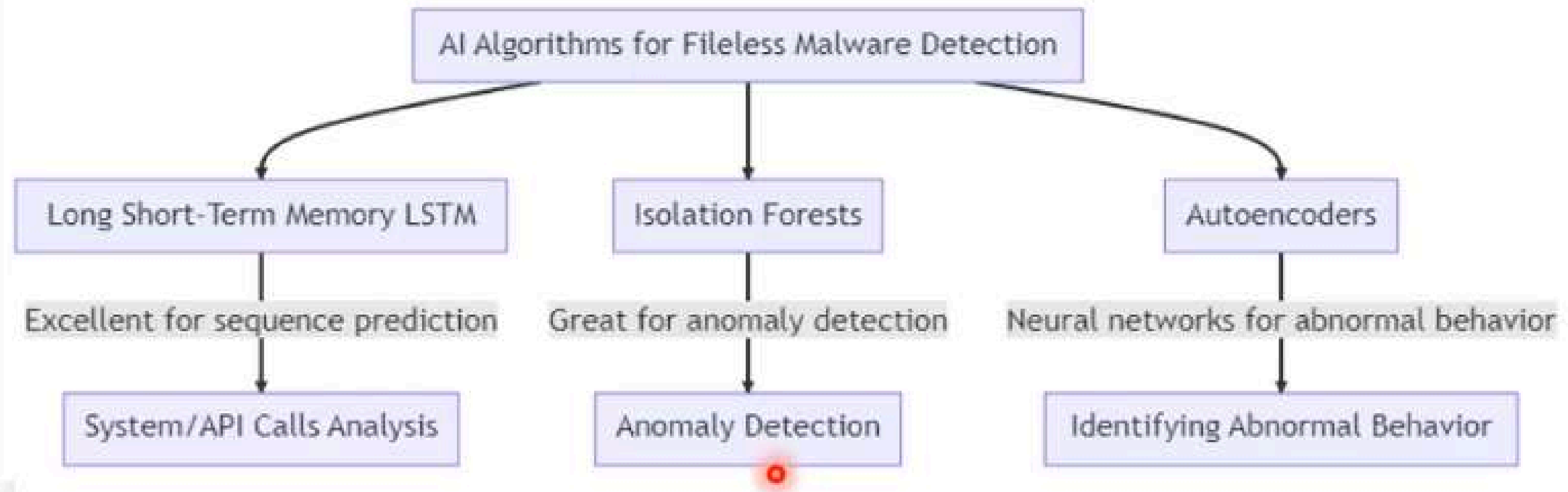


# Dataset Essentials: Unpacking Fileless Malware Features

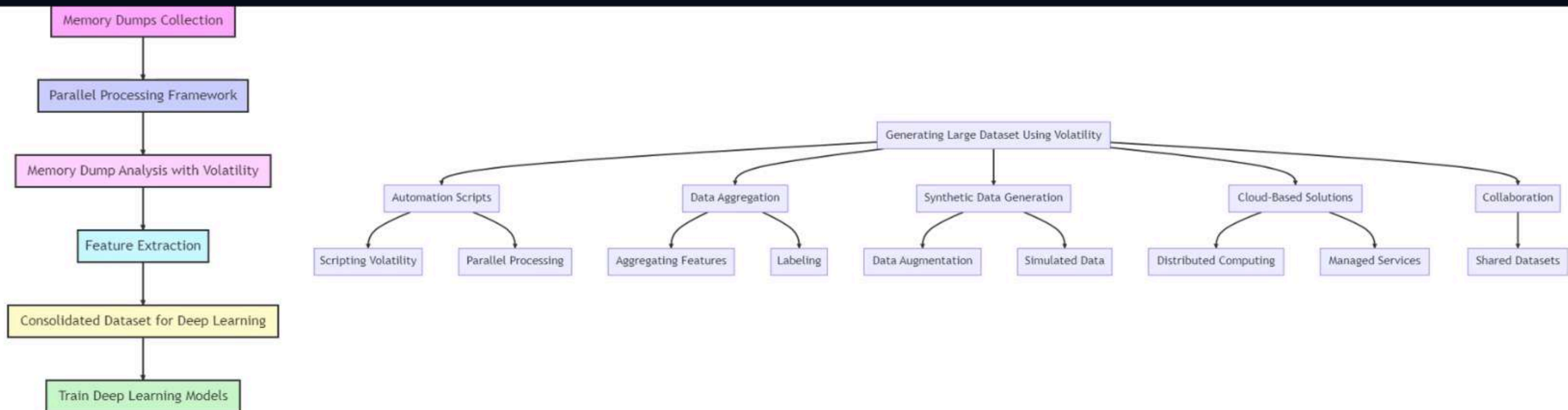




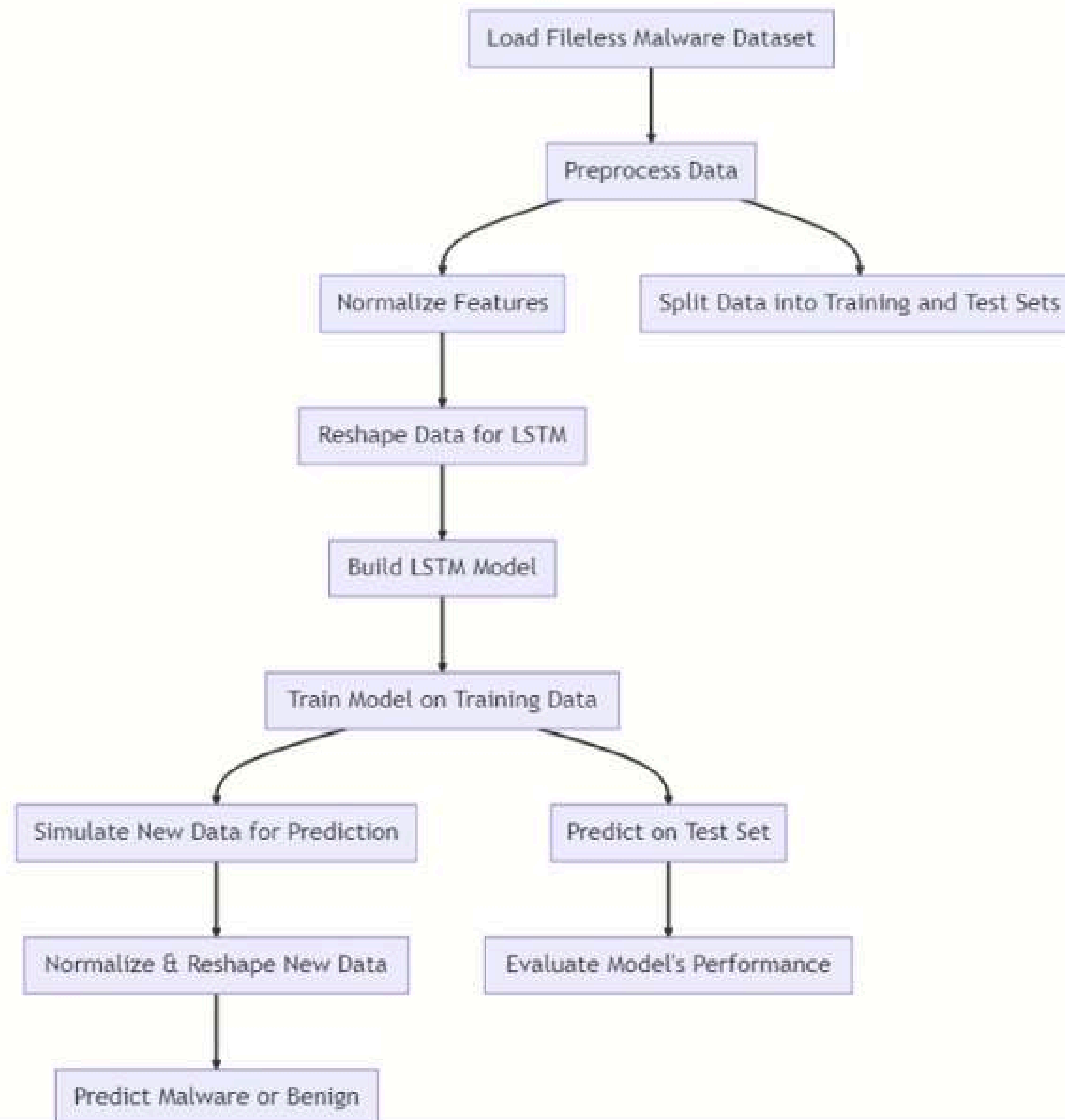






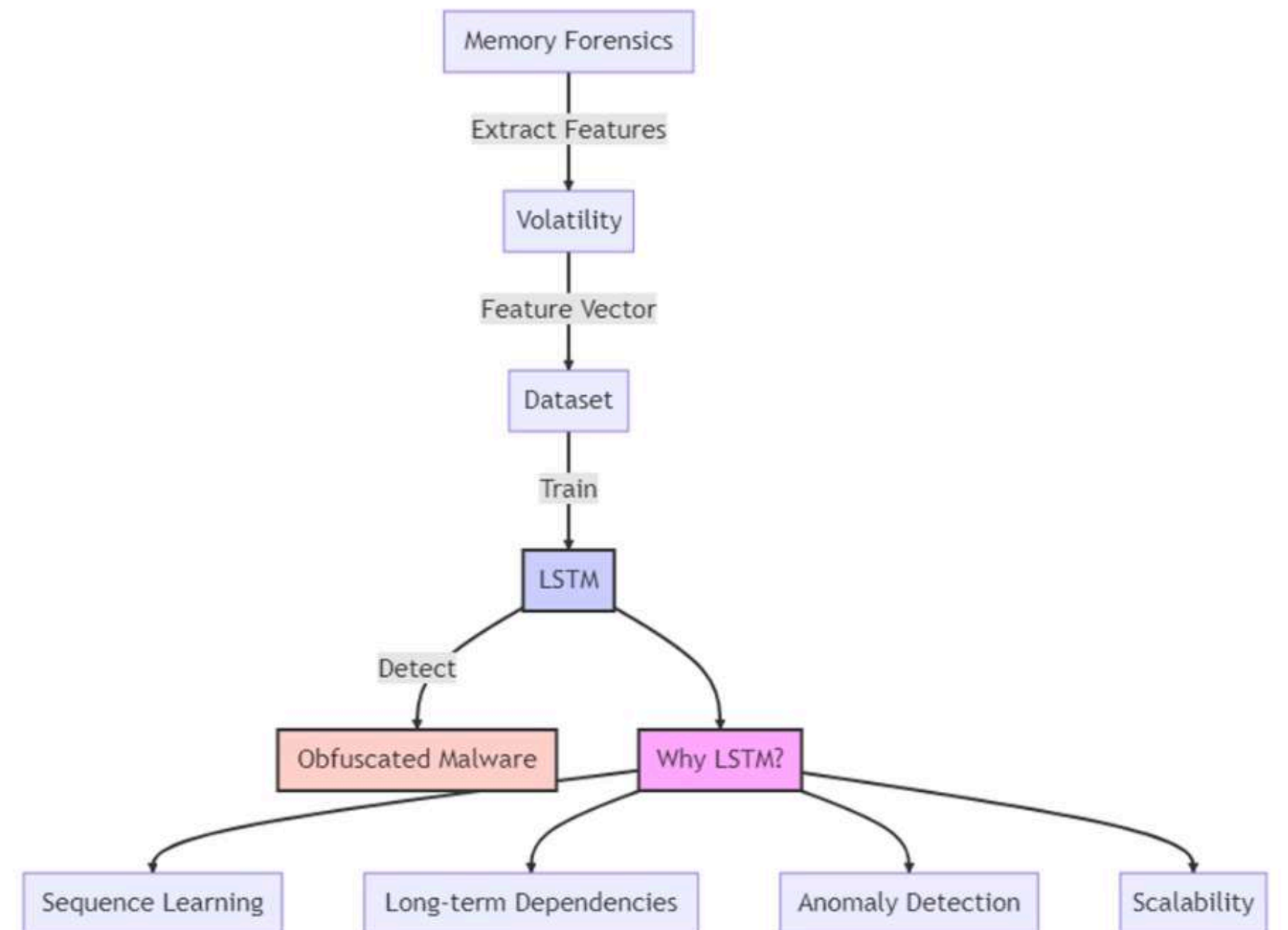
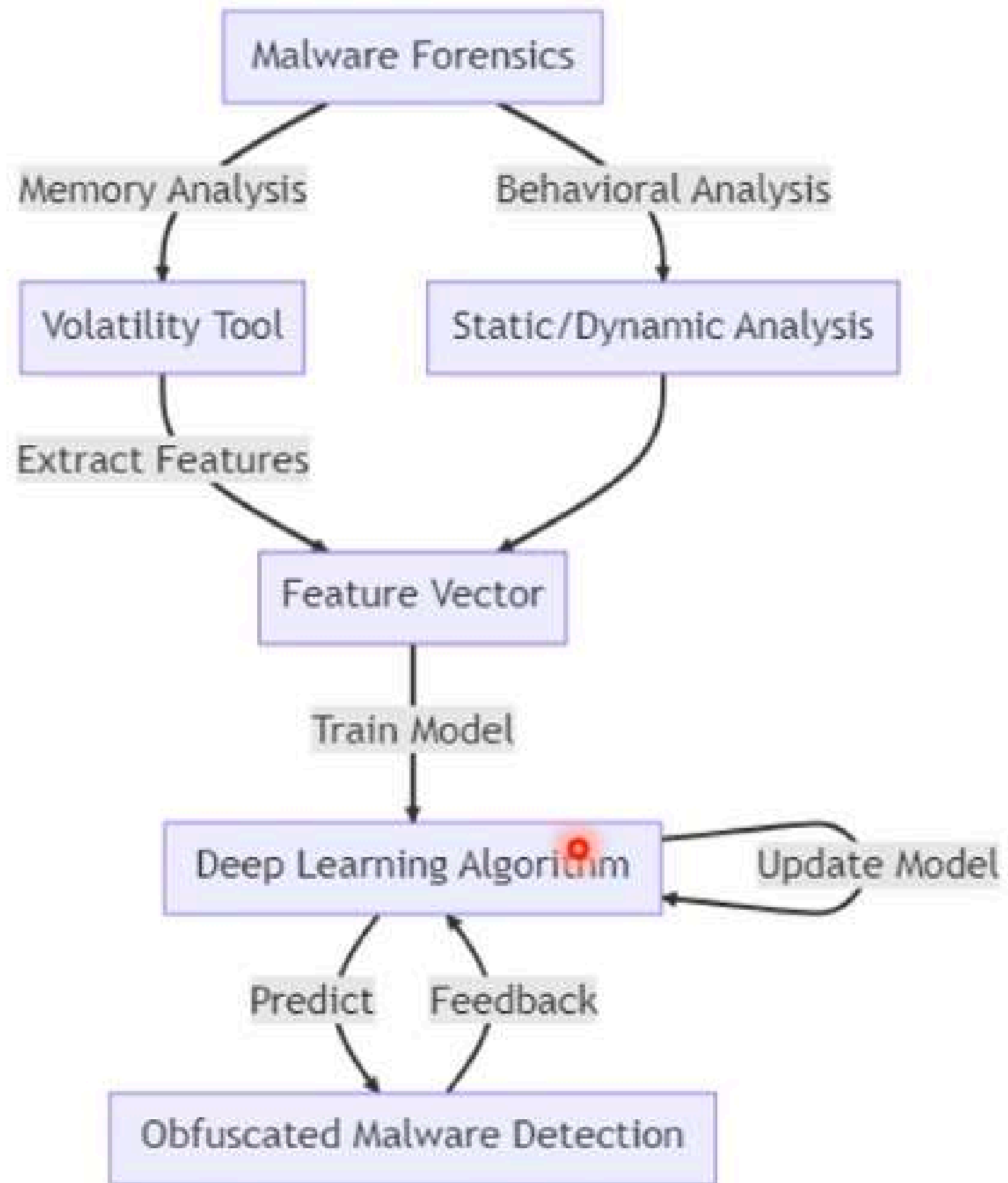


