Lab 1: Setup Lab Environments

Thangamuthu
Balaji

I. INTRODUCTION

This report details how the lab environments for the Cyber Defense course were created and what results came from it. The main goal of this lab was to build a basic setup for running cybersecurity exercises during the semester. The work included setting up a virtual machine (VM), installing a web application for testing security, creating accounts on different platforms, and setting up remote access to the lab resources.

II. VIRTUAL MACHINE SETUP

1. The initial task was to set up a virtual machine (VM) using Virtual Machine Manager. For this VM, we chose Ubuntu 20.04.6 LTS (Desktop version) and assigned it 4096 MB of RAM, 2 CPU cores, and 50 GB of storage space. This VM will be the primary environment for testing and using security tools throughout the course. We opted for a virtual network with a 'default' NAT configuration to ensure the VM has internet access. During the installation, we selected the option to "Erase disk and install Ubuntu," and we went with a "Normal installation" setup, skipping any extra updates or third-party software at the start.

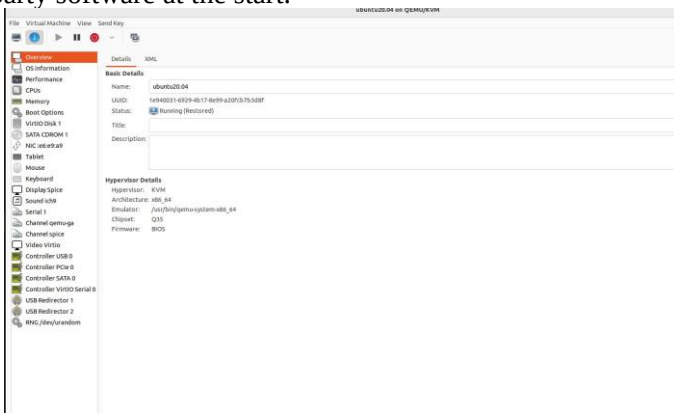

Fig. 1. VM Setup

III. WEBGOAT AND WEBWOLF INSTALLATION

WebGoat is a purposely insecure web application created by OWASP to help people learn about web application security. It was installed on an Ubuntu virtual machine using the standalone version, and it requires Java 17 to run. This setup creates a safe space to practice finding and exploiting typical web vulnerabilities. Alongside it, WebWolf was also installed to mimic real-life hacker activities, like dealing with files, managing emails, and handling incoming requests. The features of WebWolf, especially in this controlled setting, make it possible to practice safely.exploration of potential attack vectors that attackers might use in real-world scenarios.

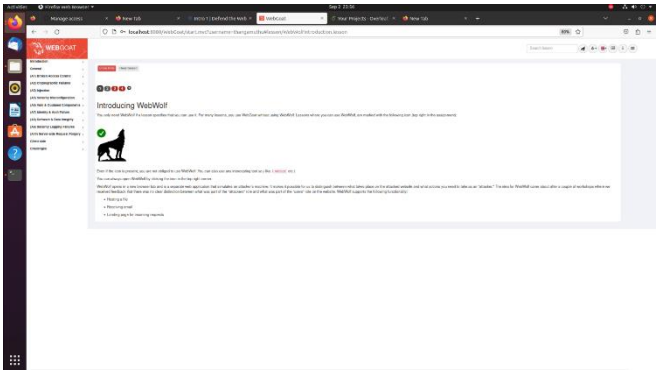

Fig. 2. Webgoat

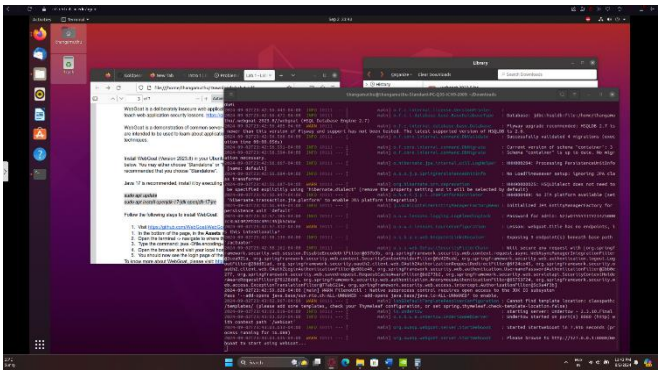

Fig. 3. Webgoat Installation

IV. OVERLEAF ACCOUNT CREATION

An Overleaf account was made to help write and submit lab reports in LaTeX format. This tool was picked because it’s easy to work with and great for formatting scientific papers. We used a template provided by the course to make sure all the reports follow the same format and guidelines.

