Connecting to cluster:

First we need to generate another key pair for the cluster in our PowerShell satya folder… (the key pair will save In .ssh folder)…..we give the name as cluster key…

So, the key generated and saved in our .ssh folder. And gave the public key to prof in the submission.

A screenshot of a computer

Description automatically generated

We downloaded and installed the cisco AnyConnect from the link provided by the professor.

A close-up of a text

Description automatically generated

A computer screen with a lock and text

Description automatically generated

First we will get blank…. leave it like that

Now next week:

We are connecting to the cluster with our private key:

1st we need to enter the vpn.iit.edu in that blank space and click connect and it willa ks iit password…enter it…it will ask okta mfa…click on your mobile yes…

So we get connected tovpn

Now open the spark cluster page rice.iit.edu

A screenshot of a computer

Description automatically generated

This page will only open if we get connected to that cisco vpn. Or else this page will not get connected.

Now go to our terminal and do this:

A screenshot of a computer

Description automatically generated

Just to test whether out vpn connection got connected or not…for future reference ..this is not a mandatory step.

Now we take our cluster private key store in our local computer satya .ssh folder…copy that key as path and paste it in the command in that blue part. Then our hawk id(just id not with @hawk.iit.edu) and the remaining part In that command

A screenshot of a computer

Description automatically generated

Type who to check who else got logged in

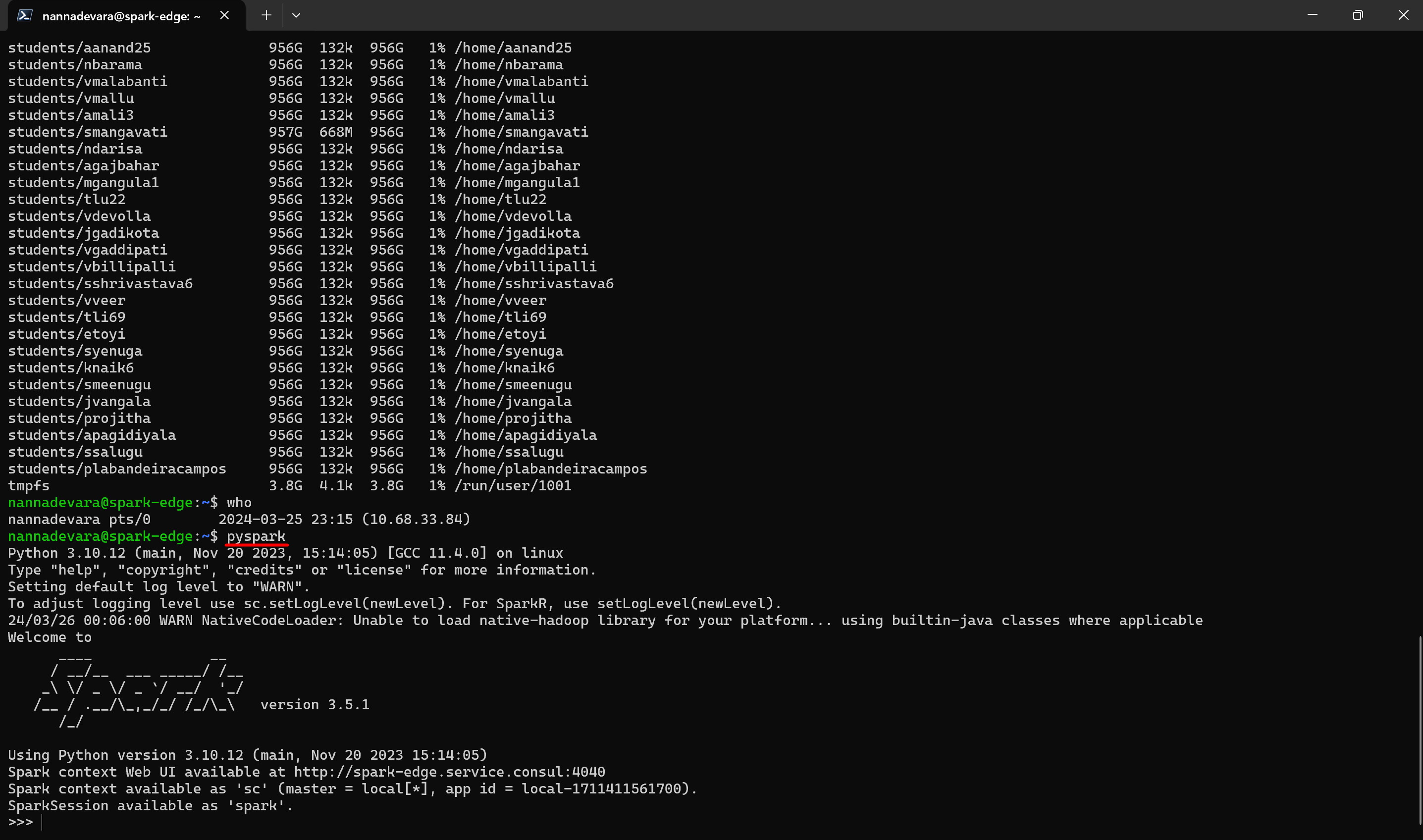
A screenshot of a computer

Description automatically generated

Type pyspark and check if you are getting this or not..im getting the spark shell..

