

Name: Jasmin Raiza Tendencia	Date Performed: 08/22/2023
Course/Section: CPE232-CPE31S4	Date Submitted: 08/29/2023
Instructor: Dr. Jonathan Taylor	Semester and SY: 1st sem/2023-2024
Activity 2: SSH Key-Based Authentication and Setting up Git	
1. Objectives: <ul style="list-style-type: none"> 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 <p>It is assumed that you are already done with the last Activity (Activity 1: Configure Network using Virtual Machines). <i>Provide screenshots for each task.</i></p> <p>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.</p> <p>What is ssh-keygen?</p> <p>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.</p> <p>SSH Keys and Public Key Authentication</p> <p>The SSH protocol uses public key cryptography for authenticating hosts and users. The authentication keys, called SSH keys, are created using the keygen program.</p> <p>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.</p> <p>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.</p>	
Task 1: Create an SSH Key Pair for User Authentication <ul style="list-style-type: none"> 1. The simplest way to generate a key pair is to run <i>ssh-keygen</i> 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`.

```
tendencia@workstation:~$ ssh-keygen
Generating public/private rsa key pair.

Enter file in which to save the key (/home/tendencia/.ssh/id_rsa): Enter passphrase
ase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/tendencia/.ssh/id_rsa
Your public key has been saved in /home/tendencia/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:/iYxcqRNfk387M4ZyZWAdkYuNc7nzQlsJ6CULKjEKKs tendencia@workstation
The key's randomart image is:
+---[RSA 3072]---+
|      o   ooo. +      |
|    . . o ..o. X .    |
|   o . . . = % o     |
|  . . . o . B B.+    |
|.      *S  o o ++    |
|E      o.* . ..oo    |
|      o.+ .+         |
|      ... ..o        |
|      o. .+          |
+-----[SHA256]-----+
```

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.

```
tendencia@workstation:~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/tendencia/.ssh/id_rsa):
/home/tendencia/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/tendencia/.ssh/id_rsa
Your public key has been saved in /home/tendencia/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:ANajxM2xw2XXI200Rufy2iilLqe3ic5SiCtj7DGIG3o tendencia@workstation
The key's randomart image is:
+---[RSA 4096]---+
|      .o+..o.+oo      |
|      .oo=+ ..=+o      |
|      . .=. = o..      |
|      . o. .+         |
|..      . S o .        |
|= . . . .              |
|== .                   |
|**Eoo.                 |
|**==+.                 |
+-----[SHA256]-----+
```

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.

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/tendencia/.ssh/id_rsa
Your public key has been saved in /home/tendencia/.ssh/id_rsa.pub
```

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.

```
tendencia@workstation:~$ ls -la .ssh
total 24
drwx----- 2 tendencia tendencia 4096 Aug 29 03:51 .
drwxr-x--- 15 tendencia tendencia 4096 Aug 15 16:23 ..
-rw----- 1 tendencia tendencia 3389 Aug 29 03:53 id_rsa
-rw-r--r-- 1 tendencia tendencia  747 Aug 29 03:53 id_rsa.pub
-rw----- 1 tendencia tendencia 2240 Aug 15 18:40 known_hosts
-rw----- 1 tendencia tendencia 1120 Aug 15 18:31 known_hosts.old
```

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`

```
tendencia@workstation:~$ ssh-copy-id -i ~/.ssh/id_rsa tendencia@workstation
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/tendencia/.ssh/id_rsa.pub"
The authenticity of host 'workstation (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:ZmCw9iE3G13FlKE/J6IWh1XMFrS2cLXP0+Jxb4Rl1z0.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
tendencia@workstation's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'tendencia@workstation'"
and check to make sure that only the key(s) you wanted were added.
```

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.

Server 1:

```
tendencia@workstation:~$ ssh-copy-id -i ~/.ssh/id_rsa tendencia@server1
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/tendencia/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
tendencia@server1's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'tendencia@server1'"
and check to make sure that only the key(s) you wanted were added.
```

Server 2:

```
tendencia@workstation:~$ ssh-copy-id -i ~/.ssh/id_rsa tendencia@server2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/tendencia/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
tendencia@server2's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'tendencia@server2'"
and check to make sure that only the key(s) you wanted were added.
```

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?

Server 1:

```
tendencia@workstation:~$ ssh 'tendencia@server1'
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-26-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.

0 updates can be applied immediately.

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
Last login: Tue Aug 15 18:38:33 2023 from 192.168.56.101
tendencia@server1:~$
```

Server 2:

```
tendencia@workstation:~$ ssh 'tendencia@server2'
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-26-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.