V. GITHUB REPOSITORY SETUP

A GitHub repository was set up on github.iu.edu to keep all our course documents, like lab reports, in one place. The repository is neatly organized so that it's easy to find files and look back at previous work as the course moves forward. The instructor and AI were added as collaborators to help with feedback and grading.

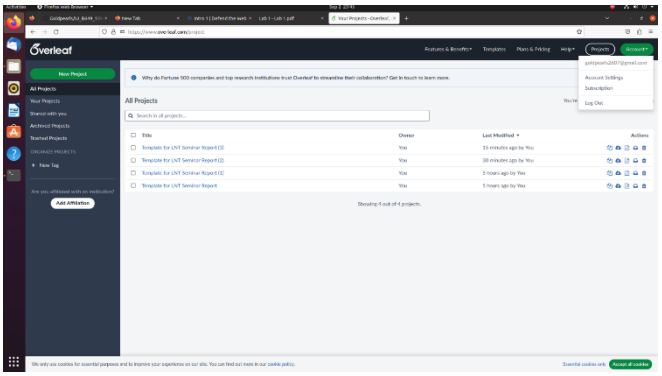

Fig. 4. Overleaf

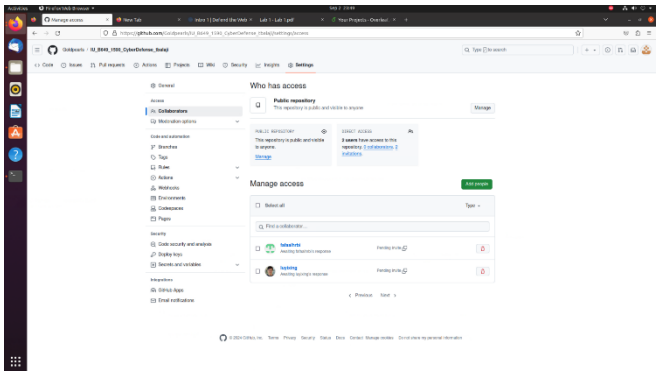

Fig. 5. Github

VI. REMOTE DESKTOP CONFIGURATION

To allow remote access to lab resources, the IU Research Desktop (RED) was set up to enable secure SSH connections to the lab systems. This configuration is essential for carrying on lab work beyond the physical lab space and offers the convenience of accessing the virtual machine from a distance. The setup process included creating a Carbonate account and using the terminal to start remote sessions.

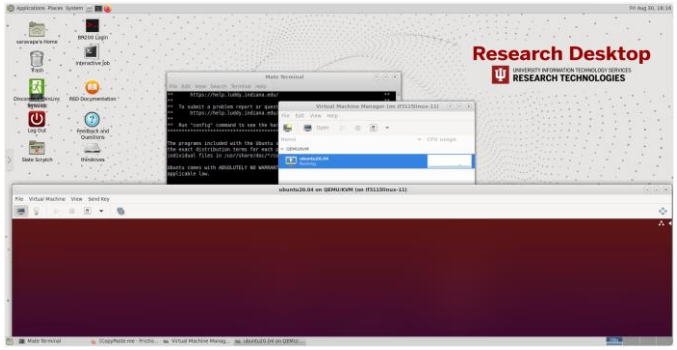

Fig. 6. Remote Desktop

VII. CONCLUSION

1. Lab 1 was a great introduction to how to set up and manage a cybersecurity testing environment. During this lab, I figured out how to configure virtual machines, install and use security applications like WebGoat and WebWolf, and organize documents and repositories with tools such as Overleaf and GitHub. The skills I picked up are really useful for both school and future jobs, especially in cybersecurity roles. For next time, it would be cool to automate some of the VM setup and dive deeper into more complex attack scenarios in WebGoat and WebWolf to boost my knowledge and skills in cyber defense.

REFERENCES

[1] WebGoat Project. (2023). WebGoat: A Deliberately Insecure Web Application. Retrieved from [WebGoat GitHub](https://github.com/WebGoat/WebGoat)

[2] Indiana University. (2024). IU Research Desktop (RED) Documentation. (https://red.uits.iu.edu/)

[3] OWASP Foundation. (2023). OWASP WebWolf. Retrieved from [OWASP WebWolf](https://github.com/WebGoat/WebGoat/wiki/WebWolf)