Name: Juanson, Aliya Dane P.	**Date Performed: September 9, 2024**
Course/Section:CpE31S2	**Date Submitted: September 11, 2024**
Instructor: Sir Robin Valenzuela	**Semester and SY:**

Activity 2: SSH Key-Based Authentication and Setting up Git

1. **Objectives:**

1.1 Configure remote and local machine to connect via SSH using a KEY instead of using a password

1.2 Create a public key and private key

1.3 Verify connectivity

1.4 Setup Git Repository using local and remote repositories

1.5 Configure and Run ad hoc commands from local machine to remote servers

Part 1: Discussion

It is assumed that you are already done with the last Activity (**Activity 1: Configure Network using Virtual Machines).** *Provide screenshots for each task*.

It is also assumed that you have VMs running that you can SSH but requires a password. Our goal is to remotely login through SSH using a key without using a password. In this activity, we create a public and a private key. The private key resides in the local machine while the public key will be pushed to remote machines. Thus, instead of using a password, the local machine can connect automatically using SSH through an authorized key.

What Is ssh-keygen?

Ssh-keygen is a tool for creating new authentication key pairs for SSH. Such key pairs are used for automating logins, single sign-on, and for authenticating hosts.

SSH Keys and Public Key Authentication

The SSH protocol uses public key cryptography for authenticating hosts and users. The authentication keys, called SSH keys, are created using the keygen program.

SSH introduced public key authentication as a more secure alternative to the older .rhosts authentication. It improved security by avoiding the need to have password stored in files and eliminated the possibility of a compromised server stealing the user's password.

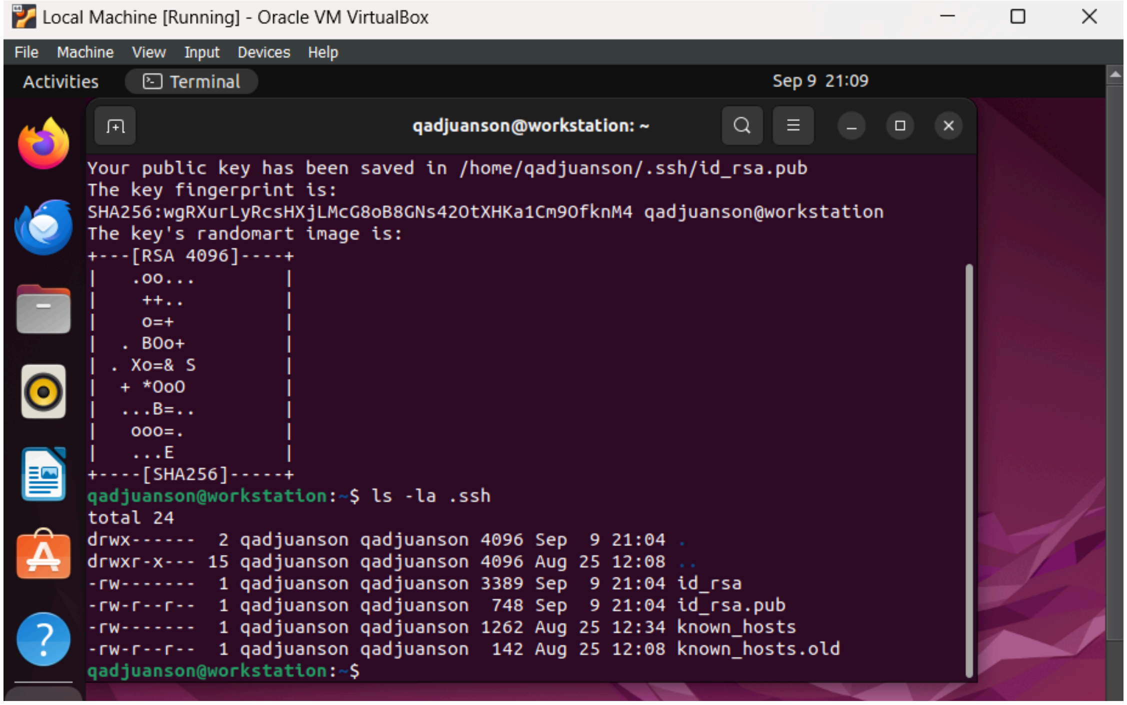
However, SSH keys are authentication credentials just like passwords. Thus, they must be managed somewhat analogously to usernames and passwords. They should have a proper termination process so that keys are removed when no longer needed.

Task 1: Create an SSH Key Pair for User Authentication

1. The simplest way to generate a key pair is to run *ssh-keygen* without arguments. In this case, it will prompt for the file in which to store keys. First,

the tool asked where to save the file. SSH keys for user authentication are usually stored in the users .ssh directory under the home directory. However, in enterprise environments, the location is often different. The default key file name depends on the algorithm, in this case *id_rsa* when using the default RSA algorithm. It could also be, for example, *id_dsa* or *id_ecdsa*.

2. Issue the command *ssh-keygen -t rsa -b 4096.* The algorithm is selected using the -t option and key size using the -b option.
3. When asked for a passphrase, just press enter. The passphrase is used for encrypting the key, so that it cannot be used even if someone obtains the private key file. The passphrase should be cryptographically strong.
4. Verify that you have created the key by issuing the command *ls -la .ssh.* The command should show the .ssh directory containing a pair of keys. For example, id_rsa.pub and id_rsa.