0 updates can be applied immediately.

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
Last login: Tue Aug 15 18:40:18 2023 from 192.168.56.101
tendencia@server2:~$
```

Reflections:

Answer the following:

1. How will you describe the ssh-program? What does it do?
 - The SSH program, which stands for Secure Shell, is a cryptographic network protocol used for secure communication over an unsecured network. It provides a secure way to access remote servers and devices. In other words, functions as a network protocol ensuring a secure link between different hosts. In the task, we employed SSH to establish a smooth connection from a server to a workstation. This setup enabled secure access to servers (server1 and server2) directly from our workstation. One noteworthy aspect was the elimination of the need to enter a password or passphrase every time one attempts to access the host. Through this, the SSH program was comprehensively illustrated as a network protocol enabling remote interaction and management of hosts with enhanced security measures.
2. How do you know that you already installed the public key to the remote servers?
 - Through the process of verifying whether you can access the host from your workstation using the “ssh” command alongside the designated user and host details. If the command prompts successfully and does not request the password on subsequent attempts to access the server, you can confirm that the public key has been successfully installed on the remote servers.

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*

```
tendencia@workstation:~$ sudo apt install git
[sudo] password for tendencia:
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 2 not upgraded.
Need to get 4,147 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 0.17029-1 [26.5 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.9 [954 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.9 [3,166 kB]
Fetched 4,147 kB in 4s (937 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 164955 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%3a2.34.1-1ubuntu1.9_all.deb ...
Unpacking git-man (1:2.34.1-1ubuntu1.9) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.34.1-1ubuntu1.9_amd64.deb ...
Unpacking git (1:2.34.1-1ubuntu1.9) ...
Setting up liberror-perl (0.17029-1) ...
Setting up git-man (1:2.34.1-1ubuntu1.9) ...
Setting up git (1:2.34.1-1ubuntu1.9) ...
Processing triggers for man-db (2.10.2-1) ...
```

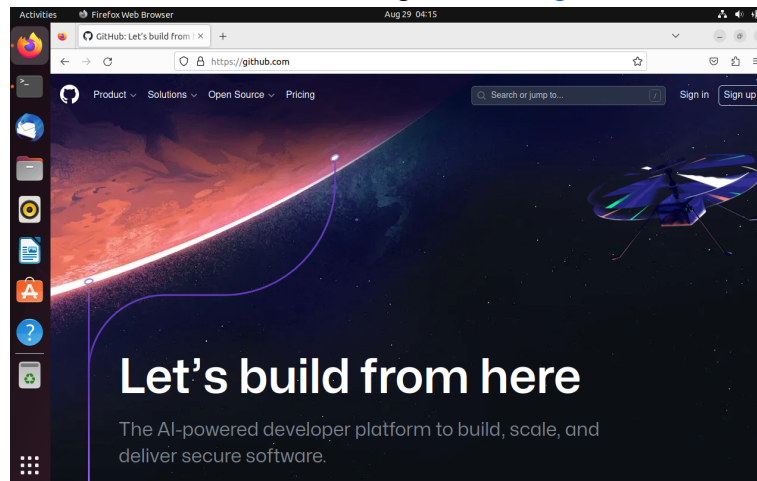
2. After the installation, issue the command *which git* again. The directory of git is usually installed in this location: *user/bin/git*.

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

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


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

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



5. Sign up in case you don't have an account yet. Otherwise, login to your GitHub account.



Sign in to GitHub

Username or email address

Password

[Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

- a. Create a new repository and name it as CPE232_yourname. Check Add a README file and click Create repository.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner *

Repository name *

jrstendencia

/

CPE232_JasminTendencia

CPE232_JasminTendencia is available.

Great repository names are short and memorable. Need inspiration? How about [jubilant-octo-fortnight](#) ?

Description (optional)

Public

Anyone on the internet can see this repository. You choose who can commit.

Private

You choose who can see and commit to this repository.

Initialize this repository with:

☒

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

☐

Add .gitignore

.gitignore template:None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

☐

Choose a license

License:None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

You are creating a public repository in your personal account.

Create repository

CPE232_JasminTendencia

Public

Pin

Unwatch 1

main

1 branch

0 tags

Go to file

Add file

<>

Code

jrstendencia

Initial commit

62cd984 now

1 commit

README.md

Initial commit

now

README.md

CPE232_JasminTendencia

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.

Settings

jrstendencia (jrstendencia)
Your personal account

Public profile
Account
Appearance
Accessibility
Notifications

Access
Billing and plans
Emails
Password and authentication
Sessions
SSH and GPG keys
Organizations
Enterprises
Moderation

Add new SSH Key

Title
CPE232

Key type
Authentication Key

Key
Begins with 'ssh-rsa', 'eddsa-sha2-nistp256', 'eddsa-sha2-nistp384', 'eddsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'

Add SSH key

- 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.