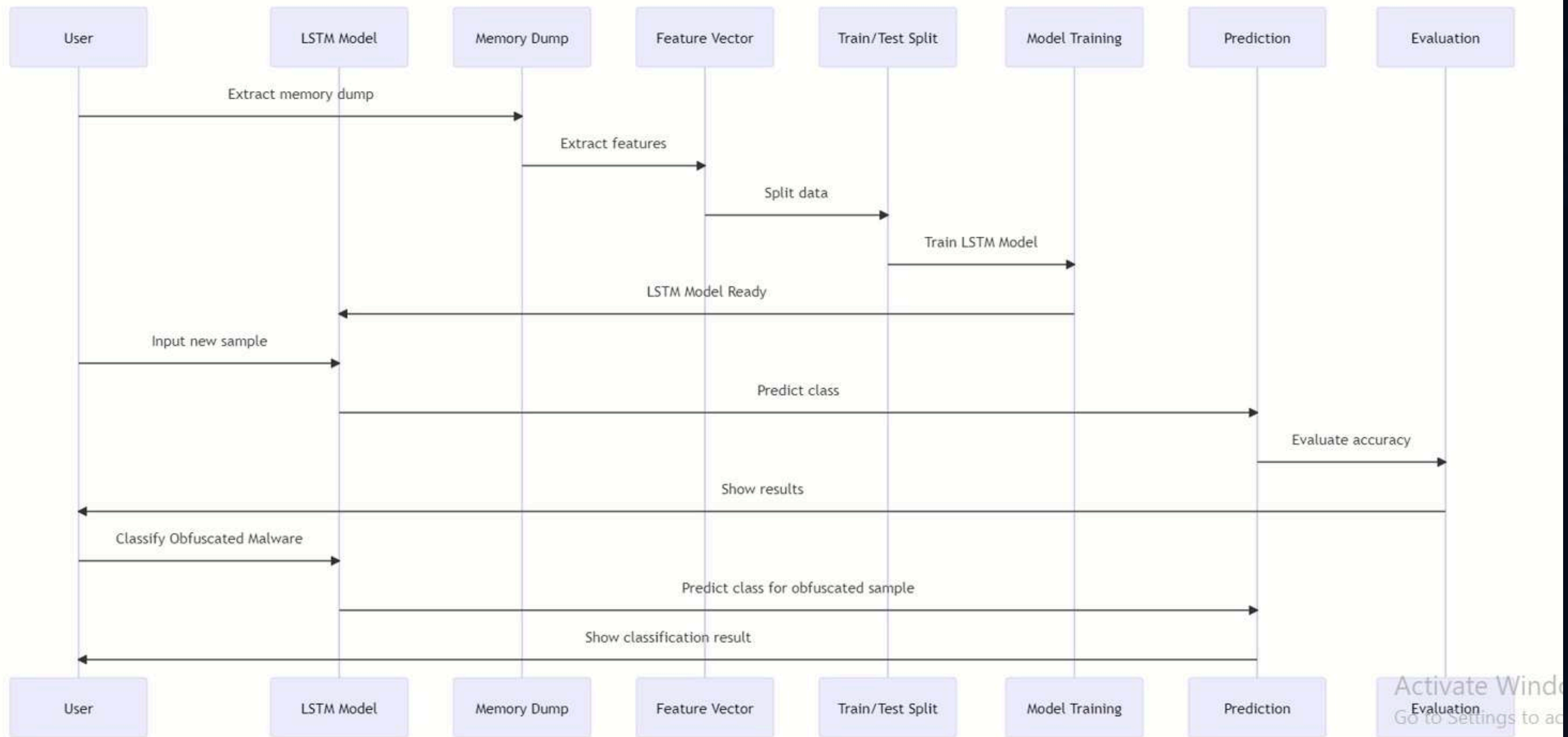




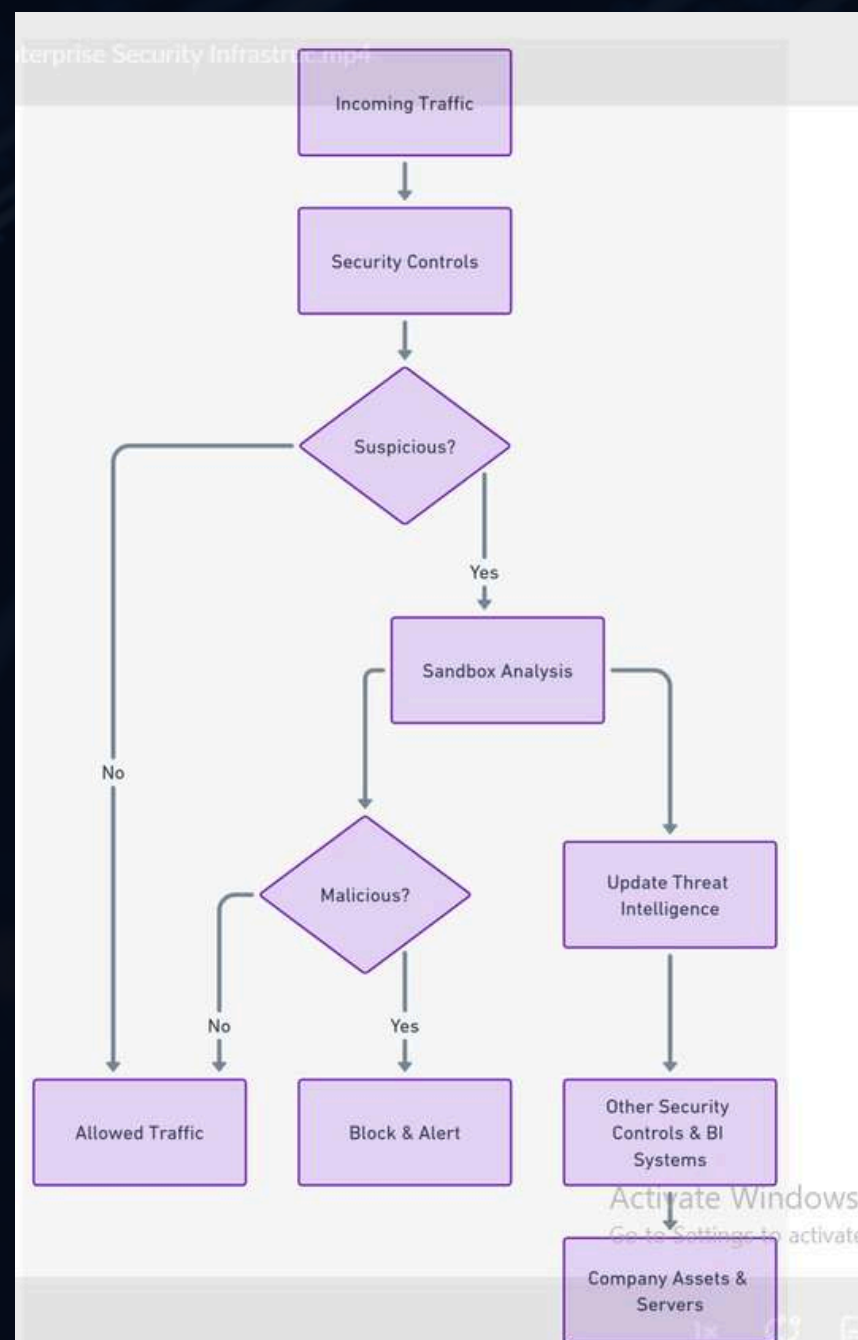
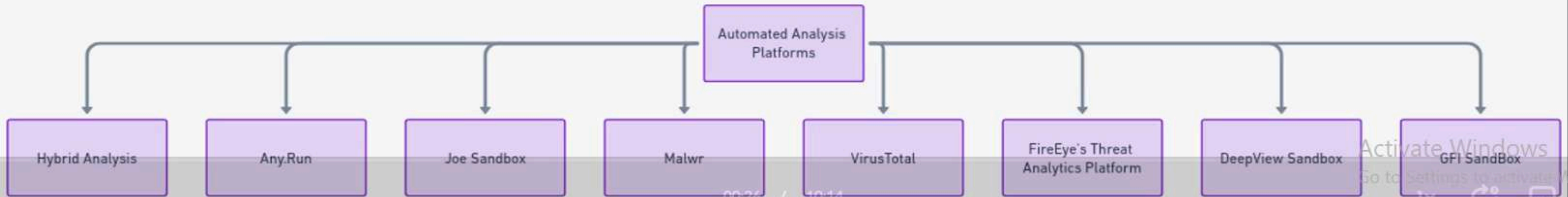
```
1 specific_calls,access_resources,suspicious_api_calls,scripting_engine_calls,malicious_dlls,dlls_loaded_into_processes,hijacked_processes,unexp
  ected_child_processes,c2_communications,data_exfiltration,unusual_memory_access,malicious_payloads_in_memory,registry_persistence_changes,mali
  cious_registry_tasks,powershell_activity,wmi_activity,javascript_activity,unusual_logon_patterns,privilege_escalation,scheduled_tasks_changes,
  behavioral_anomalies,is_malware
2 5,2,10,3,4,2,1,6,7,50,8,3,4,2,15,5,3,4,2,3,5,1
3 2,5,3,6,1,4,6,2,2,30,3,6,2,5,4,8,6,3,5,2,4,0
4 8,6,2,3,2,6,3,8,3,70,9,3,6,3,6,9,8,3,3,6,3,1
5 3,3,8,8,6,3,8,3,8,20,4,8,3,8,9,3,3,8,8,3,8,0
6 7,4,7,5,3,5,2,7,6,60,7,2,5,3,14,6,2,5,3,4,6,1
7 3,6,4,7,2,5,7,3,3,40,4,7,3,6,5,9,7,4,6,3,5,0
8 9,7,3,4,3,7,4,9,4,80,10,4,7,4,7,10,9,4,4,7,1,1
9 4,4,9,9,7,4,9,4,9,30,5,9,4,9,10,4,4,9,9,4,9,0
10 6,3,6,4,2,4,1,5,5,45,6,1,3,1,12,4,1,3,1,2,4,1
11 1,4,1,5,1,3,5,1,1,15,1,5,1,4,3,6,5,1,4,1,3,0
12 10,8,5,6,5,8,6,10,6,90,11,6,8,6,8,11,10,6,6,8,6,1
13 5,5,10,10,8,5,10,5,10,40,6,10,5,10,11,5,5,10,10,5,10,0
14 4,2,4,3,1,2,1,4,4,35,4,1,2,1,11,3,1,2,1,1,3,1
15 2,3,2,4,1,1,4,2,2,25,2,4,2,3,2,5,4,2,3,2,2,0
16 7,5,3,2,2,5,2,6,3,55,6,2,4,2,5,6,6,2,2,5,2,1
17 6,6,7,7,4,6,7,6,7,65,7,7,6,7,8,7,7,7,7,6,7,0
18 5,7,6,8,5,8,8,7,8,75,8,8,7,8,9,8,8,8,8,7,8,1
19 8,8,9,9,6,9,9,8,9,85,9,9,8,9,10,9,9,9,9,8,9,0
20 9,9,8,10,7,10,10,9,10,95,10,10,9,10,11,10,10,10,10,9,10,1
21 10,10,10,11,8,11,11,10,11,100,11,11,10,11,12,11,11,11,11,10,11,0
22
```

