

# Web Application Programming and Hacking

**Instructor:** Dr. Phu Phung

**Student**

**Name:** Mahitha Kalaga

**Email:** kalagam1@udayton.edu



## Repository Information

Repository's URL: <https://github.com/MahithaKalaga-cyber/waph-mahitha.git>

This is a private repository for Mahitha Kalaga to store all code from the course. The organization of this repository is as follows.

## Labs

Hands-on exercises in Lectures

- Lab 0: Development Environment Setup

## Report

### The lab's overview

This lab focused on setting up the development environment for WAPH. In Part I, I installed Ubuntu 22.04 in VirtualBox and configured the system with essential packages. In Part II, I cloned both the public course repository and my private GitHub repository, and completed git-related tasks.

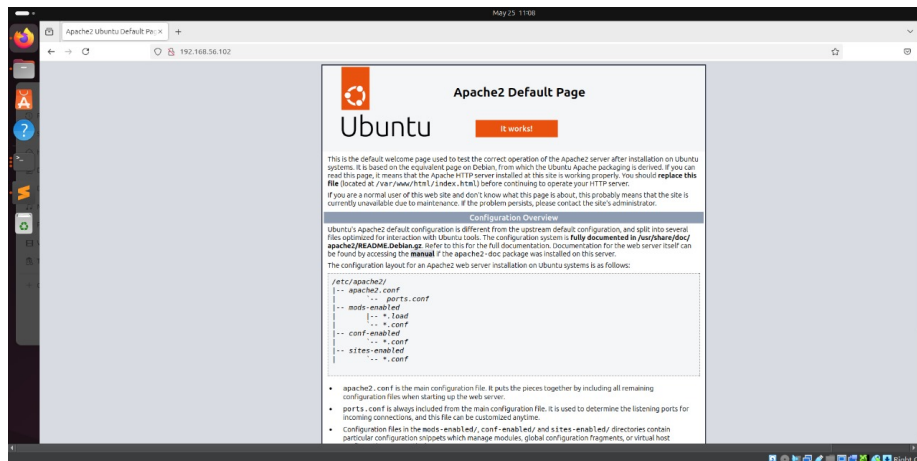
Lab's URL: Lab0

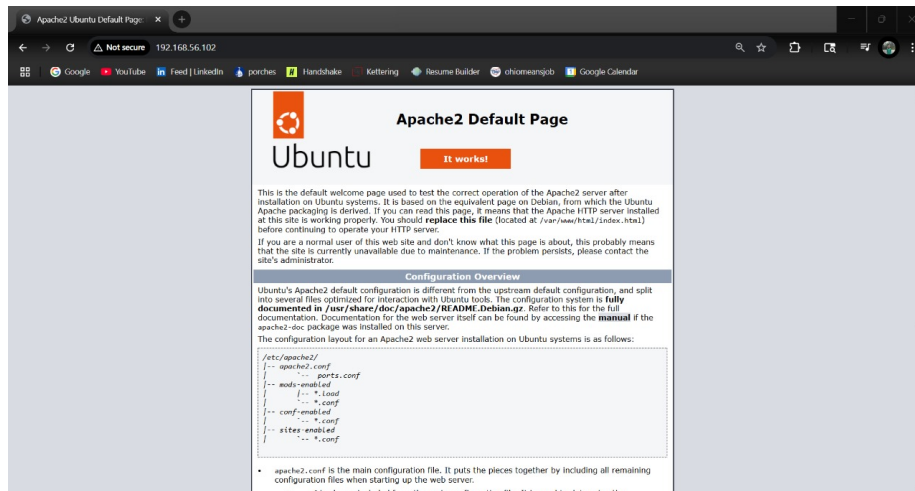
## Part 1 - Ubuntu Virtual Machine & Software Installation

Created a new VirtualBox VM using the Ubuntu 22.04 ISO and allocated 4 GB RAM and 2 CPUs. After the OS was installed, I set up the following:

- Installed system updates
- Installed Apache2, Git, and Sublime Text
- Installed Google Chrome manually via .deb package
- Installed Wireshark and configured non-root access

