

CRYPTONITE — TASK PHASE I

Congratulations for making it through the initial recruitment test.

You are now a part of **Cryptonite's** first task phase. This will last 20 days, with regular progress updates and a final interview with your mentor.

For this stage, you're going to be completing a set of modules on the website **pwn.college**.

This process first and foremost involves setting up a Linux environment, creating a pwn.college account, and learning and verifying your SSH keys.

Linux Setup

Windows

Download Windows Subsystem for Linux. This is a virtual installation of Linux built on top of Windows. Install the Ubuntu 22.04 version, it generally works smoothly and is rock-solid stable.

Link: <https://apps.microsoft.com/detail/9pn20msr04dw>

Once downloaded and installed, click on the orange icon that shows up in your apps menu. It will open Ubuntu Linux directly and put you into a terminal.

While we recommend WSL for beginners, you can try using VMware to run your own Linux virtual machine if you wish.

MacOS

You don't need to install any environment, since your operating system will already have SSH working and is generally close enough to GNU/Linux to not cause any issues.

SSH Setup

To setup SSH on **WSL** follow the steps given below.

- Run the command `ssh-keygen -f key -N ''`
- Two files should have been created, `key` and `key.pub`.
- Copy the *contents* of `key.pub` by outputting its contents to the terminal using the command `cat key.pub`. It should start with `ssh-ed25519...` and have a bunch of characters after it.

Your SSH key has been successfully generated. Now to use it with pwn.college:

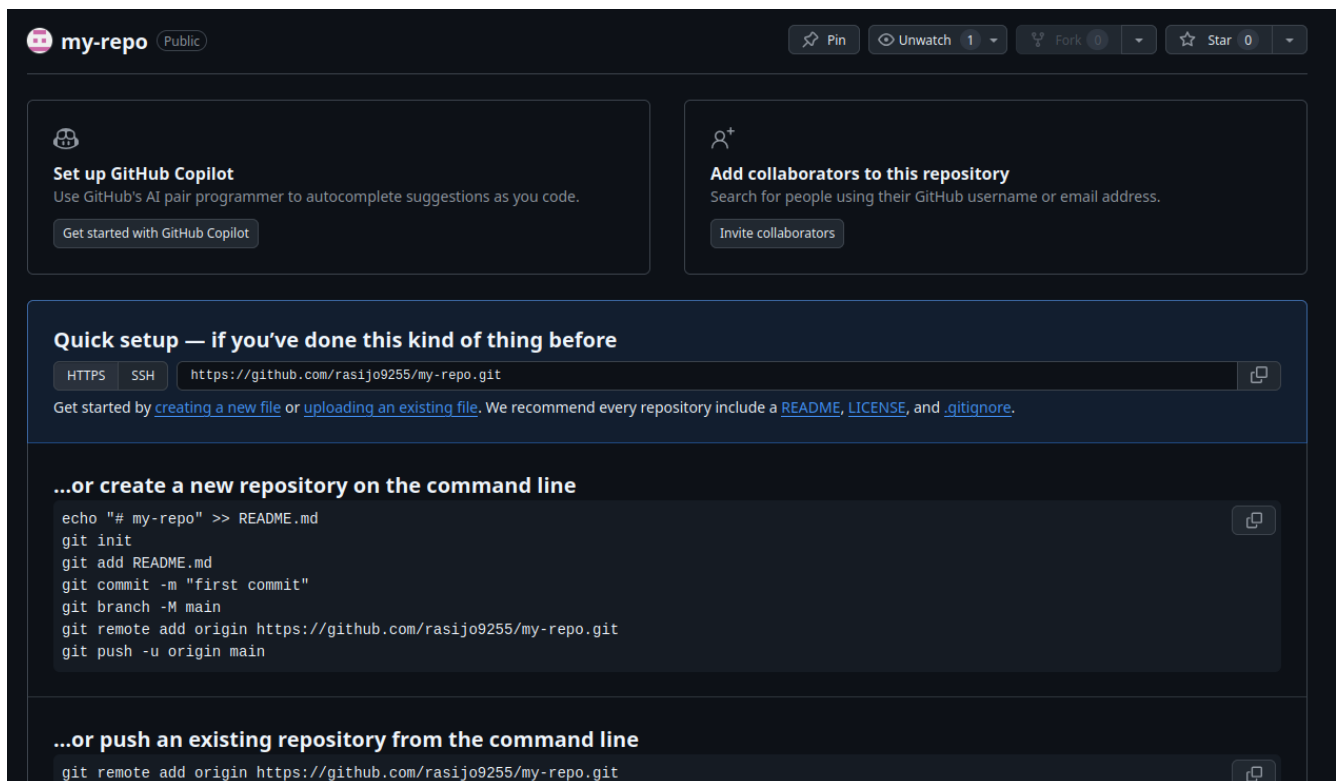
- Make a pwn.college account.
- Go to the settings tab. (it should be on the top bar with the gear icons).
- Go to "SSH Key".
- In "Current SSH Keys", input the contents of the `key.pub` file you copied.