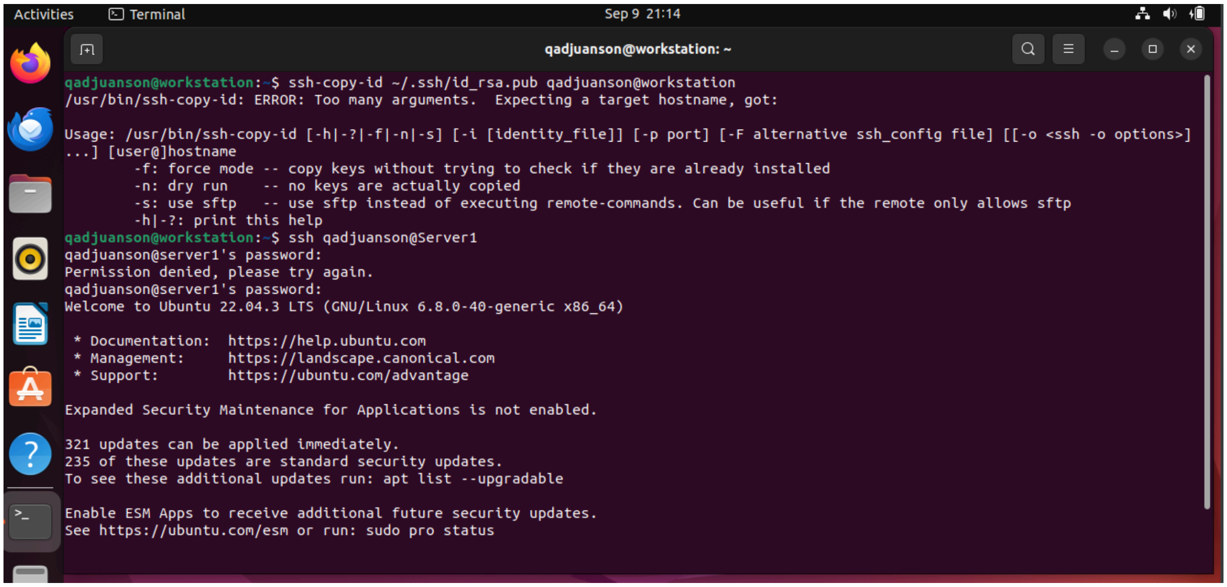

Local Machine [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
Sep 9 21:09
qadjuanson@workstation: ~

```
Your public key has been saved in /home/qadjuanson/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:wgRXurLyRcsHXjLMcG8oB8GNs42OtXHKa1Cm9OfknM4 qadjuanson@workstation
The key's randomart image is:
+---[RSA 4096]----+
|    .oo...       |
|     ++..        |
|     o=+         |
|   . BOo+        |
|  . Xo=& S       |
|   + *OoO        |
|   ...B=..       |
|    ooo=.        |
|    ...E         |
+----[SHA256]-----+
qadjuanson@workstation:~$ ls -la .ssh
total 24
drwx------  2 qadjuanson qadjuanson 4096 Sep  9 21:04 .
drwxr-x--- 15 qadjuanson qadjuanson 4096 Aug 25 12:08 ..
-rw-------  1 qadjuanson qadjuanson 3389 Sep  9 21:04 id_rsa
-rw-r--r--  1 qadjuanson qadjuanson  748 Sep  9 21:04 id_rsa.pub
-rw-------  1 qadjuanson qadjuanson 1262 Aug 25 12:34 known_hosts
-rw-r--r--  1 qadjuanson qadjuanson  142 Aug 25 12:08 known_hosts.old
qadjuanson@workstation:~$
```

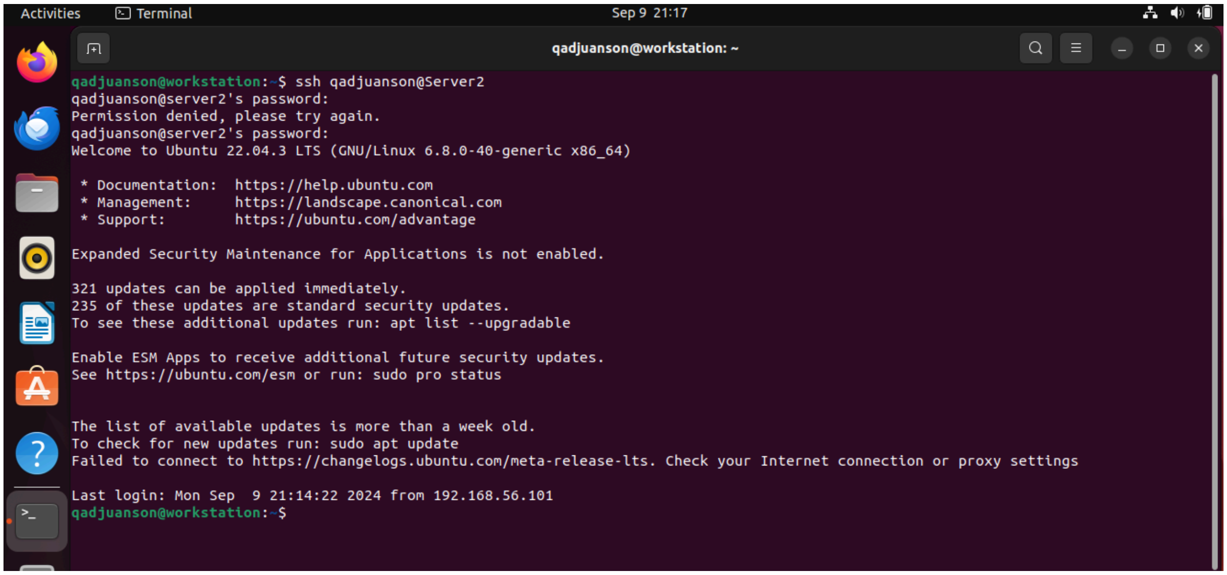
Task 2: Copying the Public Key to the remote servers

1. To use public key authentication, the public key must be copied to a server and installed in an *authorized_keys* file. This can be conveniently done using the *ssh-copy-id* tool.
2. Issue the command similar to this: *ssh-copy-id -i ~/.ssh/id_rsa user@host*
3. Once the public key has been configured on the server, the server will allow any connecting user that has the private key to log in. During the login process, the client proves possession of the private key by digitally signing the key exchange.

4. On the local machine, verify that you can SSH with Server 1 and Server 2. What did you notice? Did the connection ask for a password? If not, why?



```
qadjuanson@workstation:~$ ssh-copy-id ~/.ssh/id_rsa.pub qadjuanson@workstation
/usr/bin/ssh-copy-id: ERROR: Too many arguments.  Expecting a target hostname, got:

Usage: /usr/bin/ssh-copy-id [-h|-?|-f|-n|-s] [-i [identity_file]] [-p port] [-F alternative ssh_config file] [[-o <ssh -o options>]
...] [user@]hostname
        -f: force mode -- copy keys without trying to check if they are already installed
        -n: dry run    -- no keys are actually copied
        -s: use sftp   -- use sftp instead of executing remote-commands. Can be useful if the remote only allows sftp
        -h|-?: print this help
qadjuanson@workstation:~$ ssh qadjuanson@Server1
qadjuanson@server1's password:
Permission denied, please try again.
qadjuanson@server1's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.8.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Expanded Security Maintenance for Applications is not enabled.

321 updates can be applied immediately.
235 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```



```
qadjuanson@workstation:~$ ssh qadjuanson@Server2
qadjuanson@server2's password:
Permission denied, please try again.
qadjuanson@server2's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.8.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Expanded Security Maintenance for Applications is not enabled.