```
tendencia@workstation:~$ cat .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAQDQ/pq0VDHJQpI3JK4331aap5qGEbhkhLkdWlW+Dfv+z23aVXoyoAFMM3pfcGK
WfWs+6FZnDH/PguYwCxaIKyPIZnWgxr+FCf/H5WhfGMyVuLoJ1LHRvz5Dj9TjR25vMKnmLZZDM8PhK0uWu01cUKqs/bhqHxjfuE
5c+pqIspnID3kfpu+FDLqiraMrHXMkEncDWRHj6Zx7AWa3e0gkAP5/DNpjU9Hpvmgifslxy7X1cdAP7ysetczA0unw9VRpI/8tSv
cKKGzK30dbY1Ukxr36Wlp6+nYx1BthVEJvBqnhErBU4sCOhObtW/dEewXZ96x+F1NpjZlZaiDRgEUWt/HXRb1ePlwkD4rC7i58
uKwgXfkpEm3w/tcUAleZLmFYxz+tjdJoGbH4SONKgpZZVVHYZu8fZSrzZzo5Ag0s1T+ZvjCXv0VxsJseW/x/dloC+0+xGQMDmbPm
oHu6rZG9c75F/rS3wecdl9SxzbU9E0NSf58Gor8KU9cktxDKnKG1Sut0NkP7EedVR20woEKLwnrw/d4xRkV4YvvZo6uWssPRWP/
QXxbt7PVHL4/Mt/PB+iu8kke7YJl0Mnkp8zw1uU/Zv/31mmVrKY04e5pZenKIS4Vn/7mJysRyrpt730+zFq1pQ3sRIu+HALFyWl
hJnftiTo3KeQZoRxB0k/p0== tendencia@workstation
```

Add new SSH Key

Title
CPE232

Key type
Authentication Key

Key