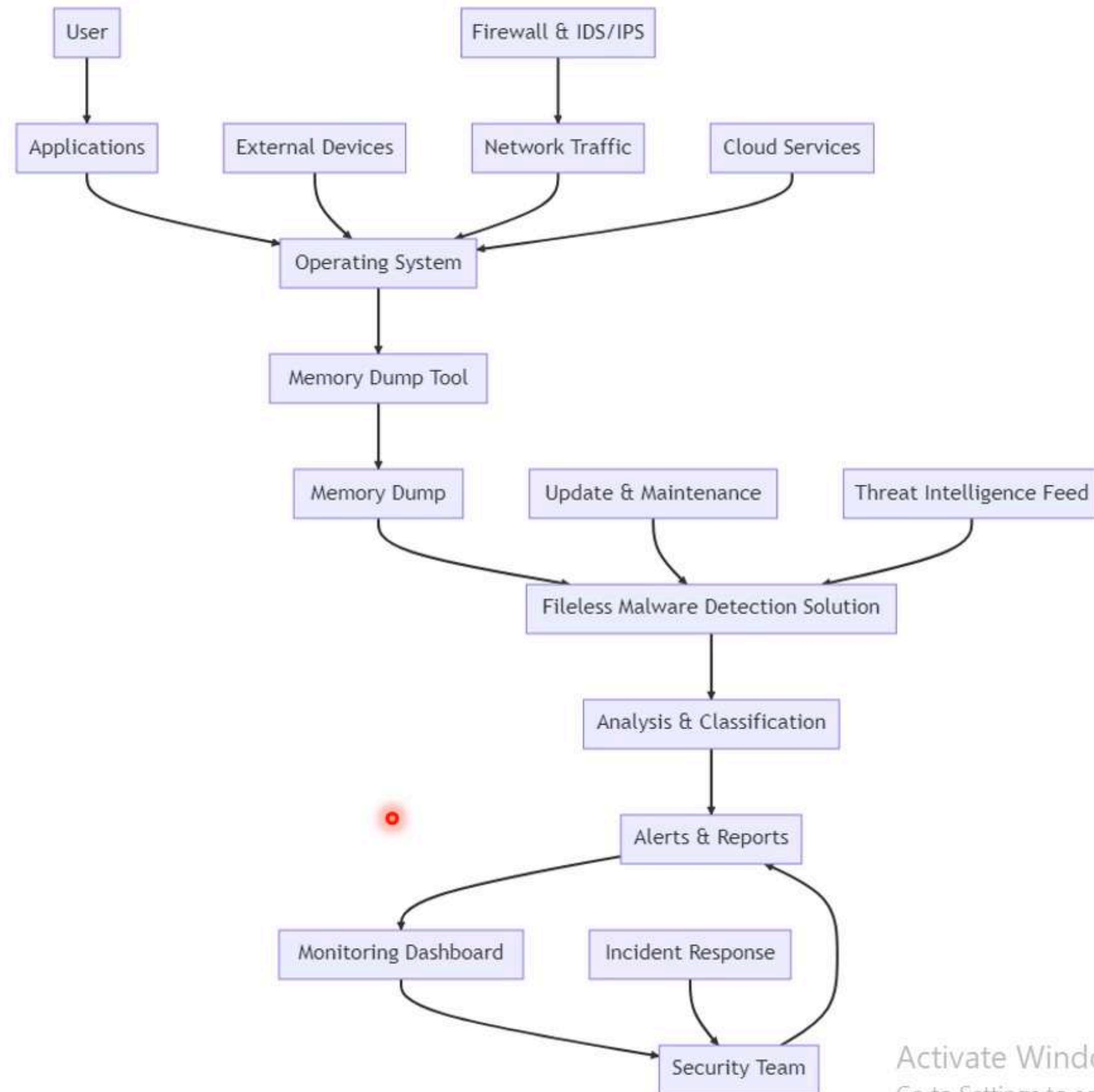




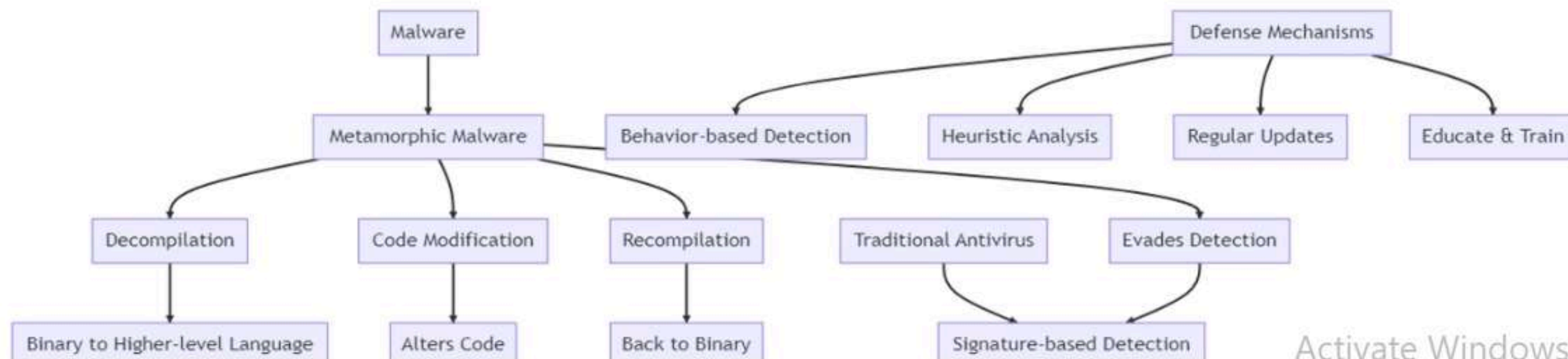
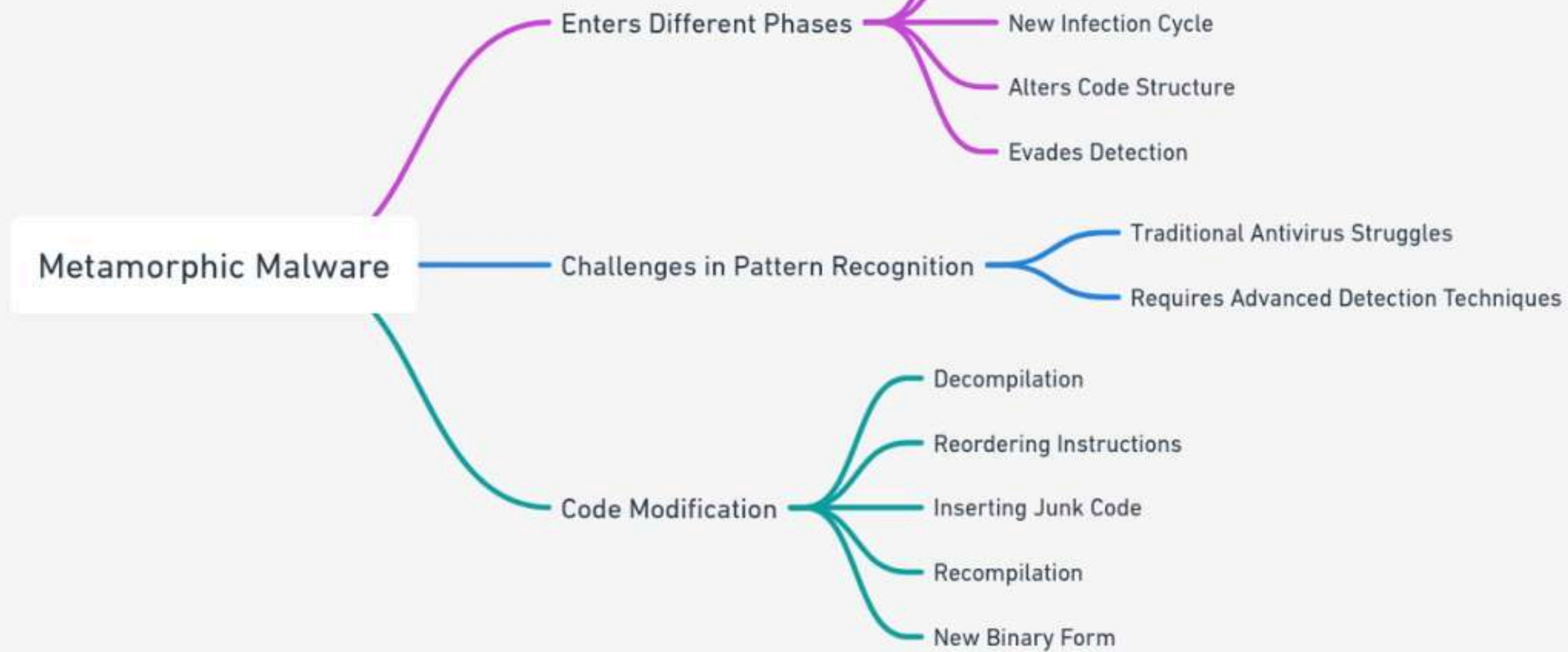