You're now ready to start using SSH with pwn.college.

GitHub

All your work in the task phase will need to be uploaded on a **public GitHub repository**. Here's how you need to set it up.

1. Create an account on GitHub and verify it using your email.
2. Create a repository called `"cryptonite_taskphase_yourfirstname"`.
3. You will be greeted to a screen like this.



4. Run the command `git clone <repo_url>` where `repo_url` is what will show up in place of `https://github.com/rasijo9255/my-repo.git` in the dark blue highlighted box.
5. A folder should have been created on your WSL system with the name of your repo. You need to change your directory to it. In this example it is `my-repo`, so I will run `cd my-repo` to go into the repo's folder.
6. Once you're in the folder of your repo, run the commands that are shown in the box with the heading *"... or create a new repository on the command line"*.
7. This will be your first *"commit"* to the repository.

You will need to get comfortable with the command line and using it to push your changes to GitHub.

A good resource to learn the Git command line is this: <https://www.youtube.com/watch?v=ToZDrki0mqc>

pwn.college

pwn.college is an educational cybersecurity platform designed to train beginners. They have several great dojos to train you in domains of cybersecurity, but we'll focus on **Linux Luminarium**, designed to get you acquainted with navigating and utilizing the Linux command line.

But first we recommend you go through the **Getting Started** dojo to acquaint you with the working of the site and how you use it.

Link to the Getting Started Dojo: <https://pwn.college/welcome/welcome/>

Link to Linux Luminarium Dojo: <https://pwn.college/linux-luminarium/>

In the link above, you'll see the list of modules the dojo contains.

This dojo has **12 modules**, each focusing on a specific domain on operating the command line (CLI).

Each module has a number of **challenges**. On completion of each challenge, you will be rewarded with a **flag**. This will be in the form `pwn.college{Random Characters}`.

An example of a flag is as follows

pwn.college{xnPgKuFmaeiqB0SEbDqThhLFQm2.qqwAlHQYjtGA5DmJ}

An example of the front page of a pwn.college module is below

The screenshot shows the 'Intro to Commands' challenge interface. At the top, the title 'Intro to Commands' is displayed in blue, with a flag icon to its left and '3256 solves' to its right. Below the title, a text box explains the challenge: 'In this challenge, you will invoke your first command! When you type a command and hit enter, the command will be invoked, as so:'. This is followed by a terminal window showing a command prompt 'hacker@dojo:~\$' where the user enters 'whoami', resulting in the output 'hacker'. Below the terminal, a text box explains: 'Here, the user executed the `whoami` command, which simply prints the username (`hacker`) to the terminal. When the command terminates, the shell once again displays the prompt, ready for the next command. In this level, invoke the `hello` command to get the flag! Keep in mind: commands in Linux are case sensitive: `hello` is different from `HELLO`.' At the bottom of the challenge area, there is a 'Start' button with a play icon. Below this, there is a 'Flag' input field and a 'Submit' button. At the very bottom of the screenshot, the next challenge 'Intro to Arguments' is partially visible, with '3173 solves'.

Press the **Start** button to start the server that you will SSH into. When a blue box shows up with the text **Challenge successfully started!**, the server has been started.

Use the command:

```
ssh -i ./key hacker@dojo.pwn.college
```

to transfer into the server where you'll be solving the challenge. Further instructions will be given in the challenge's description.

For every module, write a report detailing your thought process and how you went about solving each challenge. The format of the report should strictly be in Markdown, the simplest way to share text.

Markdown Guide: <https://docs.github.com/en/get-started/writing-on-github/getting-started-with-writing-and-formatting-on-github/basic-writing-and-formatting-syntax>

The markdown file must be uploaded to your **GitHub** and shared to your mentor within 10 days at most.

Every module's writeup you do should be written in a separate file. For example, if you finished the module **File Globbing**, you should upload your writeup in the file name **File_Globbing.md**. Also, if you used any references or videos to help you through the process, make sure to include them in your writeup.

Day 1-10

In the 10 days, you need to finish the following modules **in full**.

- **Hello Hackers**
- **Pondering Paths**
- **Comprehending Commands**
- **Digesting Documentation**

- File Globbing
- Practicing Piping

Day 11-20

For the remaining 10 days, you need to finish the following modules **in full**.

- Shell Variables
- Processes and Jobs
- Perceiving Permissions
- Untangling Users
- Chaining Commands
- Pondering PATH