To exit the spark shell type:

**exit()**



**NOW WE NEED TO REPEAT THE GIT TUTORIAL STEPS IN THIS CLUSTE ALSO**:

we need to config username and password,

Generate a new key pair

Create config file

Put the public key key on git hub

Clone our repo here.

**Config username and email:**

**git config --global user.name "Naga Satya Shilpa"**

**git config --global user.email “nannadevara@hawk.iit.edu”**

to check whether thee are added or not..type **git config --list**

A screenshot of a computer

Description automatically generated

**create key pair for authentication:**

**ssh-keygen -t ed25519**

A screen shot of a computer

Description automatically generated

The file is **/home/vagrant/.ssh/id\_ed25519 ---🡪 the old path from the old documentation..delete this line here**

**/home/nannadevara/.ssh/id\_ed25519**

Then match with the path and enter your wish file name as follows: copy the above path first and then extend the path with whatever file name suitable for this subject: (path\_github\_itmdclustervm521\_key)

So enter this here:

**/home/nannadevara/.ssh/id\_ed25519\_github\_itmdclustervm521\_key**

A black screen with white text

Description automatically generated

**Your identification has been saved in /home/nannadevara/.ssh/id\_ed25519\_github\_itmdclustervm521\_key**

**(private key)**

**Your public key has been saved in /home/nannadevara/.ssh/id\_ed25519\_github\_itmdclustervm521\_key.pub**

**(public key)**

Now display the public key:

**cat** **/home/vagrant/.ssh/id\_ed25519\_github\_itmdvm521\_key.pub**

A screenshot of a computer

Description automatically generated

**The public key is:**

**ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPfSZW/7YbuKxuySDmylge2NpshfC36F4T6YuygYqXQr nannadevara@spark-edge**

and then added this public key in github -setting – ssh gpg keys – add ssh key – paste this key – add appropriate title and save. (title could be itmdvm)

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**lets create config file**: Every user have .ssh directory under home directory

What is home directory? When you open ‘terminal’ – powershell 7 – whatever shows initially, it is our home directory. For instance mine is : vagrant@anss-521

To check whether .ssh directory is there by default or not, type **ls -a or ls -al** in home directory (vagrant@anss-521) (in bash shell ubuntu) and check like this

(note that .ssh the dot is because it is hidden file,in windows hidden files will be shown with ls command but in VirtualBox ubuntu hidden files willnot be shown with ls command. that’s why ls -al)

A screenshot of a computer

Description automatically generated

**Note the above are diff commands to show the list in that directory in different ways.**

This ssh directory only shows up only if you did that ssh keygen command earlier. Or else it wont show up. That command forces you to create this ssh directory.

Lets go to ssh directoty by using cd command:

**cd .ssh**

Now we are in ssh directory.

Type ls

A screen shot of a computer

Description automatically generated

These are our pub and private keys

Now **create a file called config** : (prof already have config file so he just did cat config, tio show us how it looks like. So when he did, he got a screen saying that when we conncect to host ~ github.com use these values. user is the github ID not the repository(whatever ID we submitted to professor), Identity file is the path to the private key (look at th ssh keygen screenshot to get this path) , so that when we connect to github , it will present the key to authenticate, and authenticate us without we typing any passwords ) A screen shot of a computer

Description automatically generated

But for us first we need to create config file:

As we are in virtualbox, we don’t have code(vscode texteditor top create and open the config file)

So we need to vi

**vi config**

Host github.com

User Shilpa206

Hostname github.com

IdentityFile **/home/nannadevara/.ssh/id\_ed25519\_github\_itmdclustervm521\_key**

**To type the content in this config file: type esc, i..(then it will go into insert mode) and press ctrl+shift+v to paste the content above and then press esc again…then press shift+colon and enter wq ..and press enter**

A screenshot of a computer

Description automatically generated

A computer screen with white text

Description automatically generated

Now lets clone it

A black screen with white text

Description automatically generated

A screenshot of a computer program

Description automatically generated