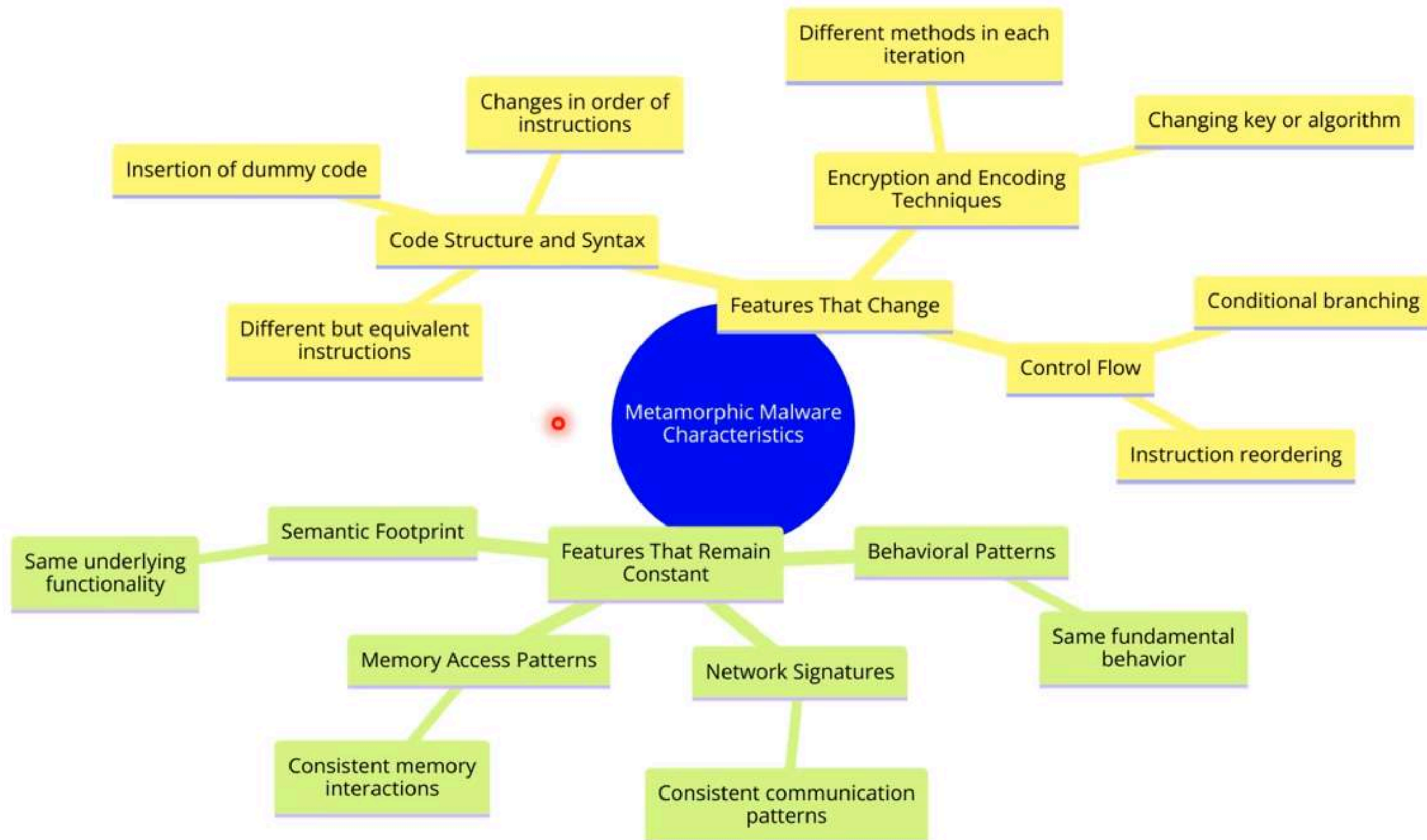














# Advanced Techniques for Metamorphic Malware Detection

## Deep Reinforcement Learning (DRL)

- Application: Dynamic Analysis in Sandboxed Environment
- Advantages
  - Adaptive Analysis
  - Continuous Learning
- Challenges
  - Training Complexity
  - Exploration vs. Exploitation

## Sequence-to-Sequence Models

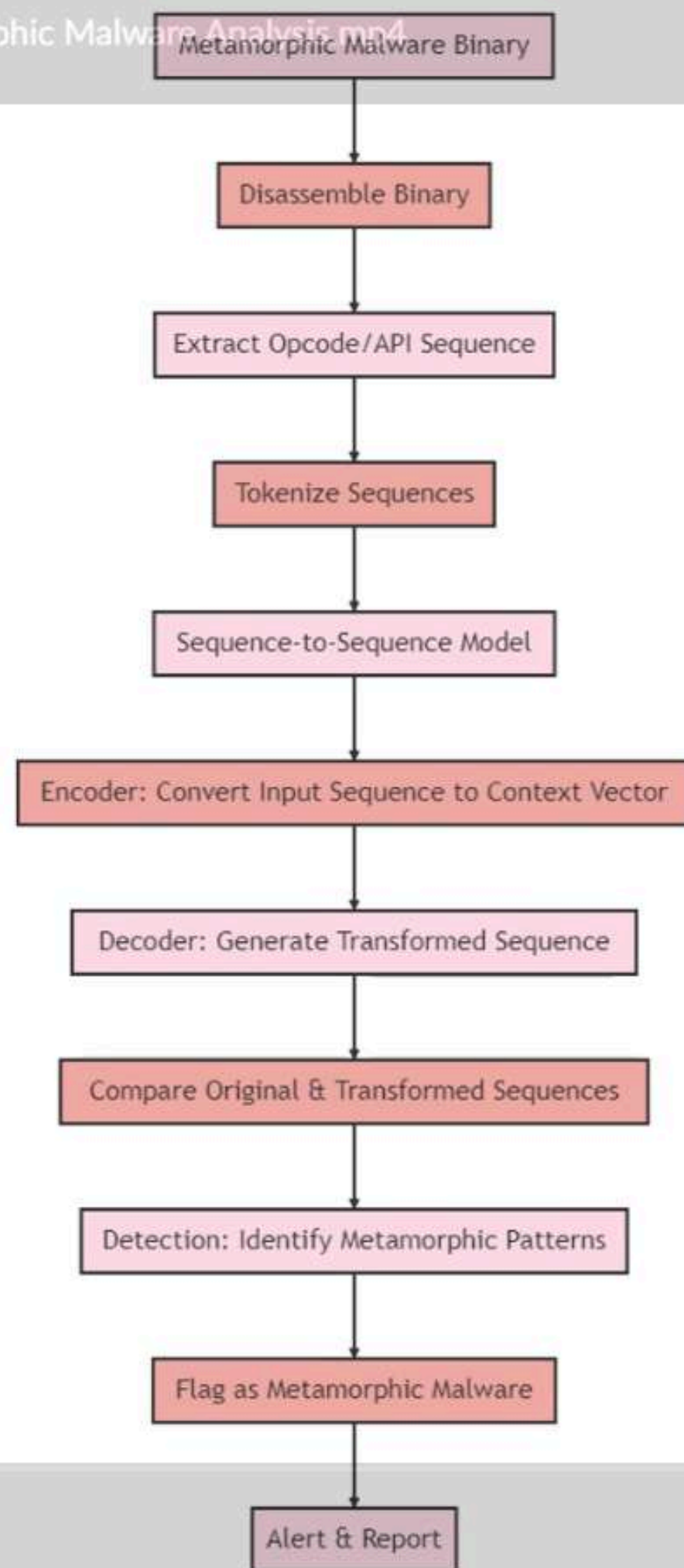
- Application: Transforming Opcode Sequences
- Advantages
  - Pattern Recognition
  - Handling Variability

## Graph Neural Networks (GNNs)

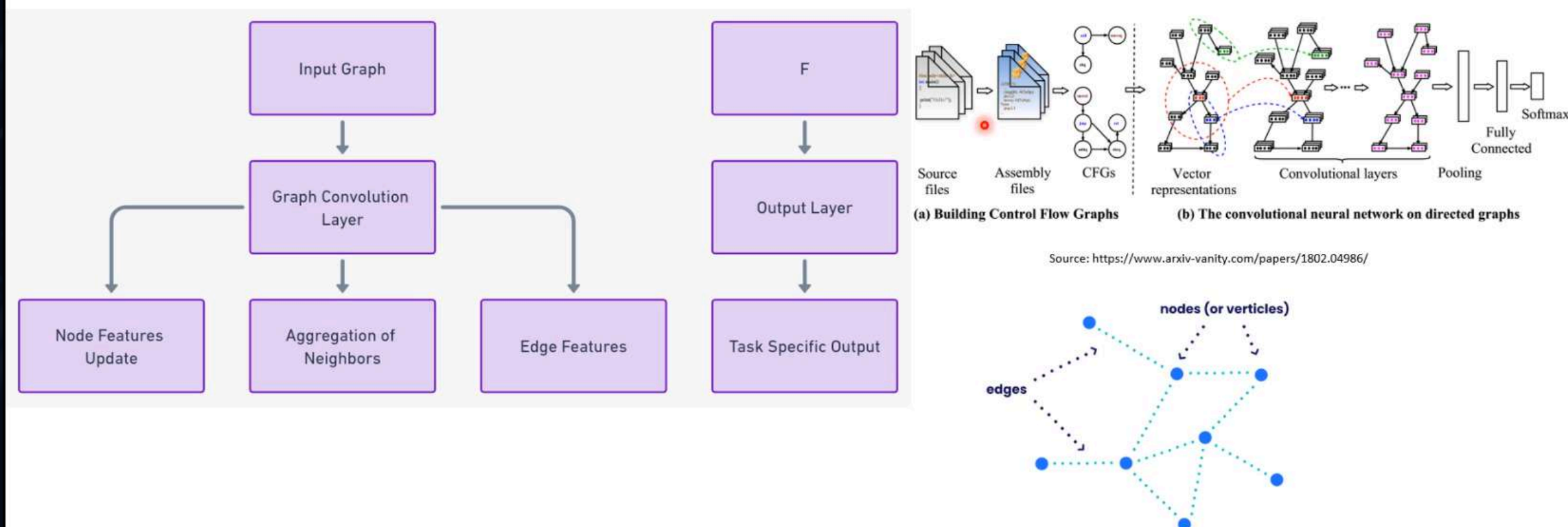
- Application: Analyzing Malware's Control/Data Flow Structures
- Advantages
  - Structural Pattern Recognition
  - Relational Learning

## Comparison & Conclusion

- Performance: Dynamic vs. Static Analysis
- Applicability: Interaction vs. Pattern Recognition
- Complexity: Resource-Intensive vs. Straightforward

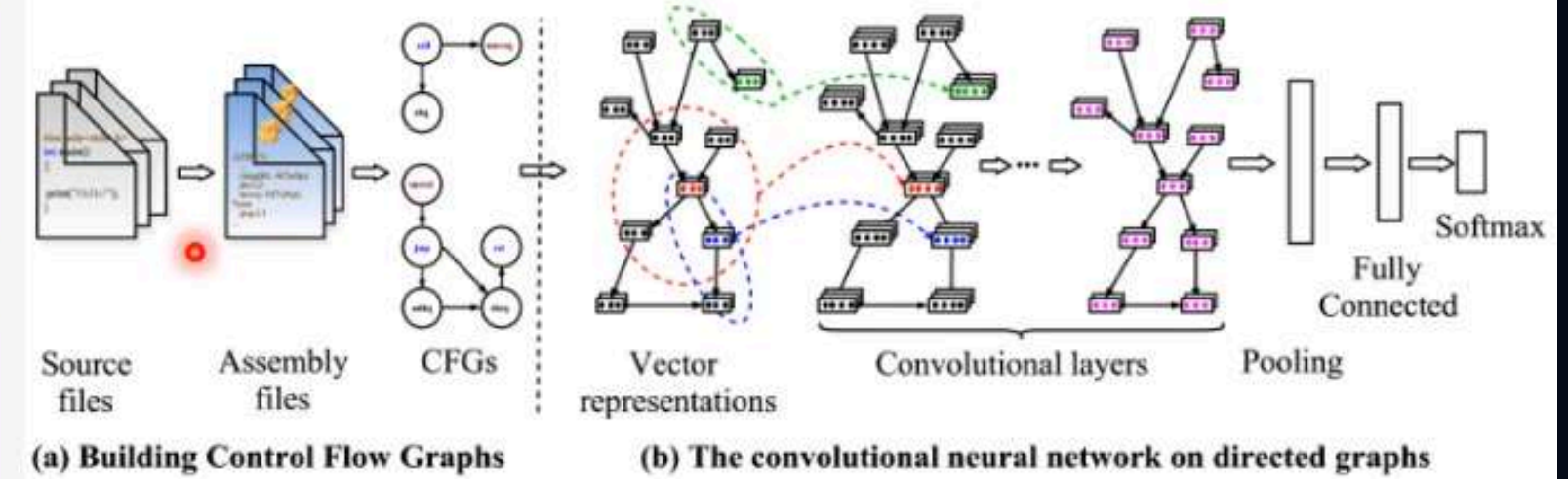
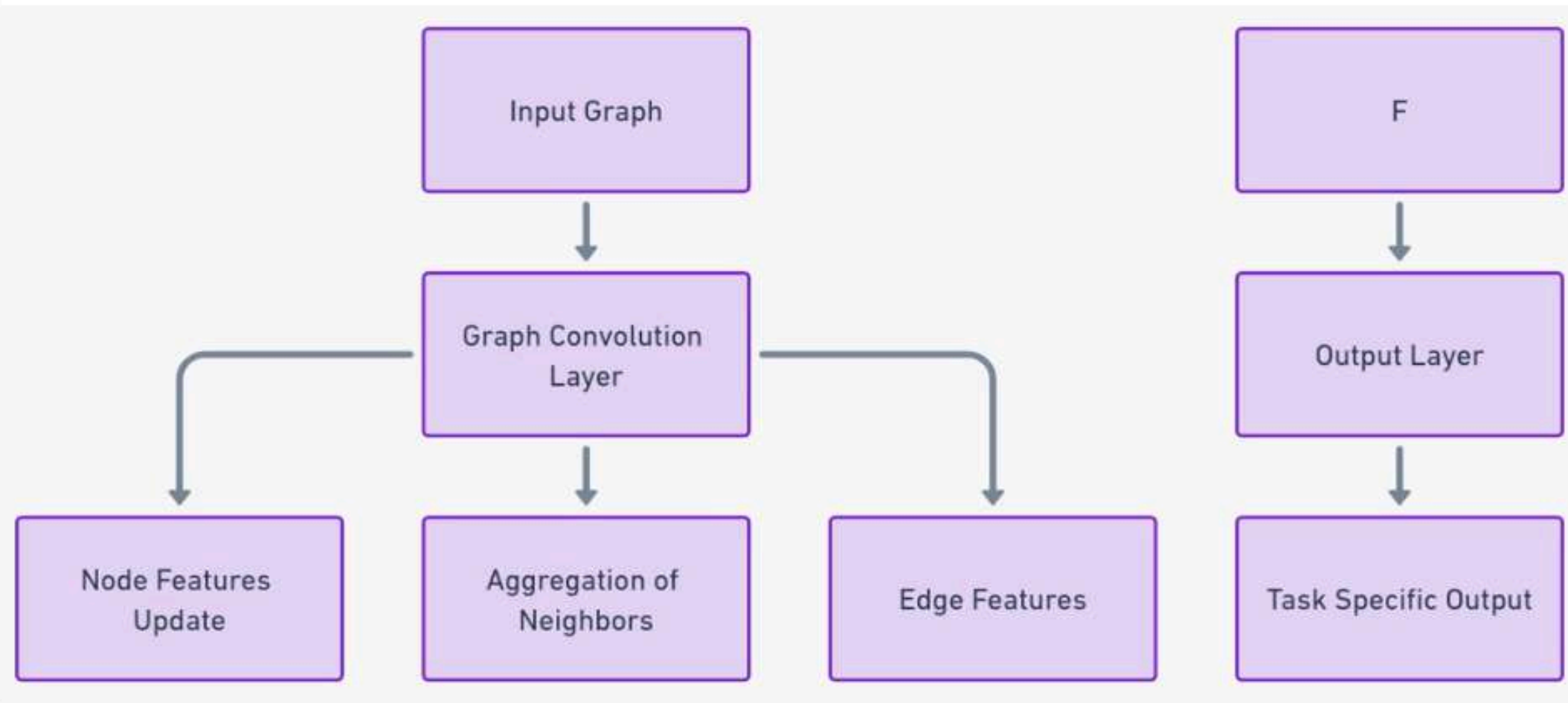