321 updates can be applied immediately.
235 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Mon Sep  9 21:14:22 2024 from 192.168.56.101
qadjuanson@workstation:~$
```

- Yes, the connection asks for a password. Based on what I've searched, it asks for a password because SSH key-based authentication is either not set up or not being used correctly.

Reflections:

Answer the following:

1. How will you describe the ssh-program? What does it do?
   - SSH (Secure Shell) is a program that allows you to securely connect to another computer over a network. Imagine you want to control a computer that is far away, but you don't want anyone else to see what you're doing. SSH helps you do that by creating a safe, encrypted link between your computer and the

remote one, so you can send commands, move files, and manage the remote system without anyone being able to intercept your actions. It's like a secure door that only you have the key to, allowing you to access and control another machine safely.

2. How do you know that you already installed the public key to the remote servers?

    - When I'm able to log into the remote server without needing to enter a password.

Part 2: Discussion

Provide screenshots for each task.

It is assumed that you are done with the last activity (**Activity 2: SSH Key-Based Authentication**).

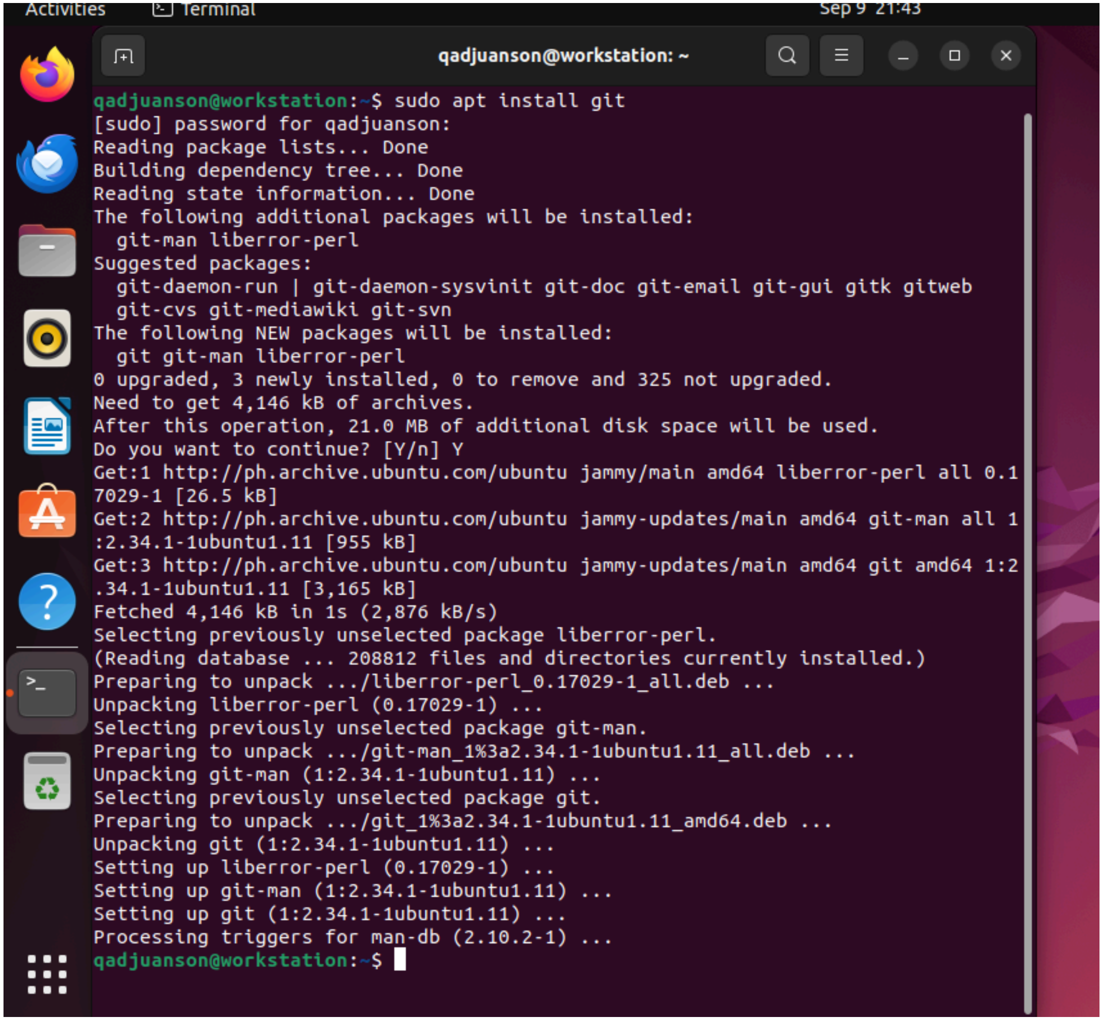
Set up Git

At the heart of GitHub is an open-source version control system (VCS) called Git. Git is responsible for everything GitHub-related that happens locally on your computer. To use Git on the command line, you'll need to download, install, and configure Git on your computer. You can also install GitHub CLI to use GitHub from the command line. If you don't need to work with files locally, GitHub lets you complete many Git-related actions directly in the browser, including:

- Creating a repository
- Forking a repository
- Managing files
- Being social

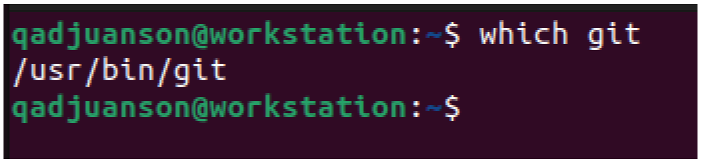
Task 3: Set up the Git Repository

1. On the local machine, verify the version of your git using the command *which git.* If a directory of git is displayed, then you don't need to install git. Otherwise, to install git, use the following command: *sudo apt install git*



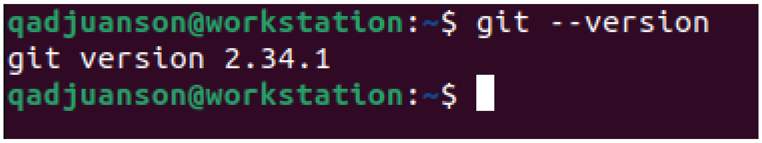
```
qadjuanson@workstation:~$ sudo apt install git
[sudo] password for qadjuanson:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb
  git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 325 not upgraded.
Need to get 4,146 kB of archives.
After this operation, 21.0 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ph.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.1
7029-1 [26.5 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1
:2.34.1-1ubuntu1.11 [955 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2
.34.1-1ubuntu1.11 [3,165 kB]
Fetched 4,146 kB in 1s (2,876 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 208812 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17029-1_all.deb ...
Unpacking liberror-perl (0.17029-1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.34.1-1ubuntu1.11_all.deb ...
Unpacking git-man (1:2.34.1-1ubuntu1.11) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.34.1-1ubuntu1.11_amd64.deb ...
Unpacking git (1:2.34.1-1ubuntu1.11) ...
Setting up liberror-perl (0.17029-1) ...
Setting up git-man (1:2.34.1-1ubuntu1.11) ...
Setting up git (1:2.34.1-1ubuntu1.11) ...
Processing triggers for man-db (2.10.2-1) ...
qadjuanson@workstation:~$ 
```

2. After the installation, issue the command *which git* again. The directory of git is usually installed in this location: *user/bin/git.*



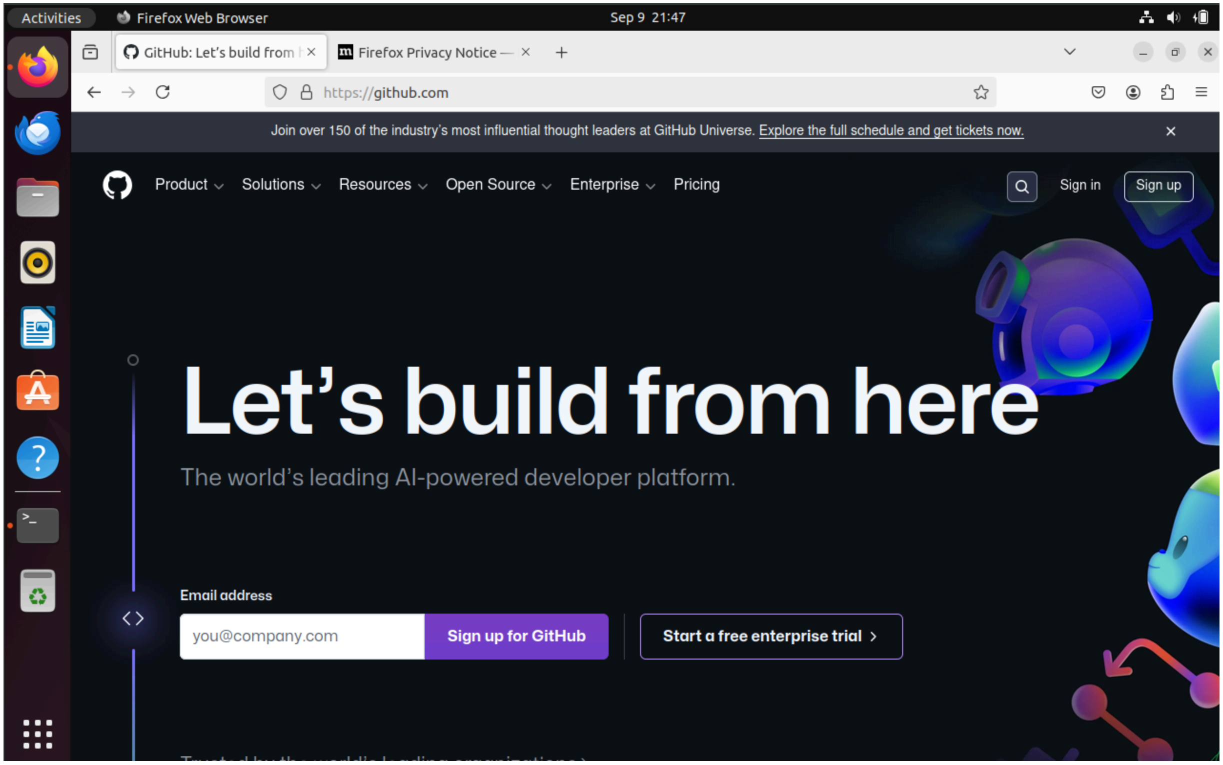
```
qadjuanson@workstation:~$ which git
/usr/bin/git
qadjuanson@workstation:~$
```

3. The version of git installed in your device is the latest. Try issuing the command *git --version* to know the version installed.



