

**Department of Computer Engineering**

**Academic Term: Jan-May 2022**

**Class: BE COMPUTERS**

**Subject Name: CLOUD COMPUTING LAB**

**Subject Code: CSL803**

<b>Title:</b>	<b>Hosting website using EC2 and S3</b>
<b>Date of Performance:</b>	<b>20/04/22</b>
<b>Date of Submission:</b>	<b>23/04/2022</b>
<b>Roll No:</b>	<b>8626, 8632, 8639</b>
<b>Name of the Student:</b>	<b>Divita Phadakale, Praditi Rede, Sheetal Sharma</b>

**Evaluation:**

<b>Sr. No</b>	<b>Rubric</b>	<b>Grade</b>
<b>1</b>	<b>On time submission(2)</b>	
<b>2</b>	<b>Preparedness(2)</b>	
<b>3</b>	<b>Output(2)</b>	
<b>4</b>	<b>Post Lab Questions (4)</b>	
	<b>TOTAL</b>	

**Signature of the Teacher:**

# Title: Hosting a website using EC2 and S3

## Table of Contents

<b>Abstract:</b> .....	<b>2</b>
<b>Features:</b> .....	<b>3</b>
<b>Technology:</b> .....	<b>3</b>
<b>Stepwise tutorial with screenshot for the project done.</b> .....	<b>4</b>
Step 1: Create a S3 Bucket with a unique name and add your files to it. We can add our css, html, js images to it. ....	4
Step 2: Create an IAM Role. An IAM Role is needed to access our S3 Stored files through EC2 instance...	4
Step 3: Create a security group A security group acts as a firewall to our EC2 instance. ....	5
Step 4: We need to create a key pair. A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance. Amazon EC2 stores the public key, and you store the private key .....	6
Step 5: Now we need to create an ec2 instance .....	7
Step 6: Once the instance is launched and running just copy your Public IP address. ....	9
Step 7: Next, we need to ssh into our ec2 instance through our terminal. For that open your cmd. And navigate to your downloads folder where our pem file is downloaded. And type the command <code>ssh -i securitykeypair.pem ec2-user@(public ip address)</code> .....	10
Step 8: Now we will be into our ec2 instance.to navigate to root user type the command <code>sudo su</code> .....	10
Step 9: We can check whether the server is installed or not using <code>chkconfig httpd on</code> .....	11
Step 10: Next we can <code>ls</code> and check if our files are uploaded onto the directory.....	11
Step 11: After this we can navigate to our website using the EC2 public ip we can see our website is hosted .....	12
<b>Conclusion:</b> .....	<b>12</b>
<b>References:</b> .....	<b>13</b>

## Abstract:

Amazon EC2 helps you in making resizable memory and compute capacity online (on cloud). Hosting on aws gives you an awesome plug and play option for your server hard disks, capacity and computing power. Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the AWS Cloud. In other words, it provides us with a virtual server with different capacities to best suit your needs. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in traffic with ease An on-demand EC2 instance from AWS is one in which the user can rent the virtual server per hour and use it to deploy his/her own applications.