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDAQDQ/pq0VDHJQpI3JK4331aap5qGEbhkhLkdWlW+Dfv+z23aVXoyoAFMM3
pfcGKwWfWs+6FZnDH/PguYwCxaIKyPIZnWgxr+FCf/H5WhfGMyVuLoJ1LHRvz5Dj9TjR25vMKnmLZZDM8PhK0uWu01cUKqs
/bhqHxjfuE5c+pqIspnID3kfpu+FDLqiraMrHXMkEncDWRHj6Zx7AWa3e0gkAP5
/DNpjU9Hpvmgifslxy7X1cdAP7ysetczA0unw9VRpI
/8tSvccKKGzK30dbY1Ukxr36Wlp6+nYx1BthVEJvBqnhErBU4sCOhObtW/dEewXZ96x+F1NpjZlZaiDRgEUWt
/HXRb1ePlwkD4rC7i58uKwgXfkpEm3w
/tcUAleZLmFYxz+tjdJoGbH4SONKgpZZVVHYZu8fZSrzZzo5Ag0s1T+ZvjCXv0VxsJseW

Add SSH key

SSH keys

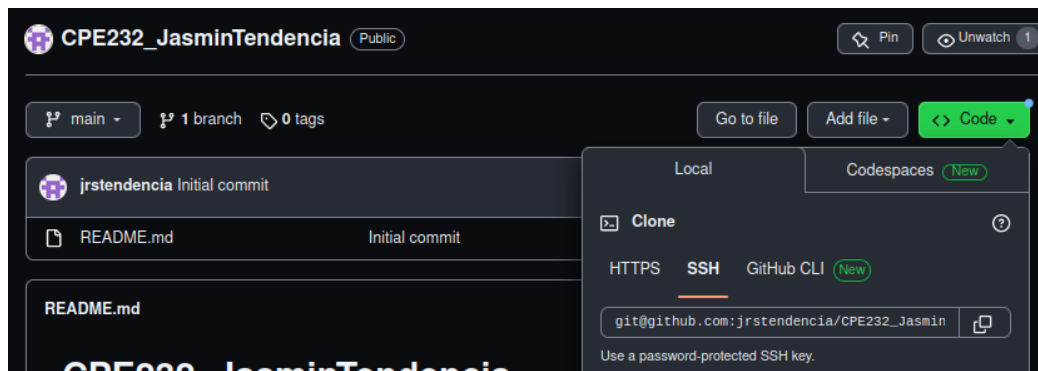
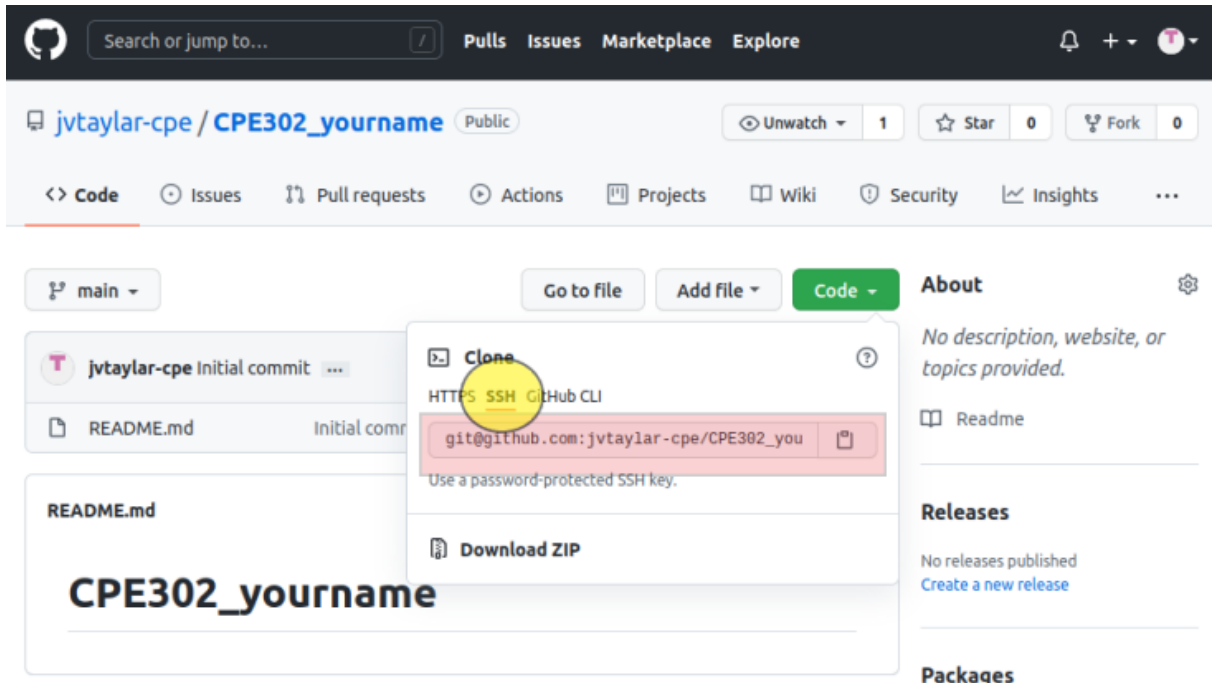
New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication Keys

<p>CPE232 SHA256:ANajxM2xw2XXI200Rufy2iiilQe3ic5SiCtj706IG3o Added on Aug 29, 2023 Never used — Read/write</p>	Delete
---	--------

- 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.



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