### Apache Web Server Testing



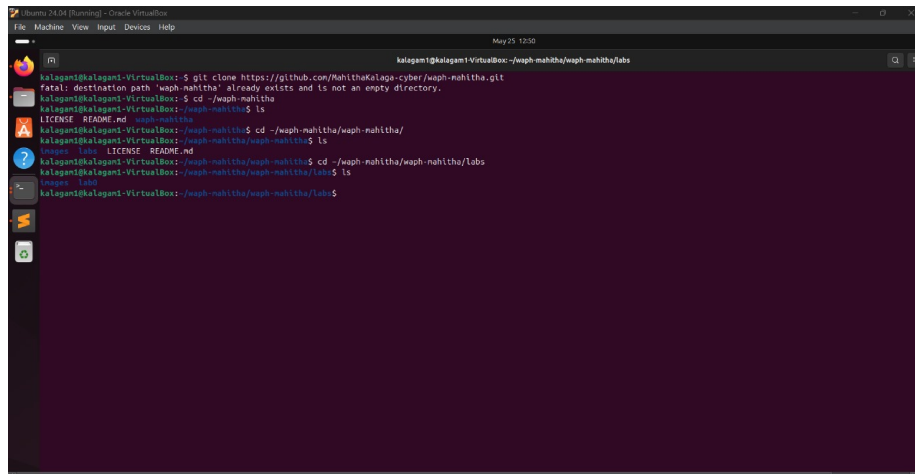


## Part 2 - git Repositories and Exercises

I created a private GitHub repository named waph-mahitha and shared it with the instructor by adding waph-phung as a collaborator. The repository is available at

### The course repository

Repository's URL: <https://github.com/MahithaKalaga-cyber/waph-mahitha.git>



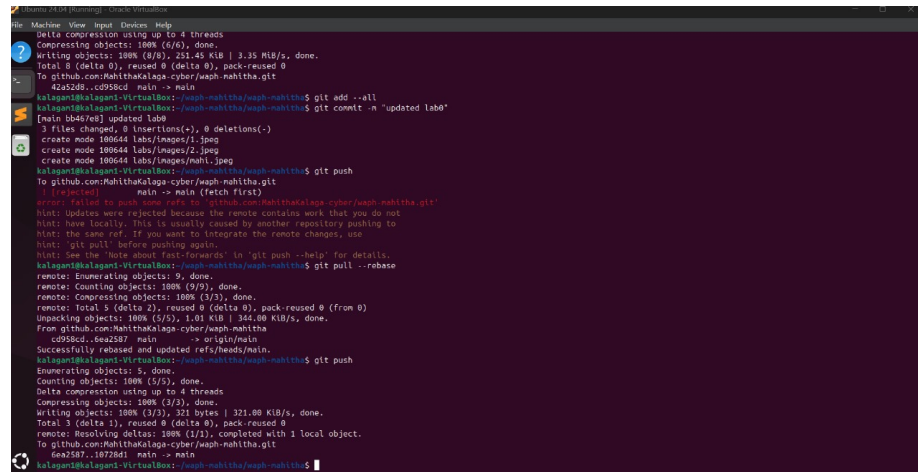
### Private Repository

To enable secure access, I generated an SSH key pair on my Ubuntu VM and added the public key to my GitHub account. This allowed me to clone the

repository via SSH without needing a password.

After cloning, I created the labs/lab0/ directory, added the lab report in README.md, and included my headshot image using a relative path. I then committed the changes and pushed them to GitHub.

Below is a screenshot showing the successful git commit and git push from my VM:



```
kalagani@kalagani-VirtualBox: /vash-mahitha/vash-mahitha$ git add --all
To github.com:MahithaKalaga-cyber/vash-mahitha.git
 42a52d8..cd958cd  main -> main
kalagani@kalagani-VirtualBox: /vash-mahitha/vash-mahitha$ git commit -m "updated lab0"
[main bb467e8] updated lab0
 3 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 labs/images/1.jpeg
 create mode 100644 labs/images/2.jpeg
 create mode 100644 labs/images/mahi.jpeg
kalagani@kalagani-VirtualBox: /vash-mahitha/vash-mahitha$ git push
To github.com:MahithaKalaga-cyber/vash-mahitha.git
 [rejected]        main -> main (fetch first)
error: failed to push some refs to 'github.com:MahithaKalaga-cyber/vash-mahitha.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forward' in 'git push -help' for details.
kalagani@kalagani-VirtualBox: /vash-mahitha/vash-mahitha$ git pull --rebase
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 5 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (5/5), 1.08 KiB | 344.00 KiB/s, done.
From github.com:MahithaKalaga-cyber/vash-mahitha
 cd958cd..6ea2587  main -> origin/main
Successfully rebased and updated refs/heads/main.
kalagani@kalagani-VirtualBox: /vash-mahitha/vash-mahitha$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), 321 bytes | 321.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:MahithaKalaga-cyber/vash-mahitha.git
 6ea2587..10728d1  main -> main
kalagani@kalagani-VirtualBox: /vash-mahitha/vash-mahitha$
```