## CFG & GNN

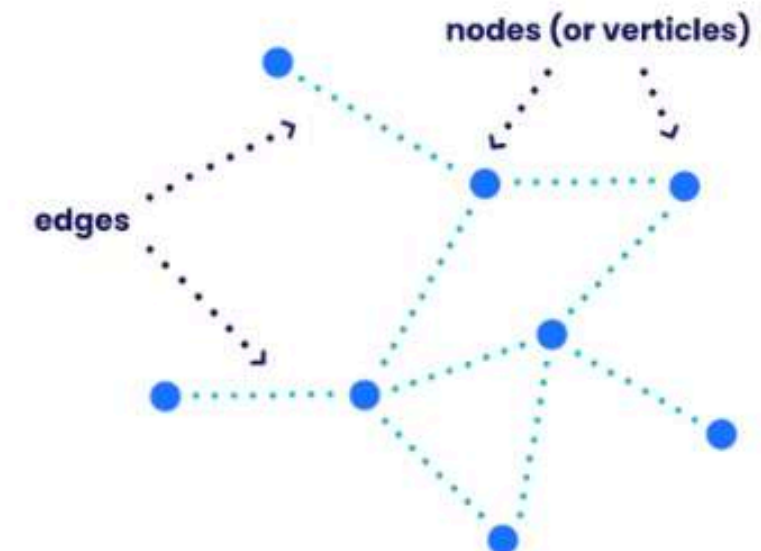




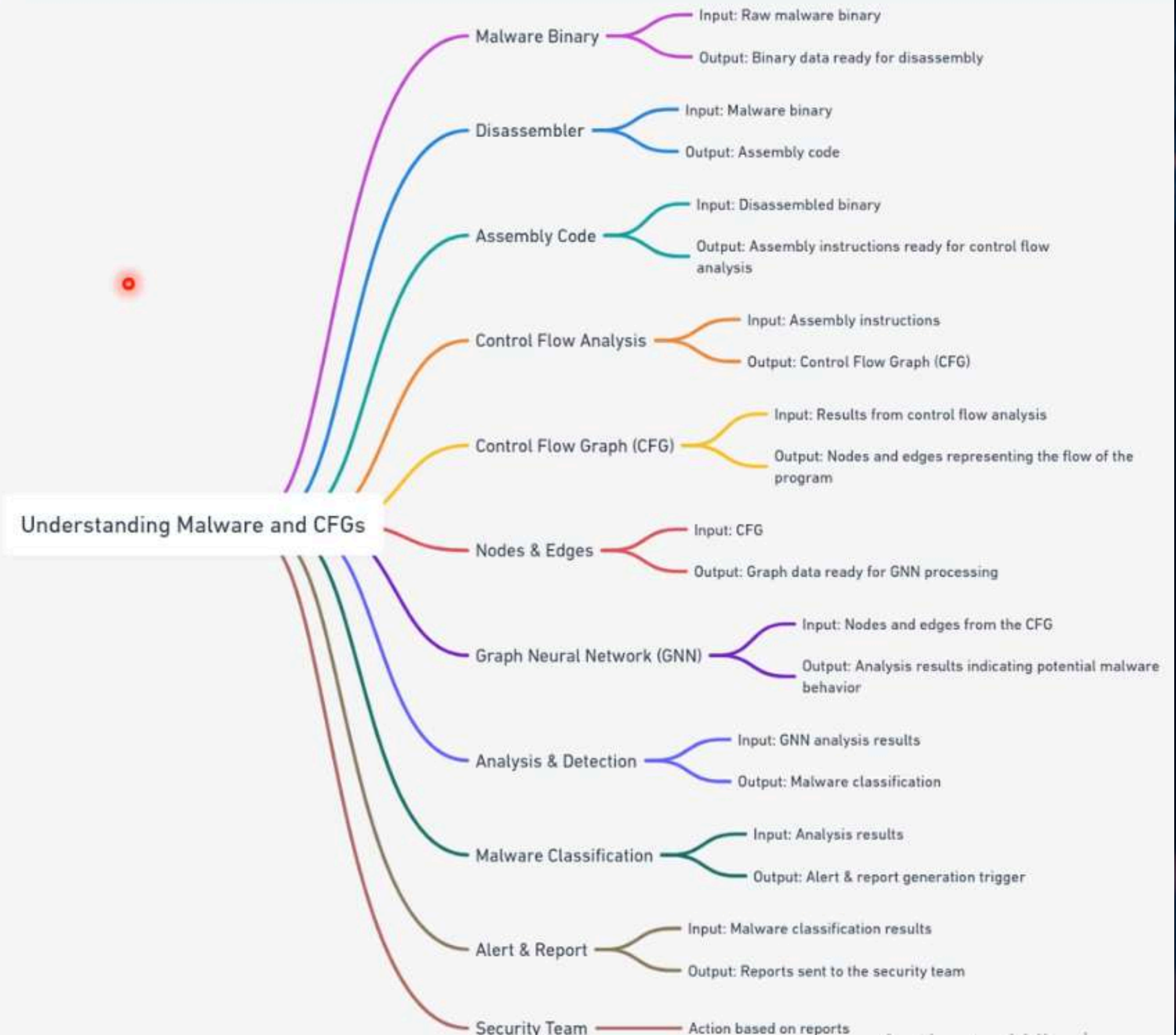
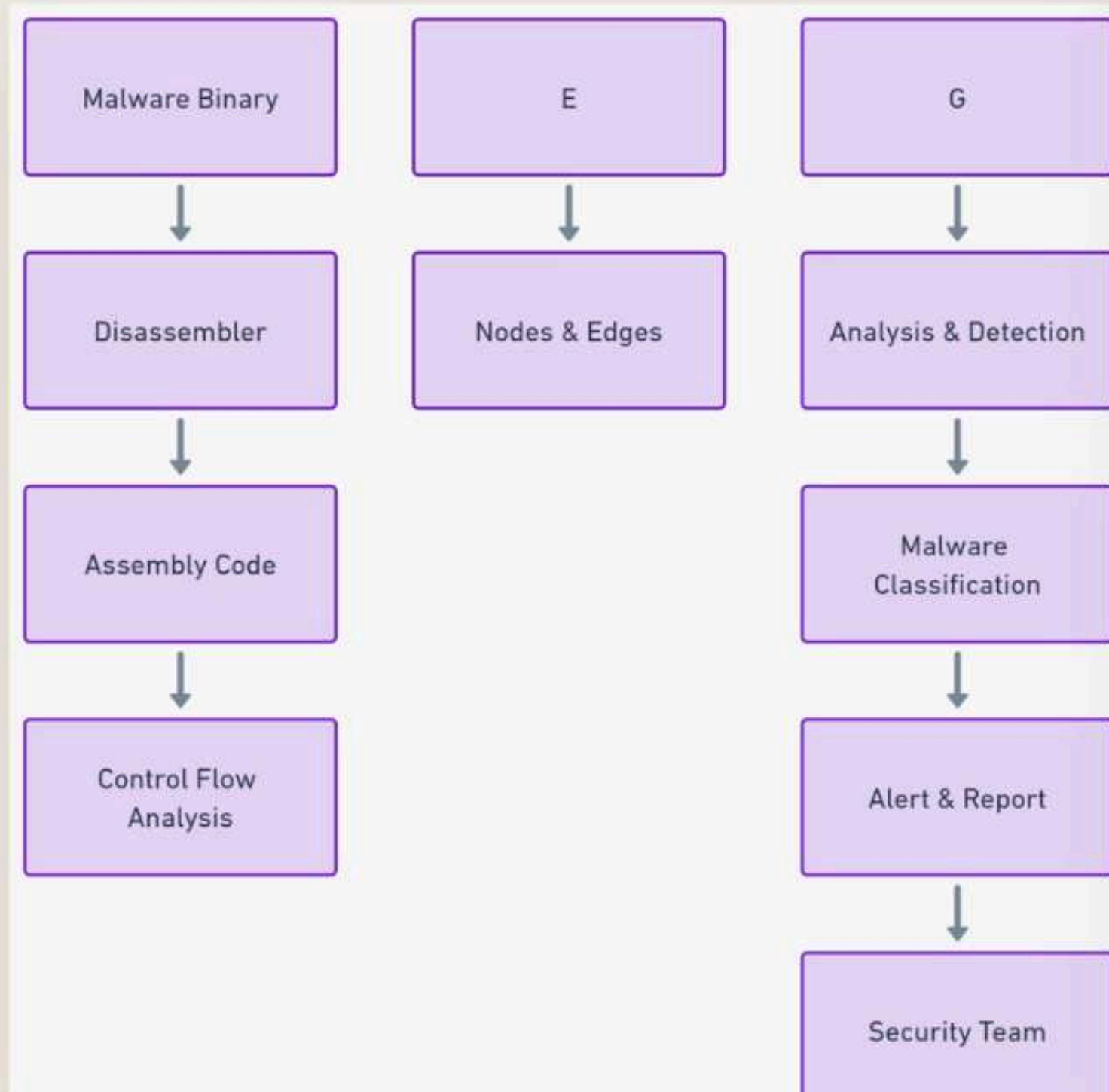
# CFG & GNN