```
tendencia@workstation:~$ git clone git@github.com:jrstendencia/CPE232_JasminTendencia.git
Cloning into 'CPE232_JasminTendencia'...
The authenticity of host 'github.com (192.30.255.113)' can't be established.
ED25519 key fingerprint is SHA256:+DIY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

- 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.

```
tendencia@workstation:~$ ls
CPE232_JasminTendencia  Documents  Music      Public  Templates
Desktop                 Downloads  Pictures   snap    Videos
tendencia@workstation:~$ cd CPE232_JasminTendencia
tendencia@workstation:~/CPE232_JasminTendencia$ ls
README.md
```

- g. Use the following commands to personalize your git.

- `git config --global user.name "Your Name"`
- `git config --global user.email yourname@email.com`

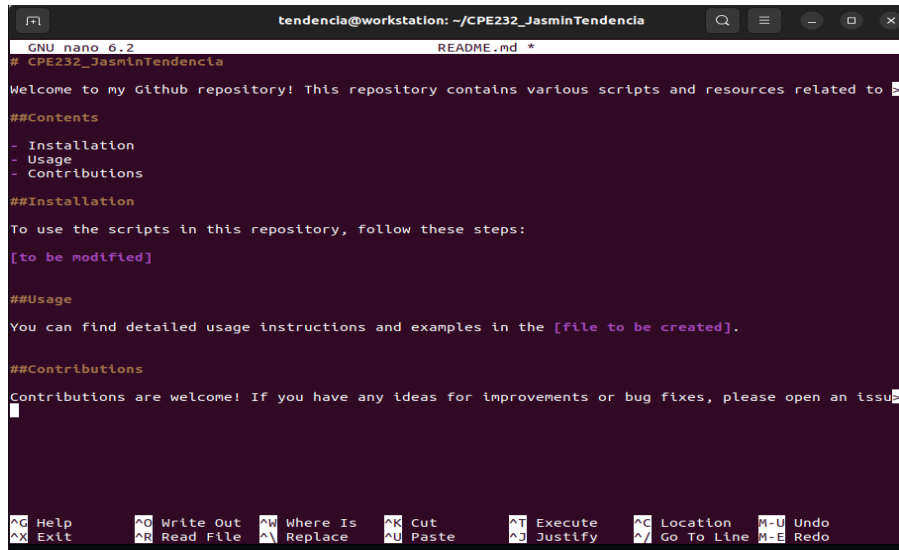
```
tendencia@workstation:~/CPE232_JasminTendencia$ git config --global user.name "Jasmin Tendencia"
tendencia@workstation:~/CPE232_JasminTendencia$ git config --global user.email qjrtendencia@tip.edu.ph
tendencia@workstation:~/CPE232_JasminTendencia$
```

- Verify that you have personalized the config file using the command `cat ~/.gitconfig`

```
tendencia@workstation:~/CPE232_JasminTendencia$ cat ~/.gitconfig
[user]
    name = Jasmin Tendencia
    email = qjrtendencia@tip.edu.ph
tendencia@workstation:~/CPE232_JasminTendencia$
```

- 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.

```
tendencia@workstation:~/CPE232_JasminTendencia$ sudo nano README.md
[sudo] password for tendencia:
```



The screenshot shows the nano text editor interface. The title bar indicates the file is 'README.md' in the directory '~/CPE232_JasminTendencia'. The editor content includes a welcome message, sections for Contents, Installation, Usage, and Contributions, and a footer with navigation shortcuts. The cursor is at the end of the Contributions section.

```
GNU nano 6.2 README.md *
# CPE232_JasminTendencia

Welcome to my Github repository! This repository contains various scripts and resources related to

##Contents
- Installation
- Usage
- Contributions

##Installation
To use the scripts in this repository, follow these steps:
[to be modified]

##Usage
You can find detailed usage instructions and examples in the [file to be created].

##Contributions
Contributions are welcome! If you have any ideas for improvements or bug fixes, please open an issue
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location ^M-U Undo
^K Exit      ^R Read File ^\ Replace   ^U Paste     ^D Justify  ^_ Go To Line ^M-E Redo
```

- 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?

```
tendencia@workstation:~/CPE232_JasminTendencia$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
tendencia@workstation:~/CPE232_JasminTendencia$
```

Upon issuing the command, it provides an overview of the local repository's status, where it displays the current branch name, lists of modified files not yet staged, shows staged files for the next commit, identifies untracked files, and offers commit information if not pushed.

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

```
tendencia@workstation:~/CPE232_JasminTendencia$ git add README.md
tendencia@workstation:~/CPE232_JasminTendencia$
```

- 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.

