Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Semester: (Fall, Year: 2025), B.Sc. in CSE (Day)

Cyber Immunity Grid (CIG)
Self Healing Smart Network

Course Title : Project Design I
Course Code : CSE-308
Section : 231-D1

Students Details

Name	ID
Promod Chandra Das	231002005
Md. Shahed Shamim Rony	231902019
Chinmoy Debnath	231902029

Submission Date : 26-10-25
Course Teacher's Name : Rusmita Halim Chaity

[For teachers use only: Don't write anything inside this box]

Lab Project Status	
Marks:	**Signature:**
Comments:	**Date:**

Contents

Chapter 1

Introduction

1.1 Objective(s)

- To design and develop a web-based simulation system that shows a smart, self-healing network using the concept of cyber immunity.
- To build a simple and interactive dashboard interface that shows how many devices are connected, their health status, infection level, and ongoing healing process.
- To visualize the behavior of the network during cyber-attacks and automatic healing.
- To understand how a self-healing cyber system can detect, isolate, and recover from attacks without manual action.

1.2 Problem Analysis and Motivations

In recent years, cyber threats have increased rapidly. Traditional networks only detect and report issues but cannot automatically fix them. Once a device is infected, it may spread the attack to others, causing system-wide failure.

To solve this issue, our project Cyber Immune Grid proposes a network that behaves like a human immune system. Just like the human body detects and heals infections automatically, this smart network detects abnormal activity, isolates infected nodes, and begins healing them automatically.

Our motivation is to simulate this process in a web-based dashboard that is simple but meaningful. The simulation will show how a future self-healing network might work and how it can make systems more secure and resilient.

1.3 Literature Review

Cybersecurity systems today increasingly depend on visualization and automation to protect networks from modern threats. According to several research works and studies:

- Modern cybersecurity focuses on dashboard systems that visualize protection and monitoring in real time, helping users understand system health and security status effectively [1].
- Web-based dashboards are being developed for Cyber Emergency Response Teams (CERTs) to track and respond to network threats through interactive interfaces [2].
- Real-time visualization platforms collect and display threat intelligence data, allowing better decision-making for protection systems and security management [3].

Our project is inspired by these research works but aims to present a simplified, educational version of such protection dashboards. The goal is to design a web-based system that simulates how devices are monitored, how attacks affect them, and how recovery or healing is visualized through an interactive dashboard.

We have decided to choose this project because recent studies show that visualization and automation play a key role in understanding and improving cybersecurity systems. Inspired by these works, our project demonstrates how web-based monitoring and self-healing logic can be presented through a simple simulation model using HTML, CSS, PHP, and MySQL.

1.4 Methodology

This project adopts a prototype development approach, focusing on building a conceptual and visual model of a self-healing smart network inspired by biological immunity. Since the project is in its early proposal stage, the exact implementation details and algorithms will evolve through further research, testing, and refinement. The current methodology outlines the general structure and intended workflow for developing the prototype.

1.4.1 System Modules

The prototype will contain the following major modules:

- **Home Page:** Displays the overall system overview and entry point to the simulation.
- **Device Status Page:** Lists connected devices, showing their health and infection status.
- **Infected List:** Displays all devices currently affected by simulated cyber-attacks.
- **Attack Severity:** Represents infection levels using color codes and severity percentages.
- **Healing Process Page:** Shows the automated recovery of infected devices with progress bars.
- **Exit Page:** Allows resetting or closing the simulation.

1.4.2 System Flow

1. The user launches the dashboard in a web browser.
2. The system loads simulated devices and initializes their health status.
3. Randomized attack events are generated to represent infection spread.
4. The automated healing process begins, updating device states dynamically.
5. The dashboard visually updates device colors and data (e.g., Green = Safe, Red = Critical).

1.4.3 Tools and Technologies

- **Front-End:** HTML, CSS, (Optionally JavaScript for dynamic updates)
- **Back-End:** PHP, MySQL
- **Server Environment:** XAMPP (Localhost)
- **Development Tools:** VS Code

1.5 Feasibility Study

1.5.1 Technical Feasibility

- **System Accuracy :**Expected accuracy 85–90 percent for infection/healing simulation.
- **Growth Potential :**Can be upgraded to include AI for real network threat detection.
- **Response Time :**Each system update takes less than 10 seconds.
- **User Friendliness :**Very easy and colorful dashboard with simple navigation.

1.5.2 Operational Feasibility

- **Operation Time :**Always runs in a browser, real-time simulation.
- **Reliability :**PHP and MySQL ensure data consistency.
- **Accuracy and Retrieval Rate :**Data is updated instantly when devices are infected or healed.