```
qadjuanson@workstation:~$ git --version
git version 2.34.1
qadjuanson@workstation:~$ 
```

4. Using the browser in the local machine, go to www.github.com.



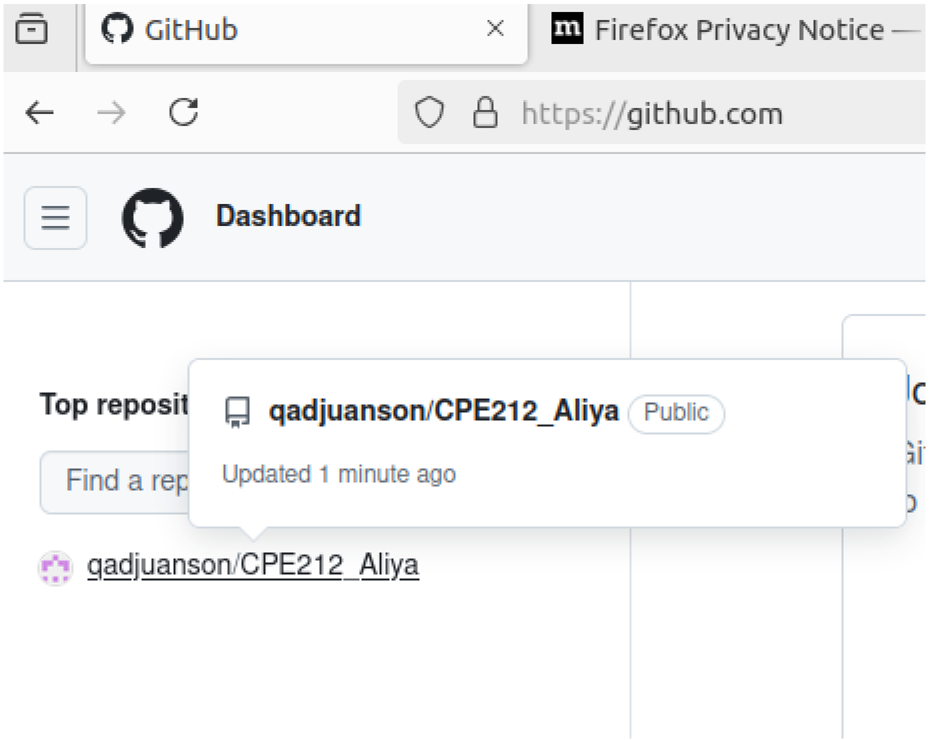
Join over 150 of the industry's most influential thought leaders at GitHub Universe. Explore the full schedule and get tickets now.

Let's build from here

The world's leading AI-powered developer platform.

Email address: you@company.com

5. Sign up in case you don't have an account yet. Otherwise, login to your GitHub account.
    a. Create a new repository and name it as CPE232_yourname. Check Add a README file and click Create repository.

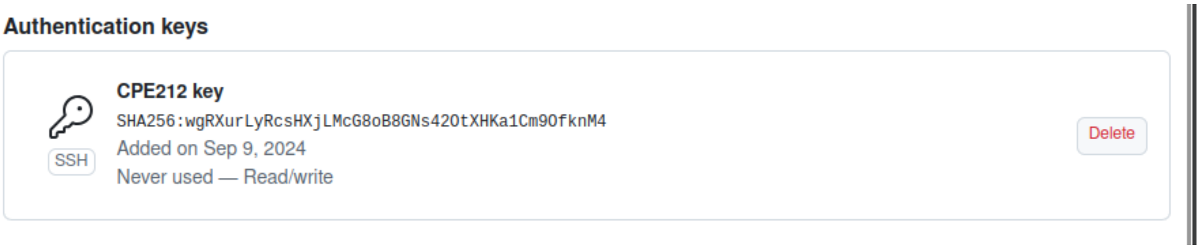


qadjuanson/CPE212_Aliya Public

Updated 1 minute ago

qadjuanson/CPE212_Aliya

b. Create a new SSH key on GitHub. Go your profile's setting and click SSH and GPG keys. If there is an existing key, make sure to delete it. To create a new SSH keys, click New SSH Key. Write CPE232 key as the title of the key.

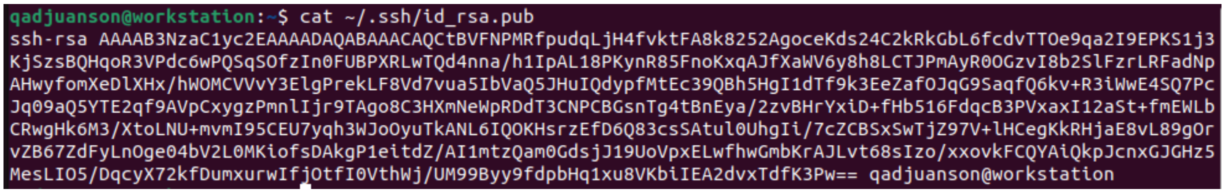


Authentication keys

SSH

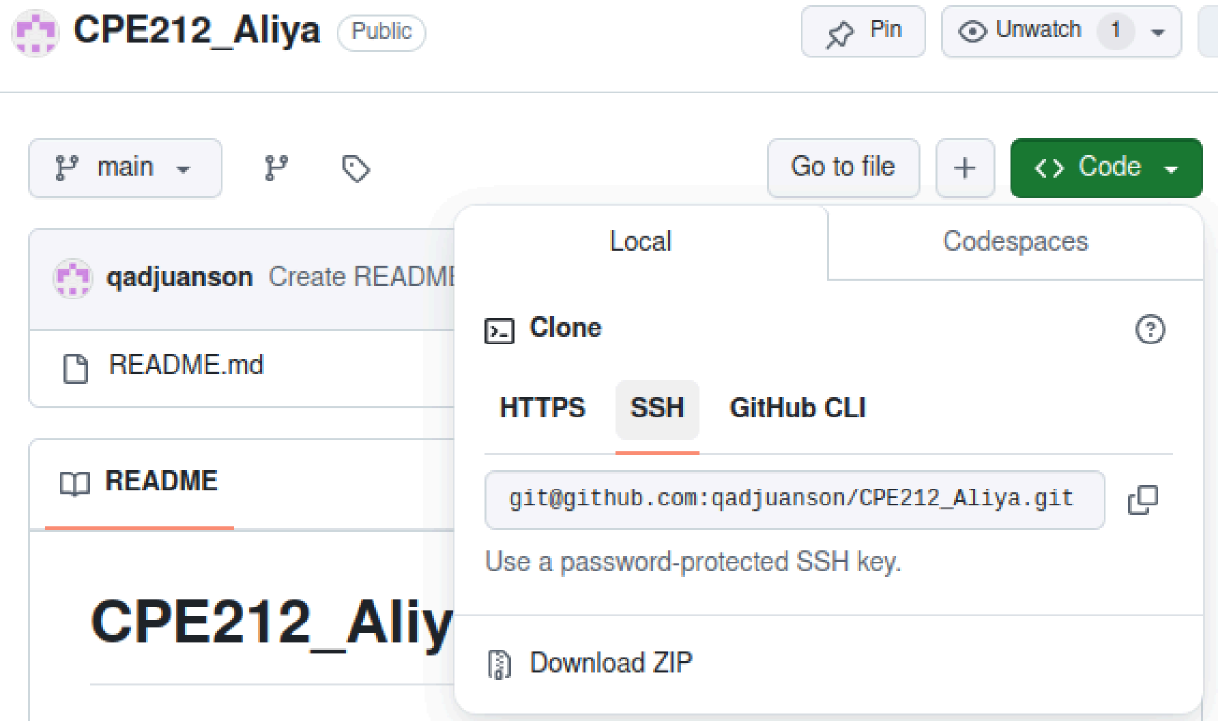
CPE212 key
SHA256:wgRXurLyRcsHXjLMcG8oB8GNs420tXHKa1Cm90fknM4
Added on Sep 9, 2024
Never used — Read/write

c. On the local machine's terminal, issue the command cat .ssh/id_rsa.pub and copy the public key. Paste it on the GitHub key and press Add SSH key.