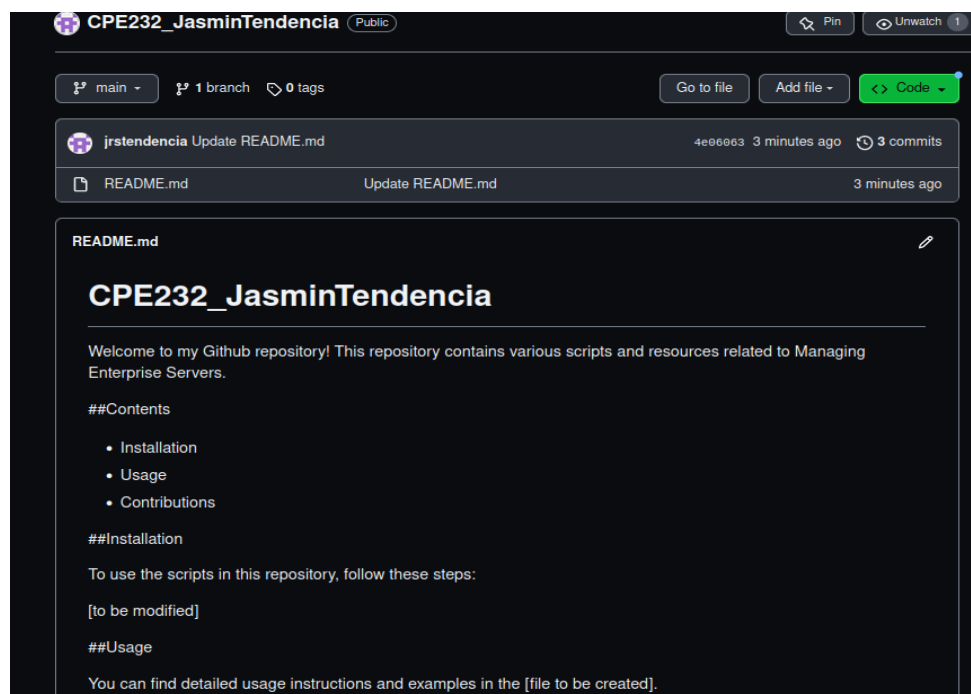
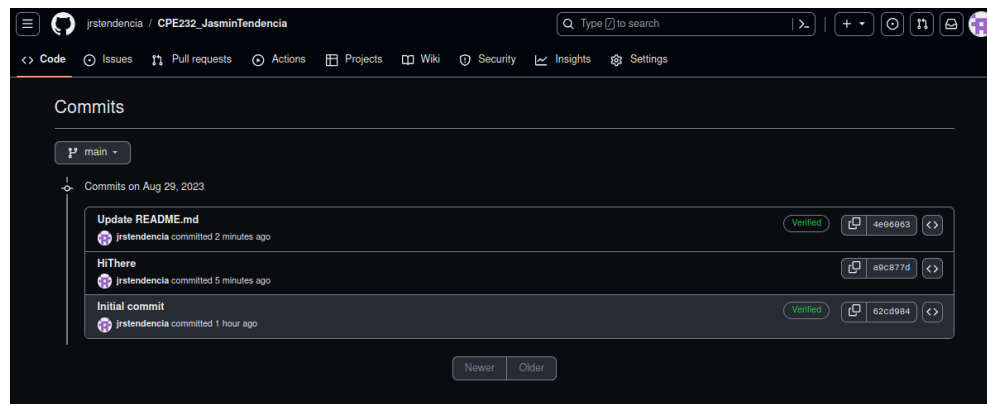
```
tendencia@workstation:~/CPE232_JasminTendencia$ git commit -m "HiThere"
[main a9c877d] HiThere
 1 file changed, 26 insertions(+), 1 deletion(-)
 rewrite README.md (100%)
```

- 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*.

```
tendencia@workstation:~/CPE232_JasminTendencia$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 569 bytes | 94.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:jrstendencia/CPE232_JasminTendencia.git
 62cd984..a9c877d  main -> main
tendencia@workstation:~/CPE232_JasminTendencia$
```

- 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.



The status indicates “3 commits” where the commitment was done 3 minutes ago.

Reflections:

Answer the following:

3. What sort of things have we so far done to the remote servers using ansible commands?
 - Up to this point in the task, we have established a connection between the workstation and the server. Additionally, we have set up a GitHub account and linked it to the workstation, this enables us to modify its content using commands.
4. How important is the inventory file?

- The inventory file in Ansible is crucial as it defines the list of remote hosts that Ansible will manage. This file specifies the connection information for remote servers, such as their IP addresses or hostnames, SSH usernames, SSH keys, and other connection-related parameters. The inventory file organizes hosts into groups, making it easier to target specific groups of servers with Ansible playbooks and tasks.

Conclusions/Learnings:

During the course of engaging with the task, I successfully created a secure and convenient SSH connection for both the server and host, ensuring seamless logins. By generating SSH keys, it ensured an encrypted and secure channel for communication, enhancing overall security. There is also verification of connectivity assuring the smooth operation of the established SSH connection and validating the correct implementation of SSH keys. Moreover, the activity covered the basics of establishing a Git repository, where we delved into the fundamentals of version control, paving the way for efficient tracking of changes which will prove valuable for upcoming tasks in this course.

In summary, this activity effectively showcased the process of establishing an SSH connection, highlighting the essential elements and terms involved in forming both the SSH connection and the Git repository which enhanced efficiency and security in the realm of remote server management and version control.