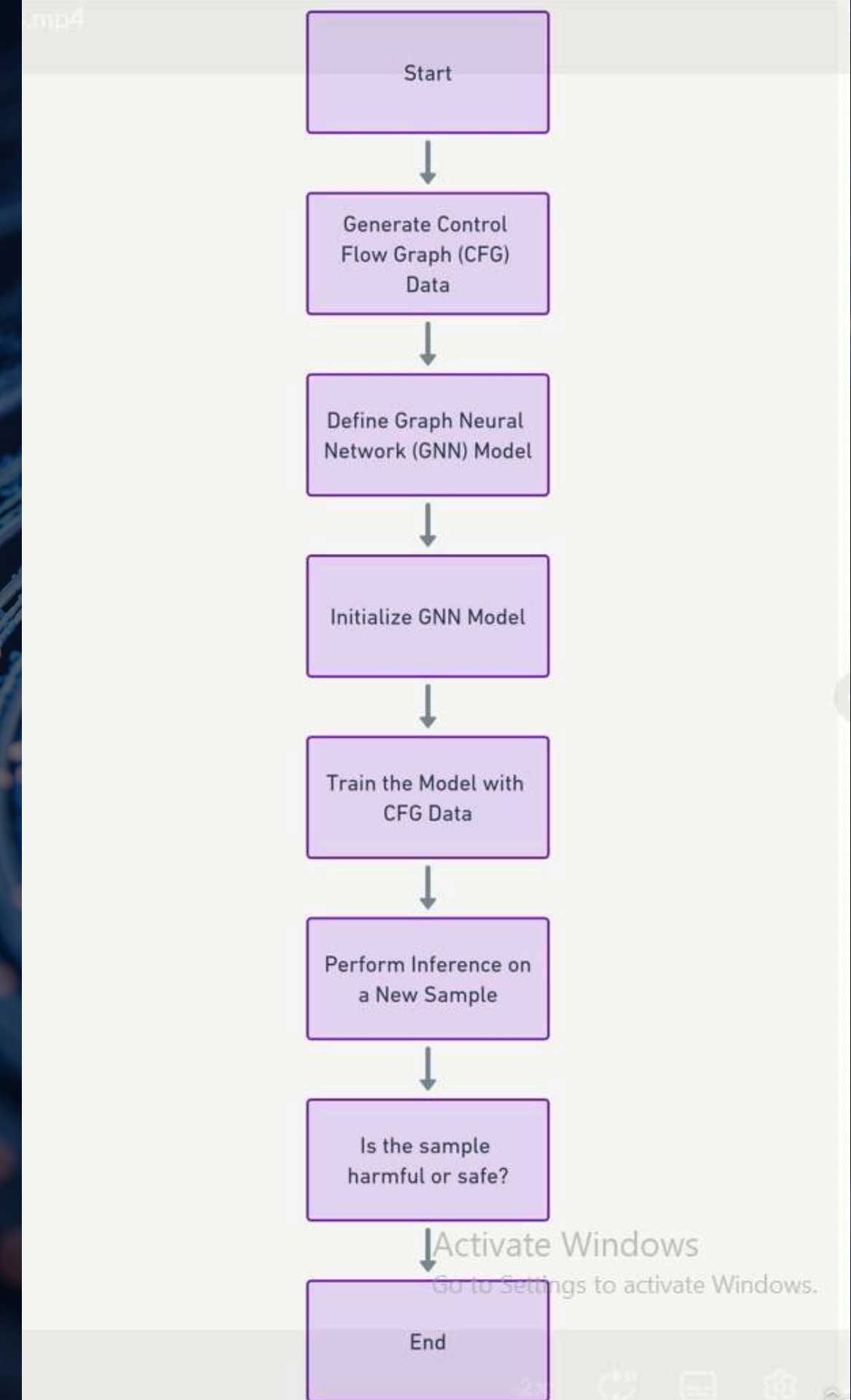
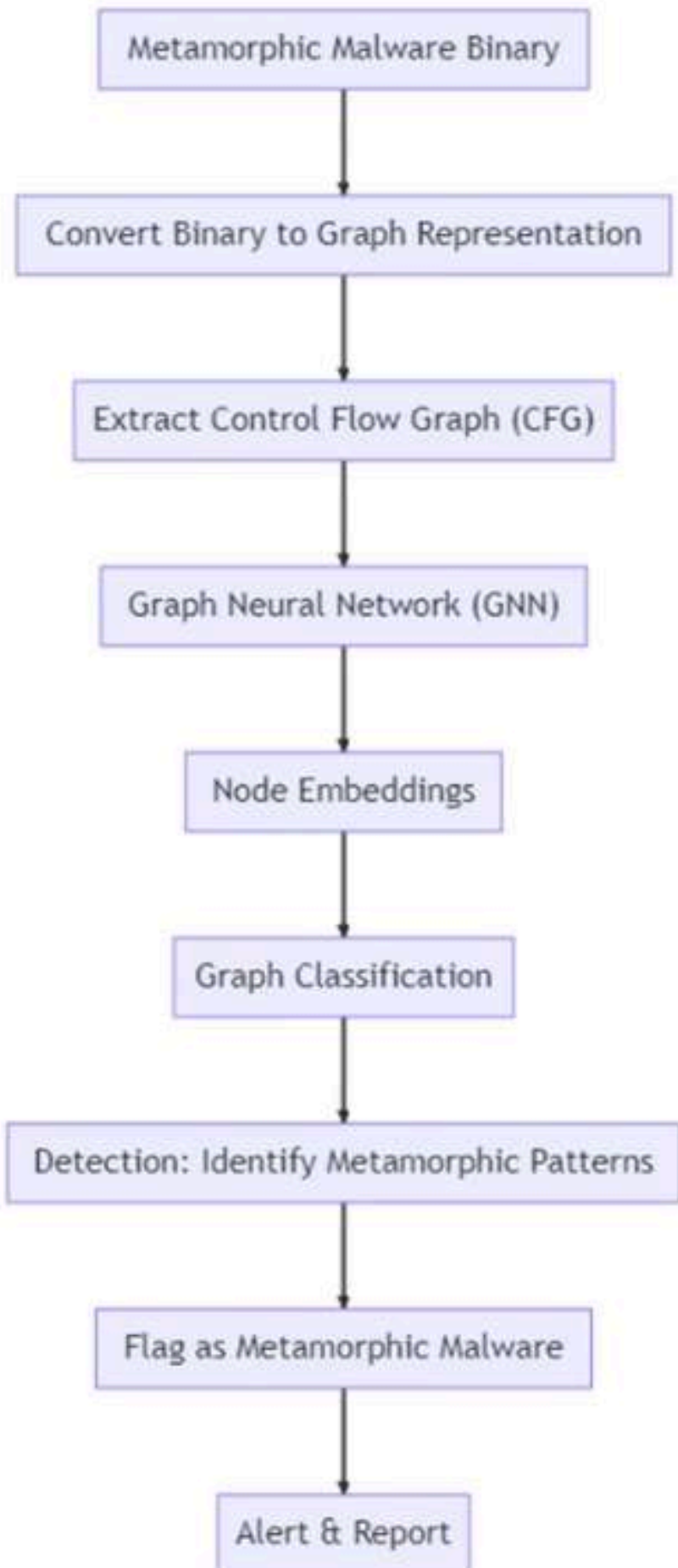
Source: <https://www.arxiv-vanity.com/papers/1802.04986/>



# Deep Dive into Malware Analysis using Control Flow Graphs (CFGs) and Graph Neural Networks (GNNs)



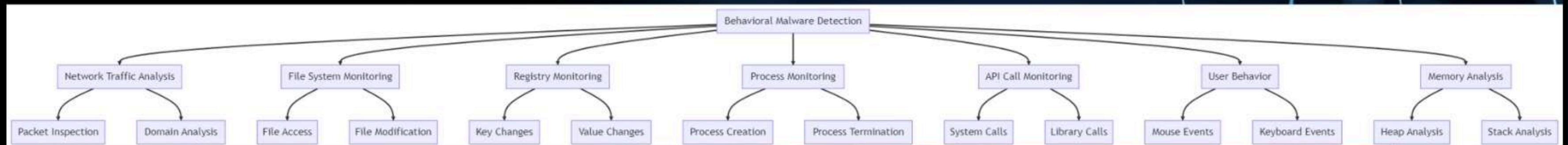
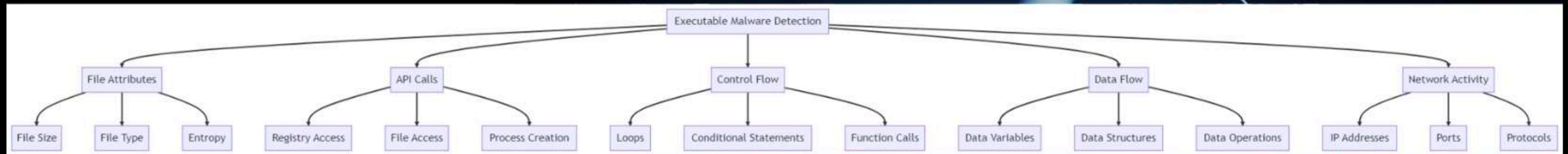








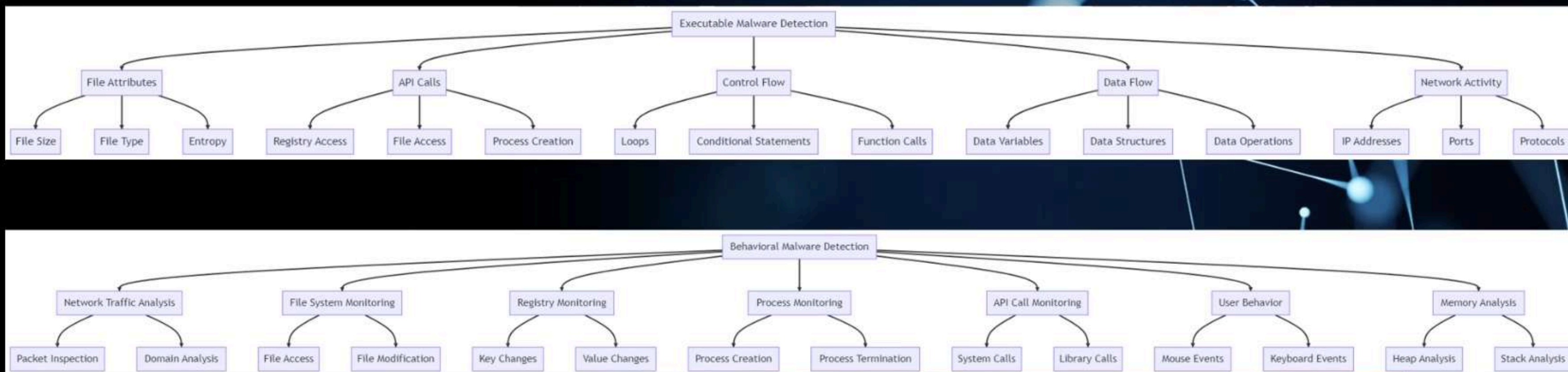
# Behavioral and Executable Malware: A Unified AI Detection Strategy