```
qadjuanson@workstation:~$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCtBVFNPMRfpudqLjH4fvktFA8k8252AgoceKds24C2kRkGbL6fcdvTTOe9qa2I9EPKS1j3
KjSzsBQHqoR3VPdc6wPQSqSOfzIn0FUBPXRLwTQd4nna/h1IpAL18PKynR85FnoKxqAJfXaWV6y8h8LCTJPmAyR0OGzvI8b2SlFzrLRFadNp
AHwyfomXeDlXHx/hWOMCVVvY3ElgPrekLF8Vd7vua5IbVaQ5JHuIQdypfMtEc39QBh5HgI1dTf9k3EeZafOJqG9SaqfQ6kv+R3iWwE4SQ7Pc
Jq09aQ5YTE2qf9AVpCxygzPmnlIjr9TAgo8C3HXmNeWpRDdT3CNPCBGsnTg4tBnEya/2zvBHrYxiD+fHb516FdqcB3PVxaxI12aSt+fmEWLb
CRwgHk6M3/XtoLNU+mvmI95CEU7yqh3WJoOyuTkANL6IQOKHsrzEfD6Q83csSAtul0UhgIi/7cZCBSxSwTjZ97V+lHCegKkRHjaE8vL89gOr
vZB67ZdFyLnOge04bV2L0MKiofsDAkgP1eitdZ/AI1mtzQam0GdsjJ19UoVpxELwfhwGmbKrAJLvt68sIzo/xxovkFCQYAiQkpJcnxGJGHz5
MesLIO5/DqcyX72kfDumxurwIfjOtfI0VthWj/UM99Byy9fdpbHq1xu8VKbiIEA2dvxTdfK3Pw== qadjuanson@workstation
```

d. Clone the repository that you created. In doing this, you need to get the link from GitHub. Browse to your repository as shown below. Click on the Code drop down menu. Select SSH and copy the link.

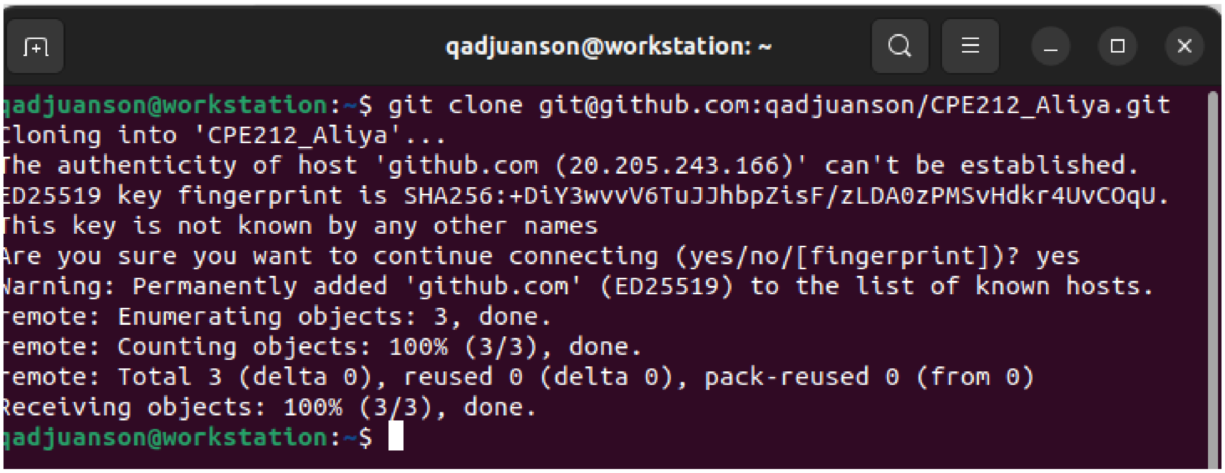


git@github.com:qadjuanson/CPE212_Aliya.git

Use a password-protected SSH key.

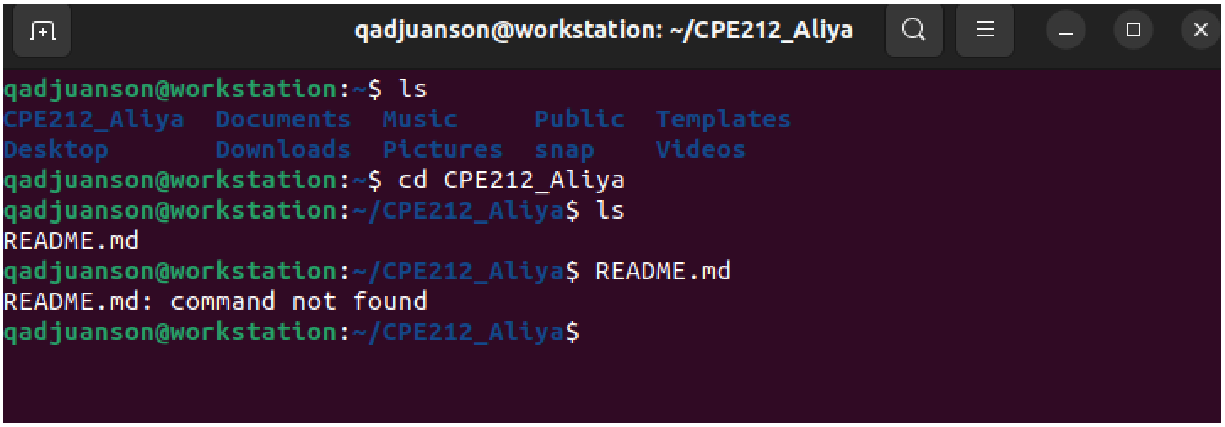
-

e. Issue the command git clone followed by the copied link. For example, *git clone git@github.com:jvtaylar-cpe/CPE232_yourname.git.* When prompted to continue connecting, type yes and press enter.



