# ALA-3 CNS

**AIM: - Malware Analyzer:**

In this activity, students will develop a Python-based malware simulator that demonstrates the behaviour of Viruses, Worms, and Trojans in a safe and controlled environment. The program will simulate how these threats spread, replicate, or disguise themselves without causing actual harm. Students will analyse the behaviour of each type of malware and generate a detailed report explaining their impact. They will submit the code and a screenshot of the simulation results on the GMIU Web Portal.

**SOLUTION:**

## REPORT

- **Project Overview:**
  - The project, hosted at malware-attacks.netlify.app, is a Python-based malware simulator designed to demonstrate how different types of malware function in a safe and educational environment. Its purpose is to help users understand the behavior, spread, and impact of common malware types — specifically Viruses, Worms, and Trojans — without causing any harm to real systems.
  - The simulator executes controlled demonstrations that mimic how these malicious programs operate internally, showing users their replication methods, infection patterns, and payload delivery mechanisms.

## Objective:

**The main goal of the project is to:**

- Educate students and cybersecurity learners about malware behavior.
- Provide a virtual lab simulation that visualizes how malware spreads and affects systems.
- Help users differentiate between various malware categories through interactive examples.

Core Functionalities

Simulation Engine (Python-based):

- Mimics how malware infects and propagates within a system.
- Uses code-based demonstrations for real-time visualization.

Safe Execution Environment:

- No actual system damage occurs; all activities are simulated.
- Allows users to safely study malware behavior patterns.

Educational Focus:

- Highlights the technical differences between Viruses, Worms, and Trojans.
- Teaches users the principles of cybersecurity defense against these attacks.

Technologies and Languages Used

- HTML — Structures the web pages and simulation controls.
- CSS — Styles the site and provides responsive layout.
- JavaScript — Implements frontend interactivity, visualization, and client-side simulation controls.
- Python — Runs the backend simulation engine, sandbox abstractions, logging, and analysis (e.g., via Flask/Django API).

## Types of Malwares Demonstrated:

### 1. Virus

**Definition:** A malicious program that attaches itself to legitimate files and executes when the host file runs.

**Behaviour Simulated:**

- Infects files by inserting malicious code.
- Spreads only when the infected file is executed.

**Purpose in Simulation:** Shows how viruses replicate and damage user files.

### 2. Worm

**Definition:** A self-replicating program that spreads across networks without user action.

**Behaviour Simulated:**

- Automatically copies itself to connected systems.
- Demonstrates exponential spreading.

**Purpose in Simulation:** Illustrates how worms cause network slowdowns and large-scale infections.

### 3. Trojan

**Definition:** A deceptive program that appears harmless but executes harmful actions once installed.

**Behaviour Simulated:**

- Masquerades as a useful program.
- Once opened, it creates backdoors or steals data.

**Purpose in Simulation:** Teaches how social engineering is used to trick users into installing malware.

- **Website Link:**
  https://malware-attacks.netlify.app/