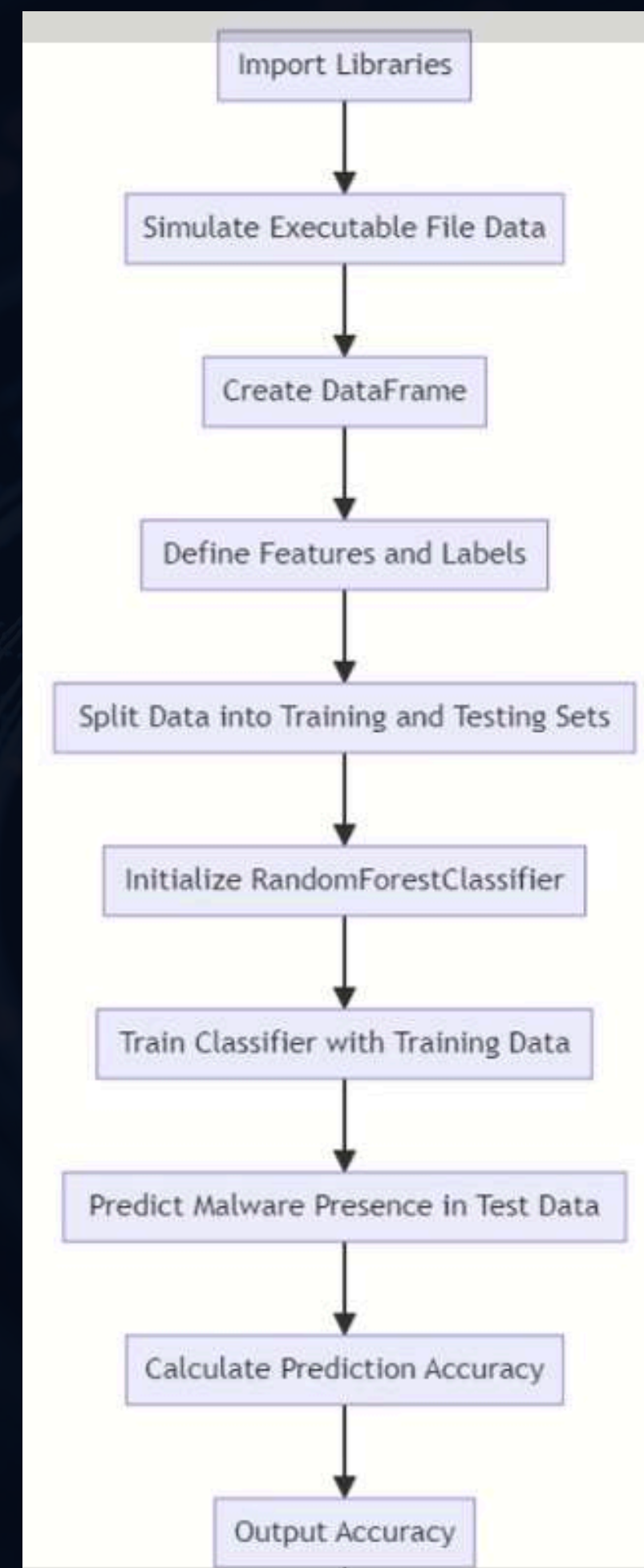




# Behavioral and Executable Malware: A Unified AI Detection Strategy



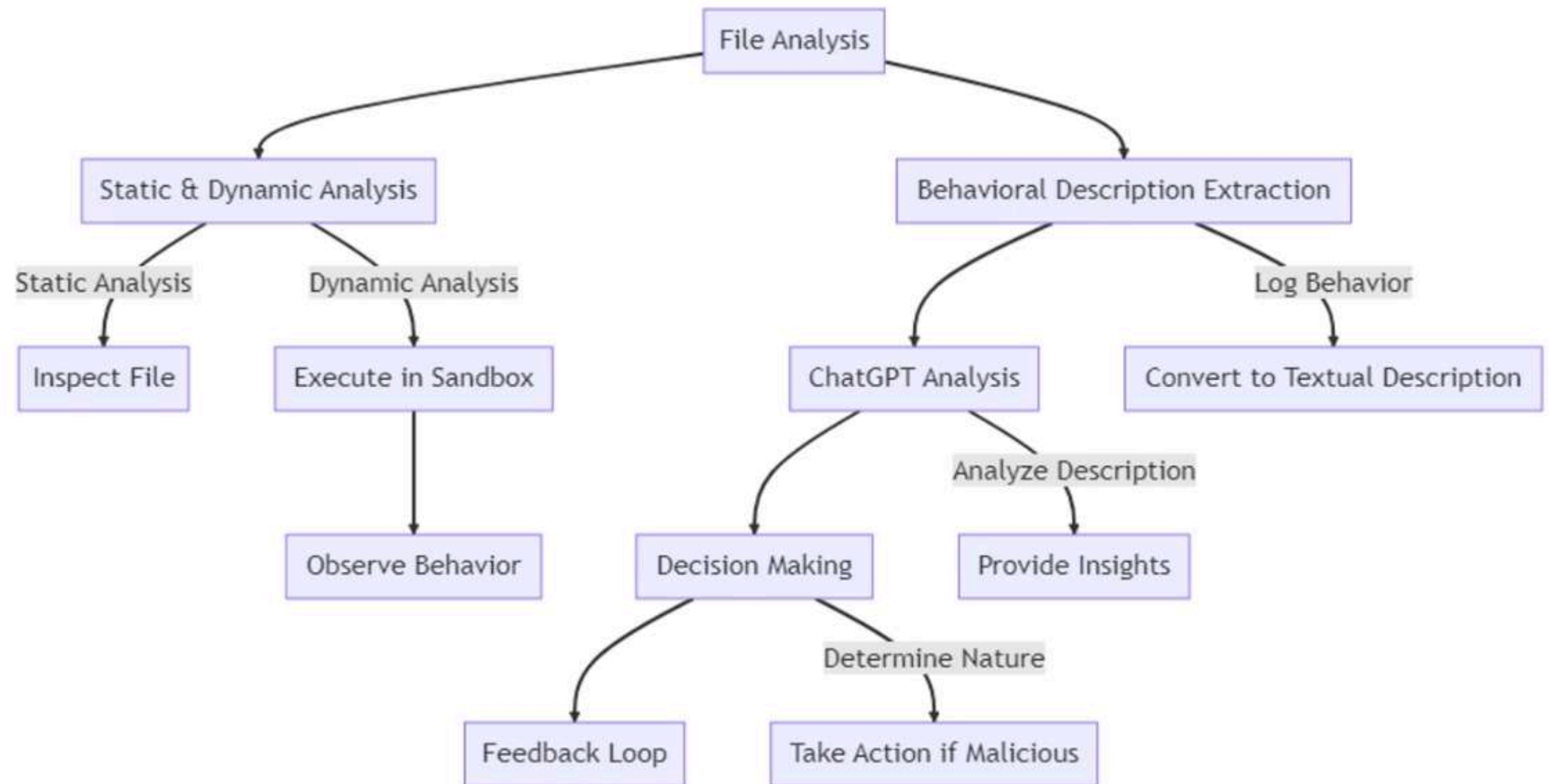


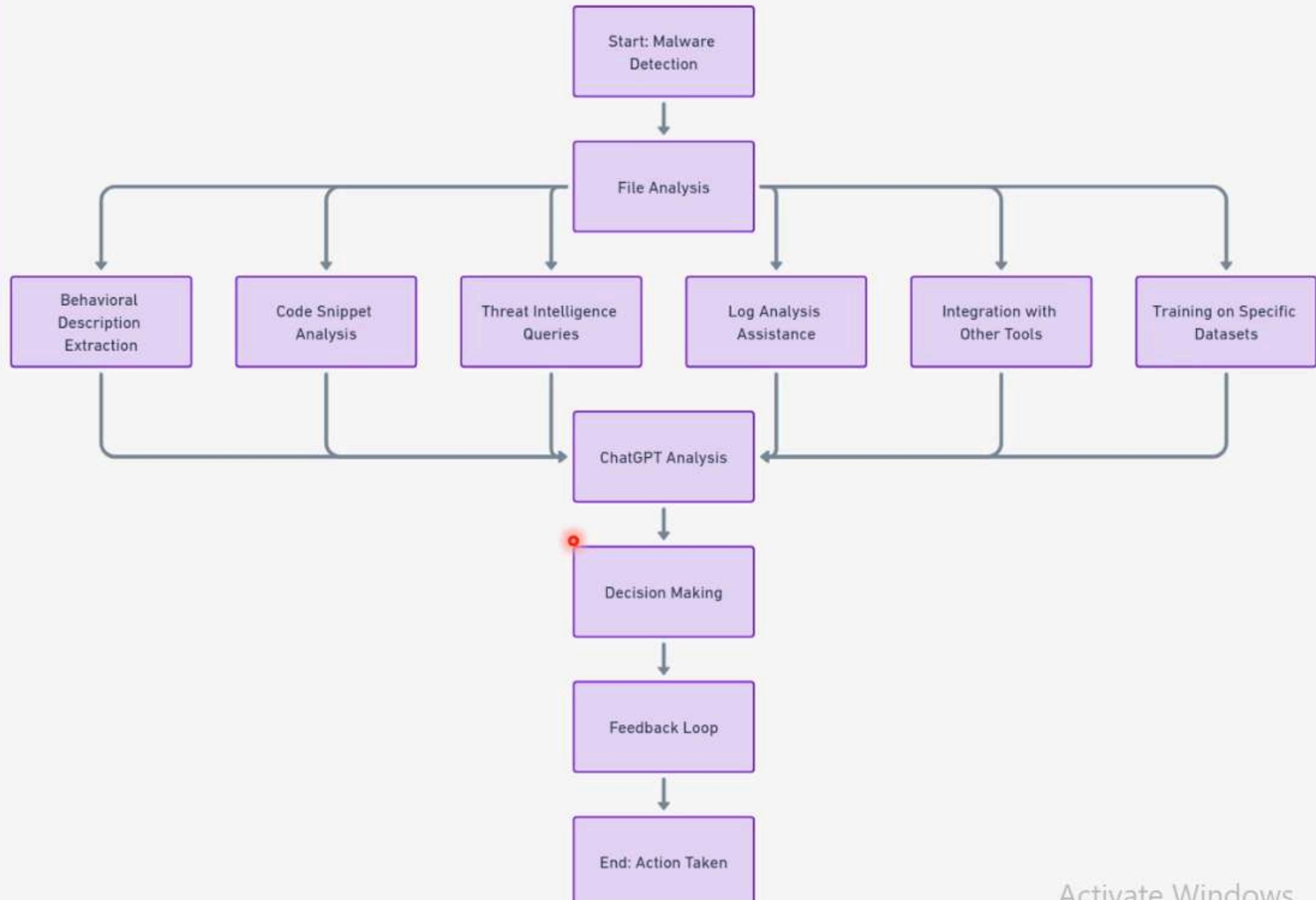




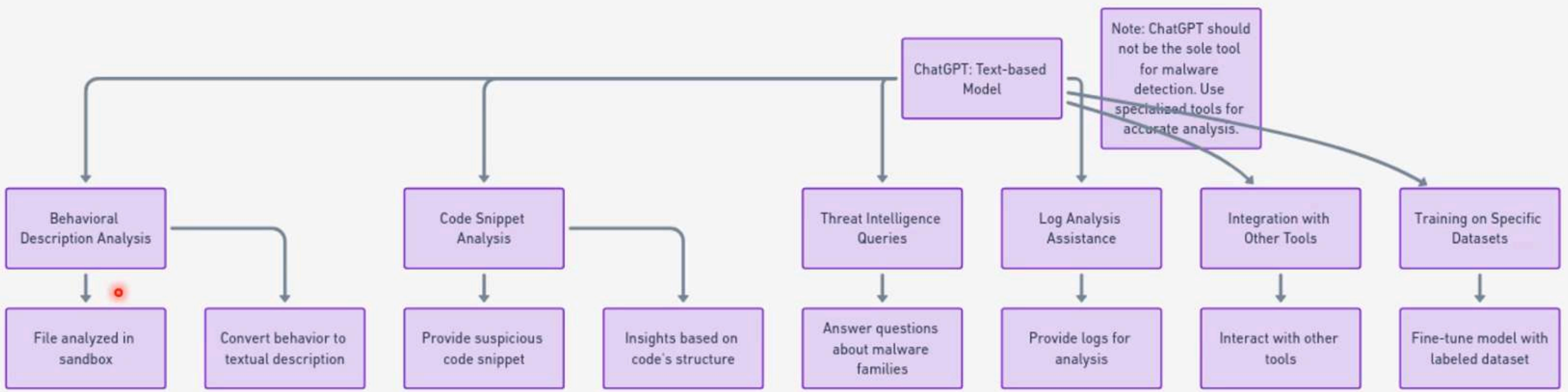


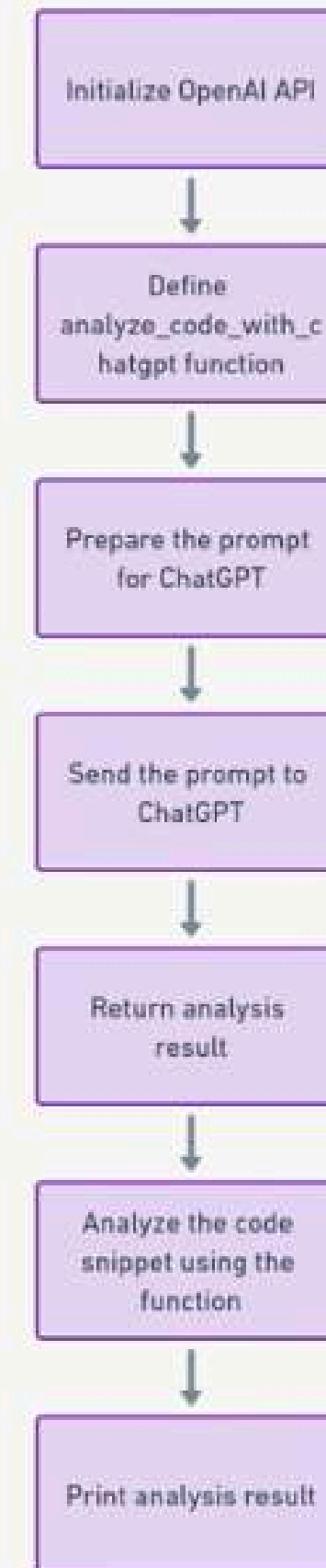
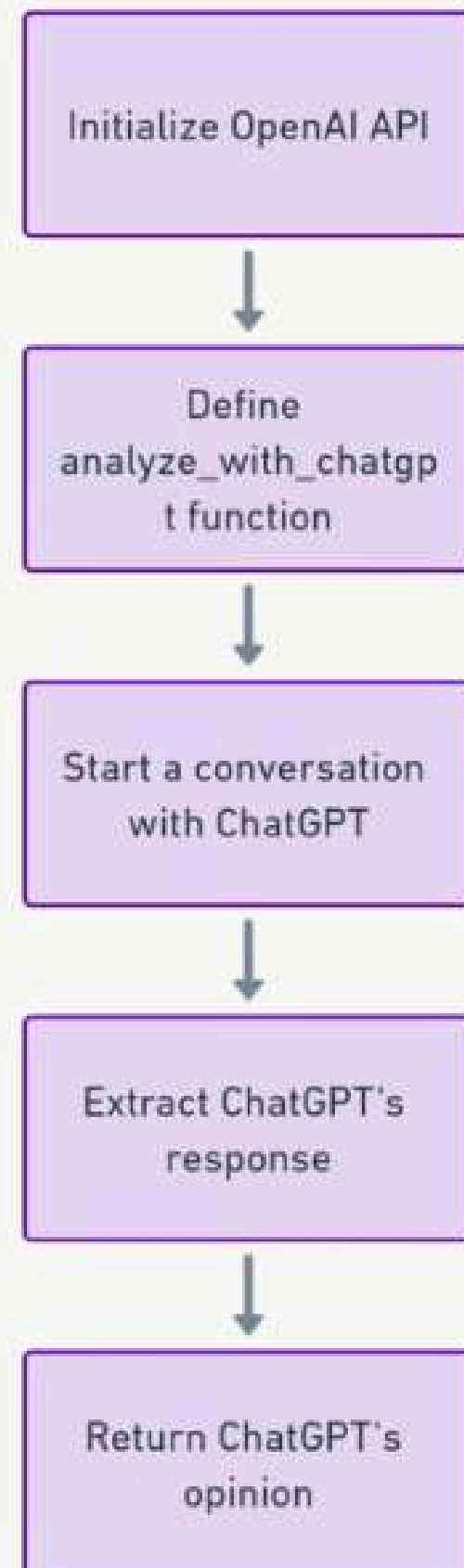
# Harnessing ChatGPT for Enhanced Malware Behavior Analysis