```
adjuanson@workstation:~$ git clone git@github.com:qadjuanson/CPE212_Aliya.git
loning into 'CPE212_Aliya'...
he authenticity of host 'github.com (20.205.243.166)' can't be established.
D25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
his key is not known by any other names
re you sure you want to continue connecting (yes/no/[fingerprint])? yes
arning: Permanently added 'github.com' (ED25519) to the list of known hosts.
emote: Enumerating objects: 3, done.
emote: Counting objects: 100% (3/3), done.
emote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
eceiving objects: 100% (3/3), done.
adjuanson@workstation:~$ 
```

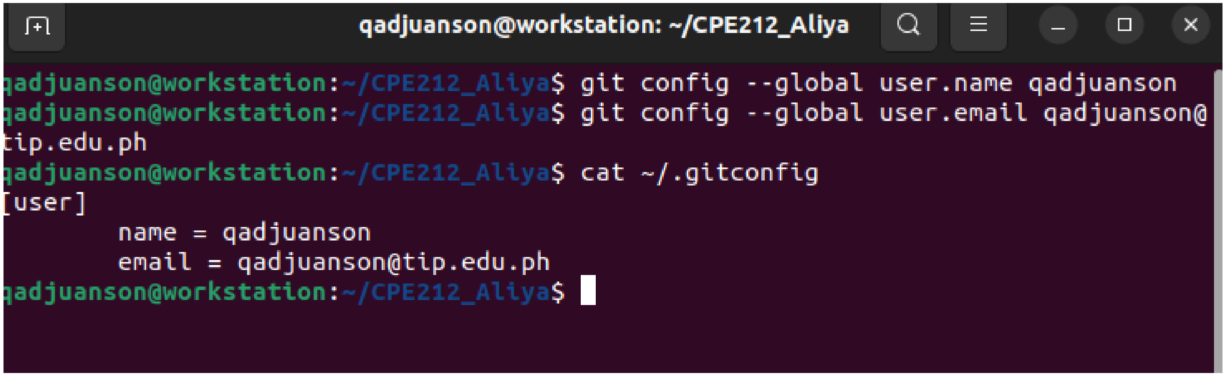
f. To verify that you have cloned the GitHub repository, issue the command *ls*. Observe that you have the CPE232_yourname in the list of your directories. Use CD command to go to that directory and LS command to see the file README.md.



```
qadjuanson@workstation:~$ ls
CPE212_Aliya  Documents  Music     Public  Templates
Desktop       Downloads  Pictures  snap    Videos
qadjuanson@workstation:~$ cd CPE212_Aliya
qadjuanson@workstation:~/CPE212_Aliya$ ls
README.md
qadjuanson@workstation:~/CPE212_Aliya$ README.md
README.md: command not found
qadjuanson@workstation:~/CPE212_Aliya$
```

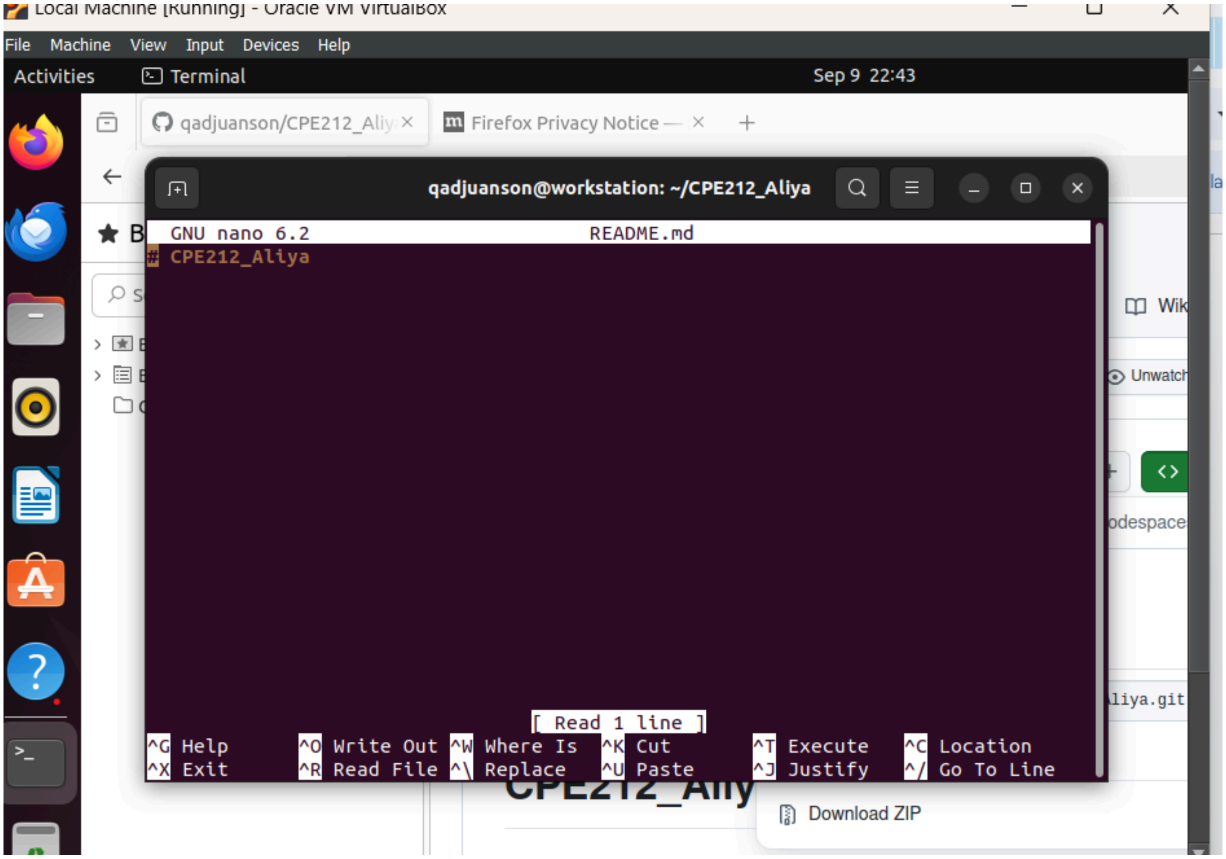
g. Use the following commands to personalize your git.

- *git config --global user.name "Your Name"*
- *git config --global user.email yourname@email.com*
- Verify that you have personalized the config file using the command *cat ~/.gitconfig*