1.5.3 Economical Feasibility

- **System Performance :**Lightweight and smooth for demonstration.
- **System Operation :**Runs on any normal computer with localhost setup.
- **User Training :**No special training required — simple web interface.
- **Budget and Resources :**
  - **Development Tools:** Free (VS Code, XAMPP)
  - **Team Members:** 3
  - **Estimated Budget:** Around 5,000 - 10,000 BDT for printing, presentation and reports.

1.6 Main Phases

The development of the **Cyber Immune Grid: Self-Healing Smart Network** project follows an **Agile Software Development Life Cycle (SDLC)** approach. The project is divided into several short and manageable phases called sprints, where each sprint delivers an incremental improvement of the system. The main phases of development are described below:

1. **Project Proposal and Planning:** In this first phase, our team selected and discussed the main idea of the project and clearly defined its scope, objectives and expected outcomes. During this stage, we also divided our team responsibilities, identified the resources we would need, and prepared a schedule to make sure the project progresses smoothly.
2. **Requirement Specification:** In this phase, our team discussed and listed both functional and non-functional requirements for the project. The functional requirements include monitoring connected devices, detecting attacks, showing the healing process, and generating basic reports. The non-functional requirements focus on making the system fast, reliable, and easy to use.
3. **SDLC Selection:** The project follows the **Agile Methodology**, which focuses on teamwork, flexibility, and continuous improvement. This approach allows our team to make changes whenever needed and adjust easily during the development process. Agile helps us work together more efficiently, review progress regularly, and make improvements step by step. Since our project is small and requires frequent updates, we chose this methodology to keep our work simple, organized, and adaptable.

1.7 Detailed Working Plan

SL	Task	Duration	Responsible Person	Phase
1	Requirement Specification and Data Collection	3 weeks	Rony	Research and Planning
2	Requirement Finalization	2 weeks	Promod	Analysis
3	System Design and Modeling	2 weeks	Chinmoy	Design
4	System Modeling and Finalization	2 weeks	Rony,Promod,Chinmoy	Design

Table 1.1: Detailed Working Plan of the Project

1.8 Gantt Chart of Project Development Timing

Weeks	1	2	3	4	5	6	7	8	9
Project Activities	▬	▬	▬						
Planning				▬	▬				
Design						▬	▬	▬	▬

Table 1.2: Gantt Chart of Project Development Timing

1.9 Budget Details of the Cyber Immune Grid System

SL	Criteria	Cost specification	Existing system (tk)	New system (tk)
1	Office Cost	Team meeting	25,000	20,000
		Project meeting	25,000	20,000
		First aid	1,000	500
2	Website Cost	Website maintenance	2,000	1,500
3	Office Equipment Cost	Computer	10,00,000	9,50,000
		Laptop	2,00,000	1,50,000
		Server / Simulator setup	3,00,000	2,50,000
		CC Camera / Network Tools	1,00,000	1,00,000
4	Salary Cost	Team Leader	2,00,000	2,50,000
		System Designer	80,000	70,000
		Software Engineer	1,00,000	1,00,000
		Developer	50,000	50,000
		QA Tester	40,000	38,000
		Support Staff	10,000	9,000
		Total cost	**21,93,000**	**20,69,000**

Table 1.3: Budget Details of Cyber Immune Grid System

1.10 Discussion & Conclusion

The **Cyber Immune Grid: Self-Healing Smart Network** system is designed to detect cyber-attacks in a simulated environment and perform automatic healing operations on affected devices. The system continuously monitors all connected nodes, identifies infected or abnormal behaviors, and initiates a self-recovery process without manual intervention.

Our web-based simulation demonstrates how an intelligent network can manage cyber threats in real time by detecting, isolating, and healing infected devices. The dashboard visually represents device status, infection severity, and the progress of ongoing healing actions, providing a clear view of the system's resilience and responsiveness.

This project highlights the potential of self-healing mechanisms and establishes a foundation for developing more advanced cyber-immune systems. By integrating automated detection and recovery processes, the Cyber Immune Grid strengthens the concept of **autonomous, resilient, and self-defending networks** for future smart infrastructures.

References

[1] Hanning Zhao and Bilhanan Silverajan. Evaluating cyber security dashboards for smart cities and buildings. In *19th International Conference on Availability, Reliability and Security (ARES 2024)*, 2024.

[2] Marc-André Kaufhold, Ali Sercan Basyurt, Kaan Eyilmez, et al. Cyber threat observatory: Design and evaluation of an interactive dashboard for computer emergency response teams. In *European Conference on Information Systems (ECIS 2022)*, 2022.

[3] G. Madhan, B. Manoj, M. Nithish Kannan, and K.S. Arun. Cyber threat intelligence dashboard: A real-time visualization platform. *EPRA International Journal of Research & Development (IJRD)*, 10(5), 2025.