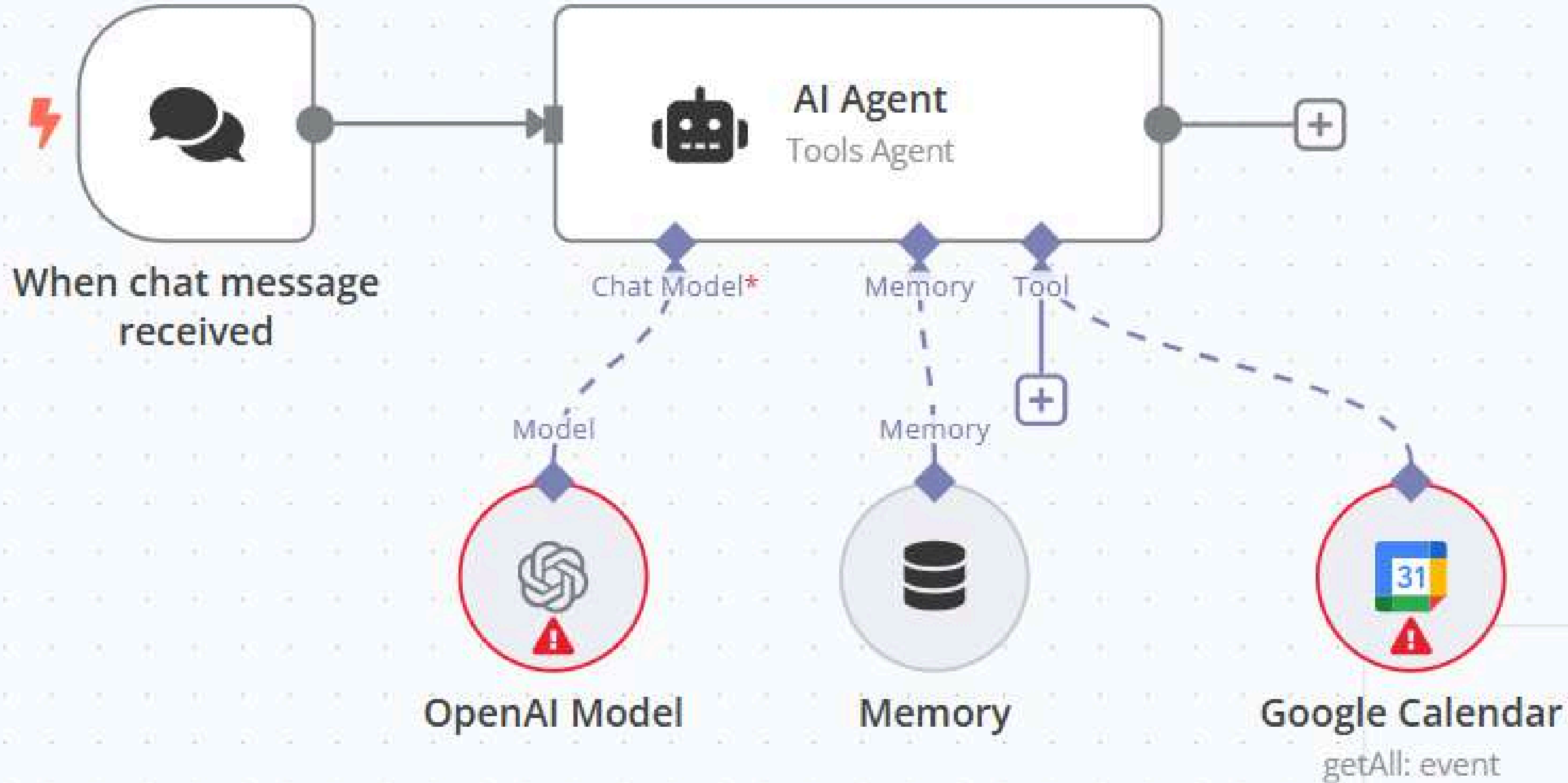


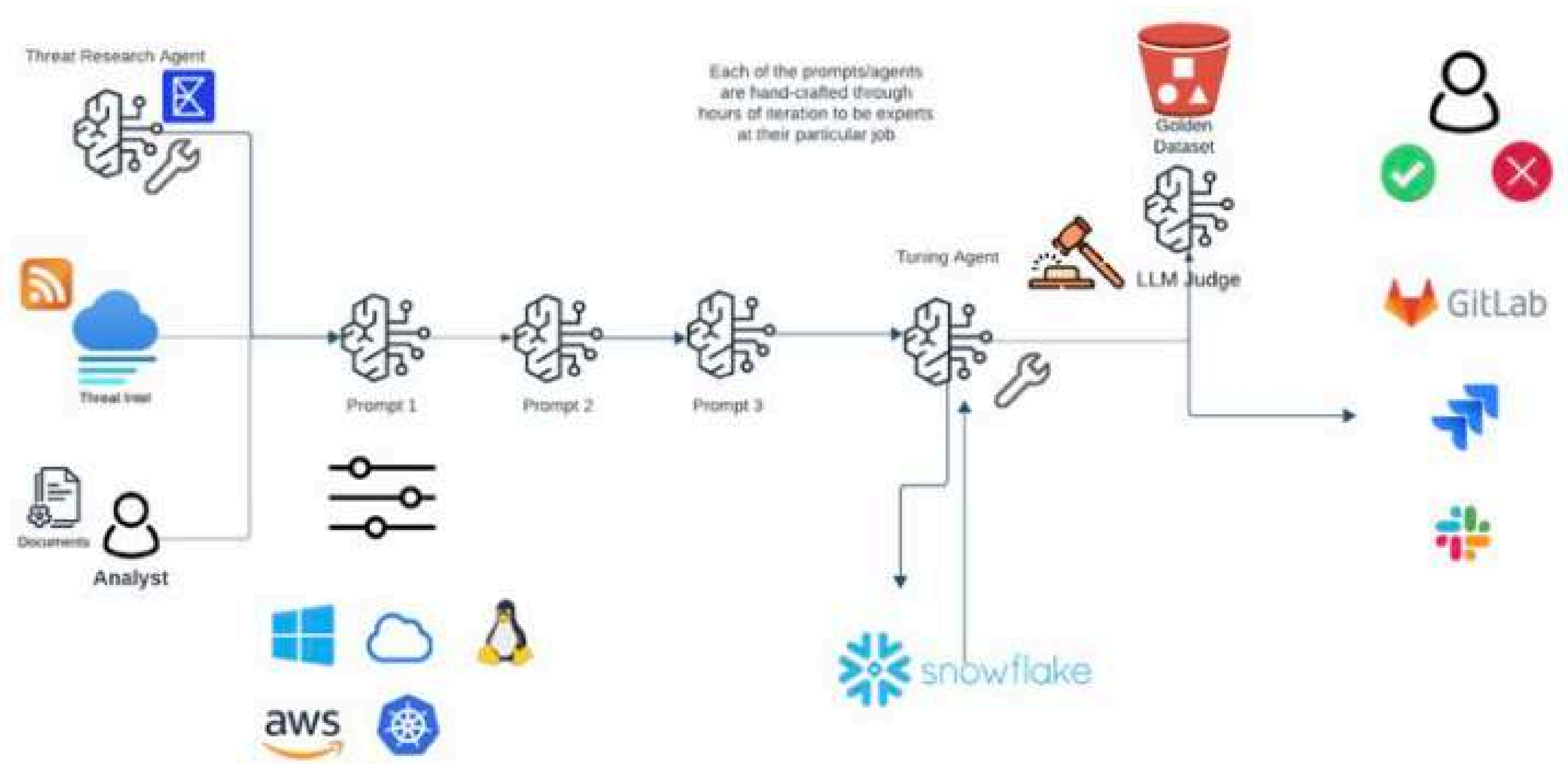
















[HTTPS://GITHUB.COM/EVILBYTECODE/GODEFENDER](https://github.com/evilbytecode/godefender)  
[CANVA=1  
HTTPS://N8N.IO/WORKFLOWS/5937-AUTOMATED-URL-PHISHING-AND-THREAT-  
ANALYSIS-WITH-NIXGUARD-AI/](https://n8n.io/workflows/5937-automated-url-phishing-and-threat-analysis-with-nixguard-ai/)





# THANK YOU!



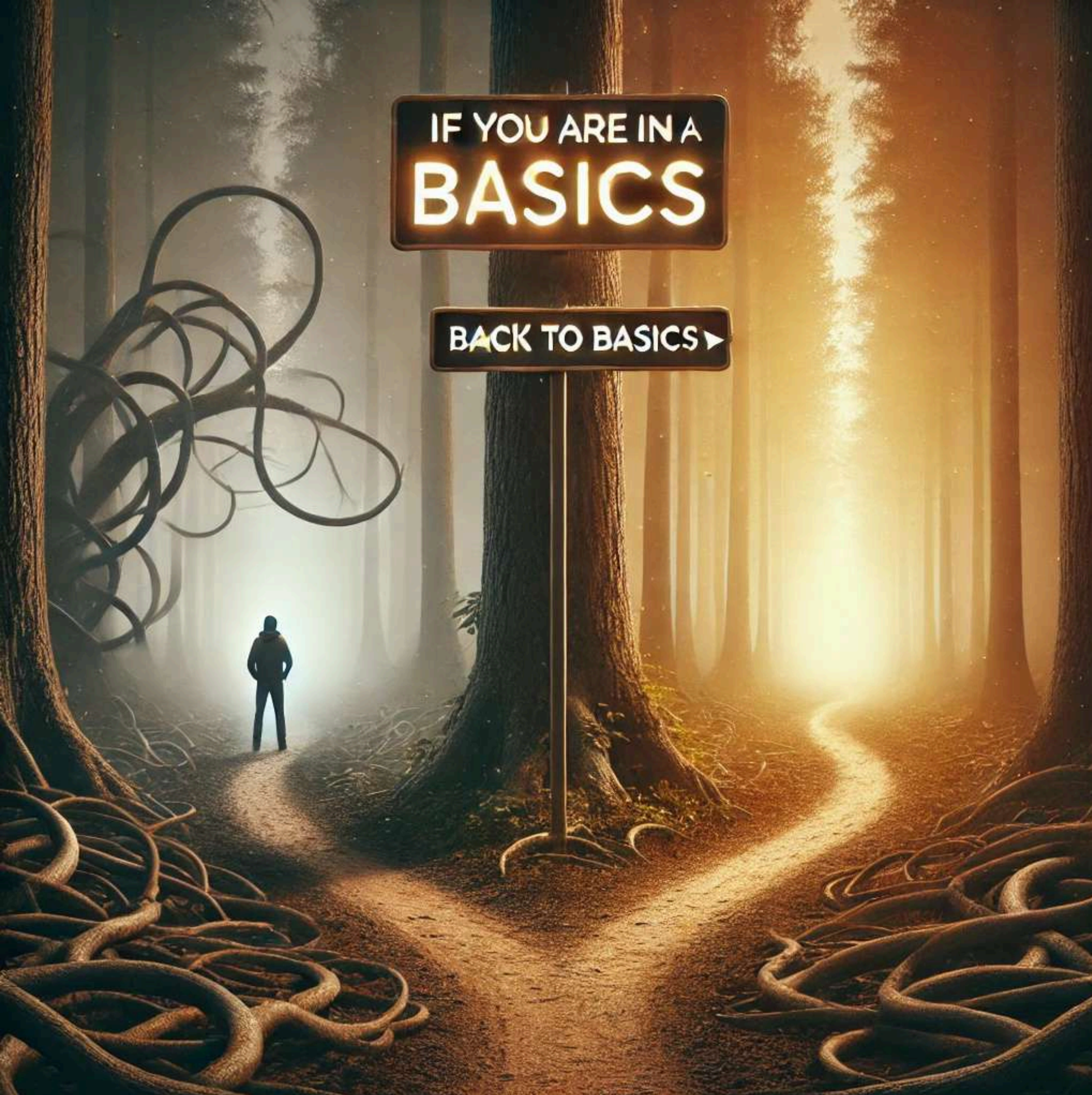
[HTTPS://DIGISURAKSHA.ORG/](https://digisuraksha.org/)











Back To Basics