```
qadjuanson@workstation:~/CPE212_Aliya$ git config --global user.name qadjuanson
qadjuanson@workstation:~/CPE212_Aliya$ git config --global user.email qadjuanson@
tip.edu.ph
qadjuanson@workstation:~/CPE212_Aliya$ cat ~/.gitconfig
[user]
        name = qadjuanson
        email = qadjuanson@tip.edu.ph
qadjuanson@workstation:~/CPE212_Aliya$ 
```

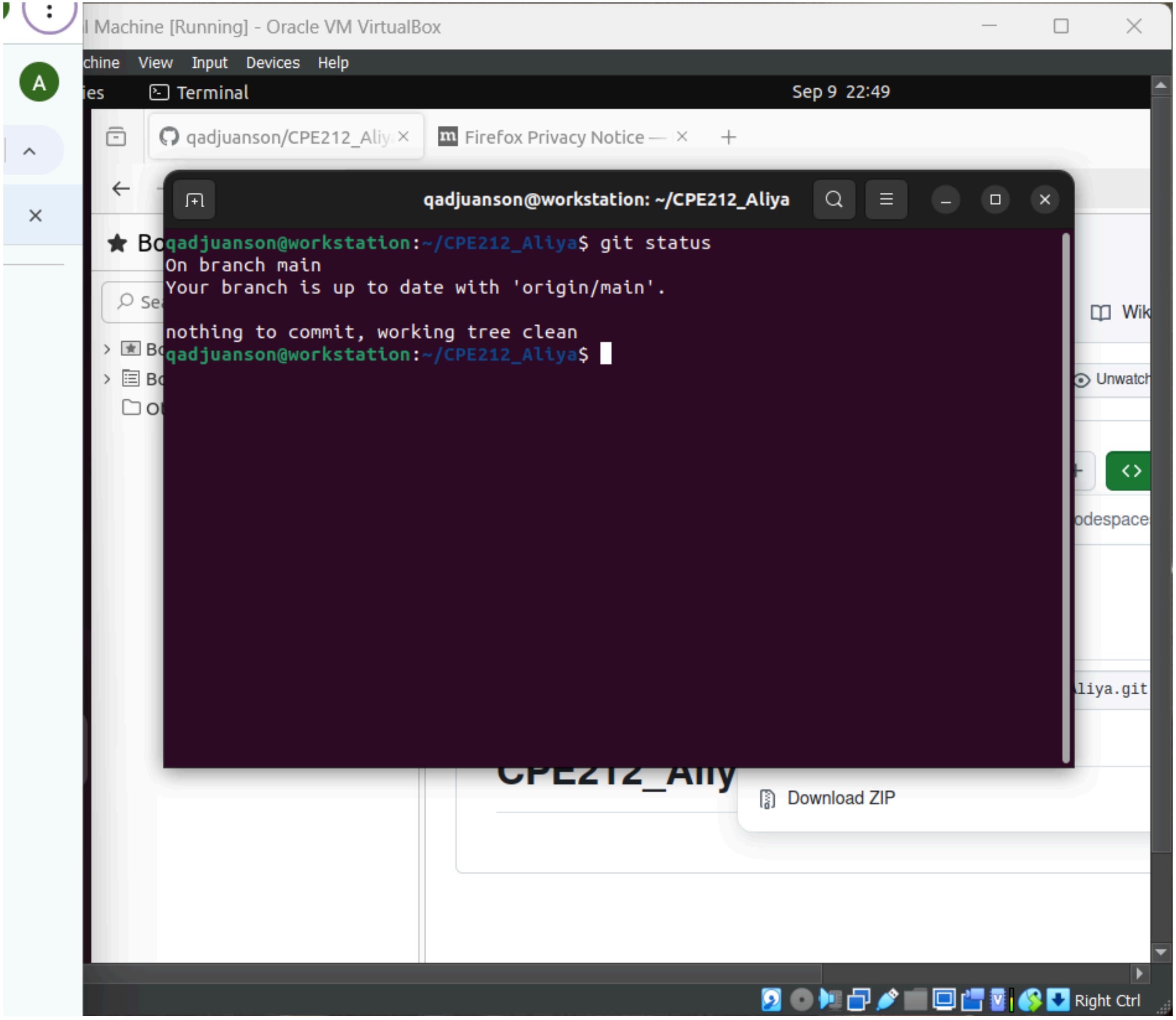
h. Edit the README.md file using nano command. Provide any information on the markdown file pertaining to the repository you created. Make sure to write out or save the file and exit.



Local Machine [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Sep 9 22:43
qadjuanson/CPE212_Aliy
Firefox Privacy Notice —
qadjuanson@workstation: ~/CPE212_Aliya
Download ZIP

```
  GNU nano 6.2                        README.md
# CPE212_Aliya

                         [ Read 1 line ]
^G Help       ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line
```

i. Use the *git status* command to display the state of the working directory and the staging area. This command shows which changes have been staged, which haven't, and which files aren't being tracked by Git. Status output does not show any information regarding the committed project history. What is the result of issuing this command?



Machine [Running] - Oracle VM VirtualBox
chine View Input Devices Help
Terminal
Sep 9 22:49
qadjuanson/CPE212_Aliy
Firefox Privacy Notice —
qadjuanson@workstation: ~/CPE212_Aliya
Download ZIP
Right Ctrl

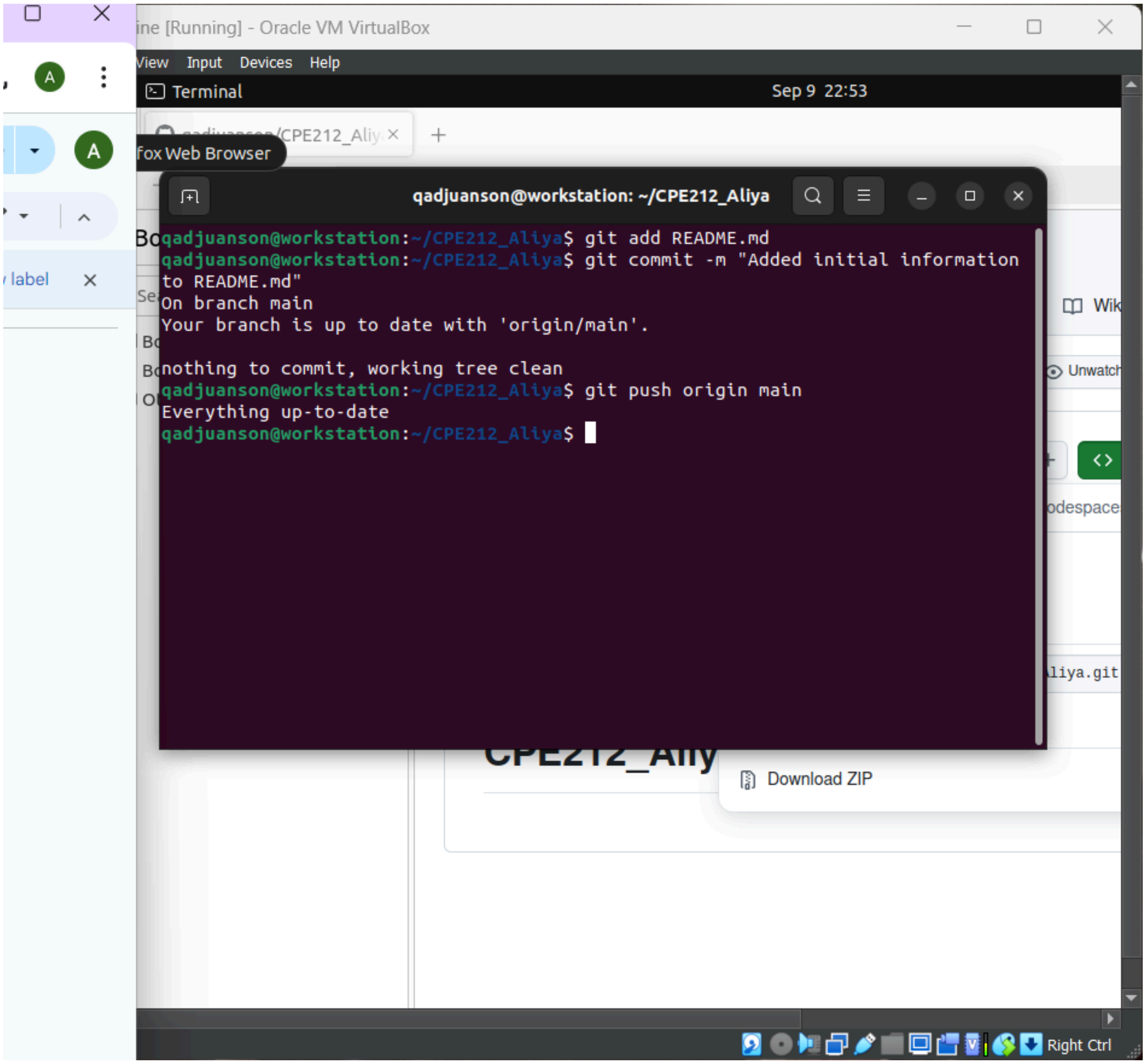
```
qadjuanson@workstation:~/CPE212_Aliya$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
qadjuanson@workstation:~/CPE212_Aliya$ 
```

j. Use the command *git add README.md* to add the file into the staging area.

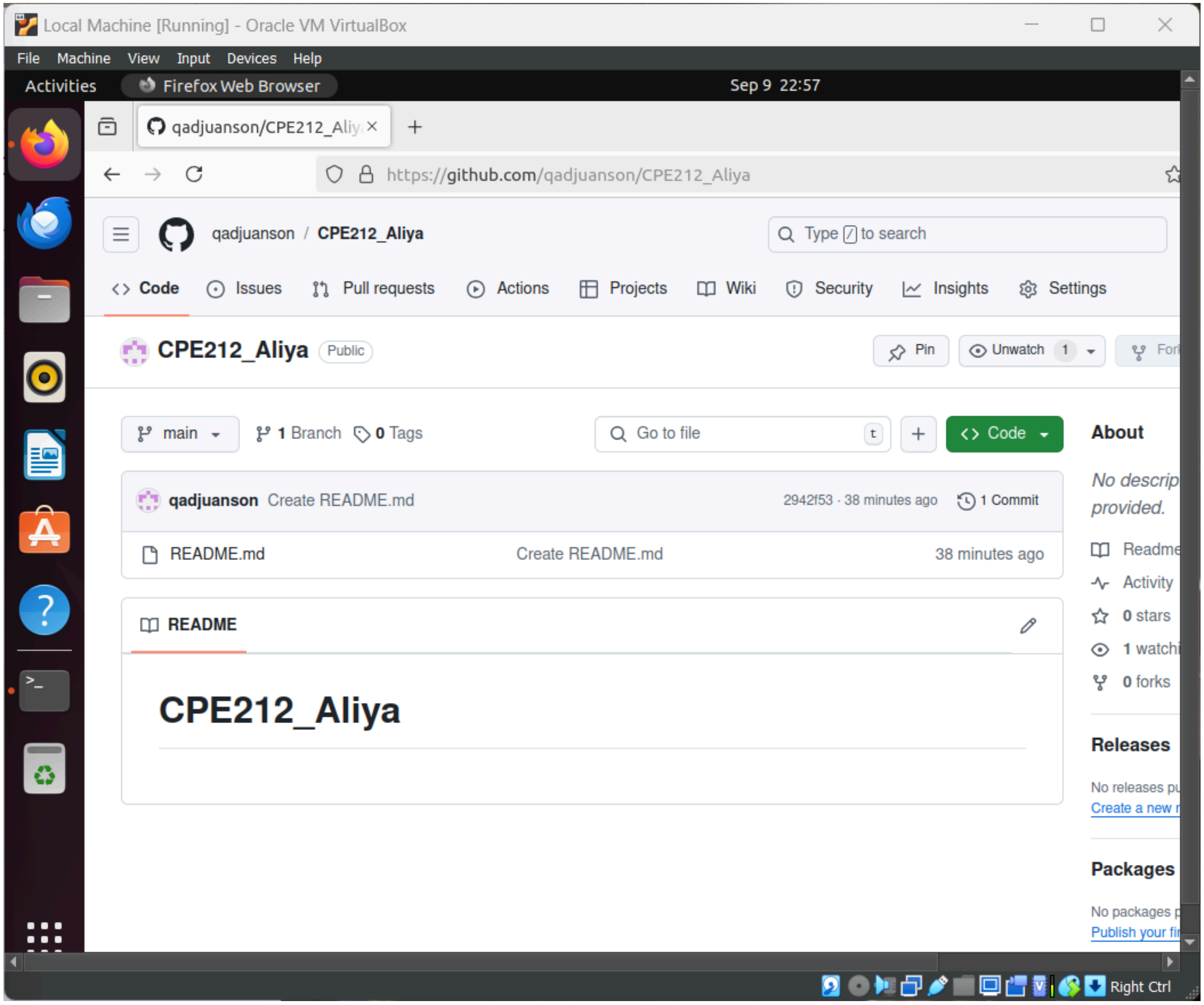
k. Use the *git commit -m "your message"* to create a snapshot of the staged changes along the timeline of the Git projects history. The use of this command is required to select the changes that will be staged for the next commit.

l. Use the command *git push <remote><branch>* to upload the local repository content to GitHub repository. Pushing means to transfer commits from the local repository to the remote repository. As an example, you may issue *git push origin main.*

m. On the GitHub repository, verify that the changes have been made to README.md by refreshing the page. Describe the README.md file. You can notice the how long was the last commit. It should be some minutes ago and the message you typed on the git commit command should be there. Also, the README.md file should have been edited according to the text you wrote.



```
qadjuanson@workstation:~/CPE212_Aliya$ git add README.md
qadjuanson@workstation:~/CPE212_Aliya$ git commit -m "Added initial information
to README.md"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
qadjuanson@workstation:~/CPE212_Aliya$ git push origin main
Everything up-to-date
qadjuanson@workstation:~/CPE212_Aliya$ 
```



Reflections:

Answer the following:

3. What sort of things have we so far done to the remote servers using ansible commands?
   - As far as I know, we haven't used any ansible commands for this activity.
4. How important is the inventory file?
   - Inventory file is important because it lists the remote servers or hosts that you want to manage. It organizes these servers/hosts into groups and can include specific connection details and variables.

Conclusions/Learnings:

- In summary, to connect local and remote machines securely using SSH keys, first create a pair of keys and set up both machines to use them instead of passwords. After confirming the connection works, set up a Git repository to link your local and remote repositories. Finally, you can run commands from your local machine to control and manage the remote servers.