## Features:

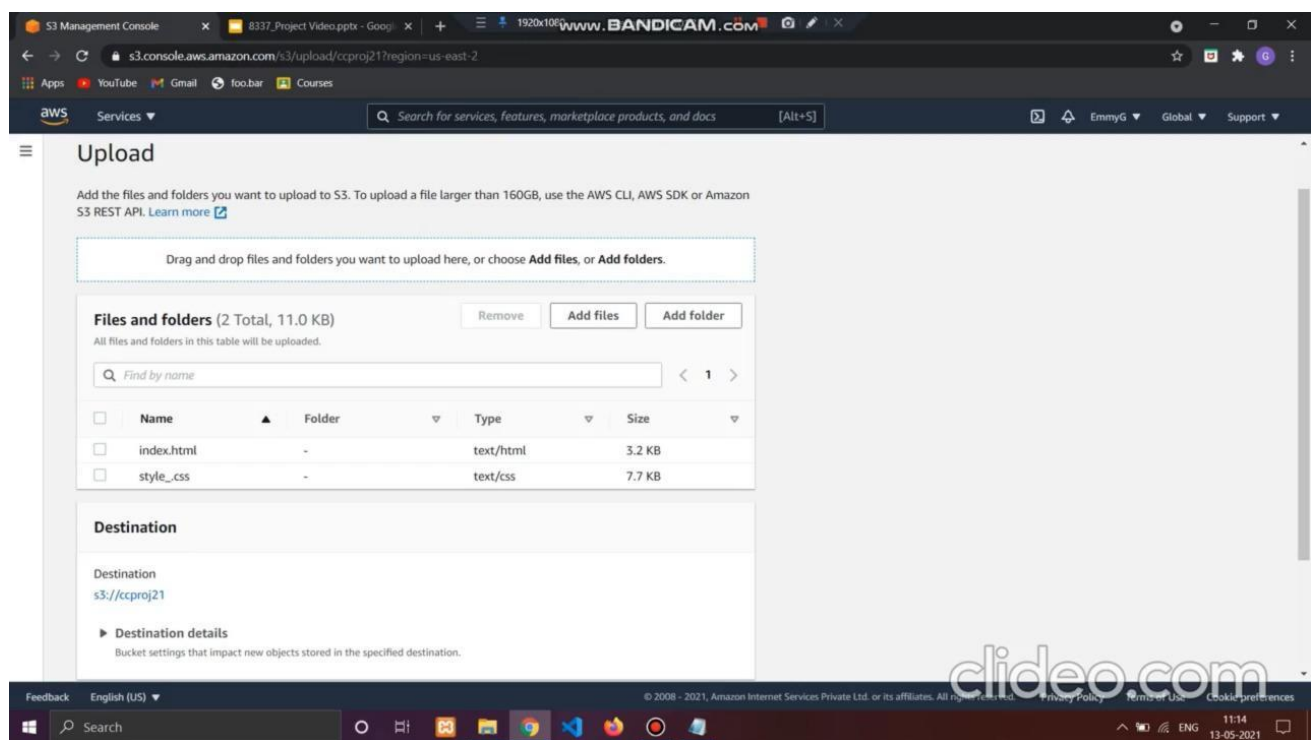
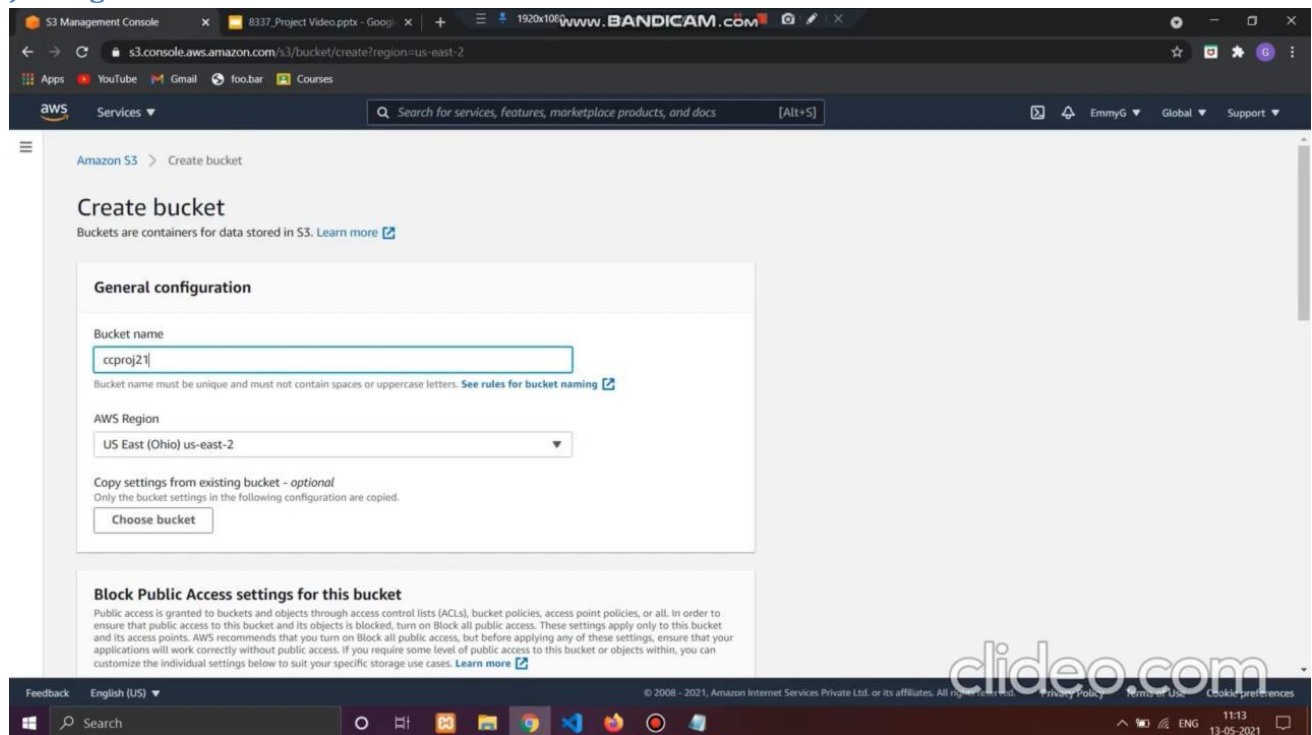
Amazon EC2 allows you to set up and configure everything about your instances from your operating system up to your applications. An Amazon Machine Image (AMI) is simply a packaged-up environment that includes all the necessary bits to set up and boot your instance. Your AMIs are your unit of deployment. You might have just one AMI or you might compose your system out of several building block AMIs (e.g., web servers, app servers, and databases). Amazon EC2 provides several tools to make creating an AMI easy including the Amazon Web Services Management Console. With S3 bucket names, prefixes, object tags, and S3 Inventory, you have a range of ways to categorize and report on your data, and subsequently can configure other S3 features to take action. S3 Batch Operations makes it simple, whether you store thousands of objects or a billion, to manage your data in Amazon S3 at any scale.

## Technology:

The technology used are AWS particularly EC2, S3, IAM Role and Security Group. Amazon EC2 presents a true virtual computing environment, allowing you to use web service interfaces to launch instances with a variety of operating systems, load them with your custom application environment, manage your network's access permissions, and run your image using as many or few systems as you desire. Amazon S3 has various features you can use to organize and manage your data in ways that support specific use cases, enable cost efficiencies, enforce security, and meet compliance requirements. Data is stored as objects within resources called "buckets", and a single object can be up to 5 terabytes in size. S3 features include capabilities to append metadata tags to objects, move and store data across the S3 Storage Classes, configure and enforce data access controls, secure data against unauthorized users, run big data analytics, and monitor data at the object, bucket levels, and view storage usage and activity trends across your organization. Objects can be accessed through S3 Access Points or directly through the bucket hostname.

## Stepwise tutorial with screenshot for the project done.

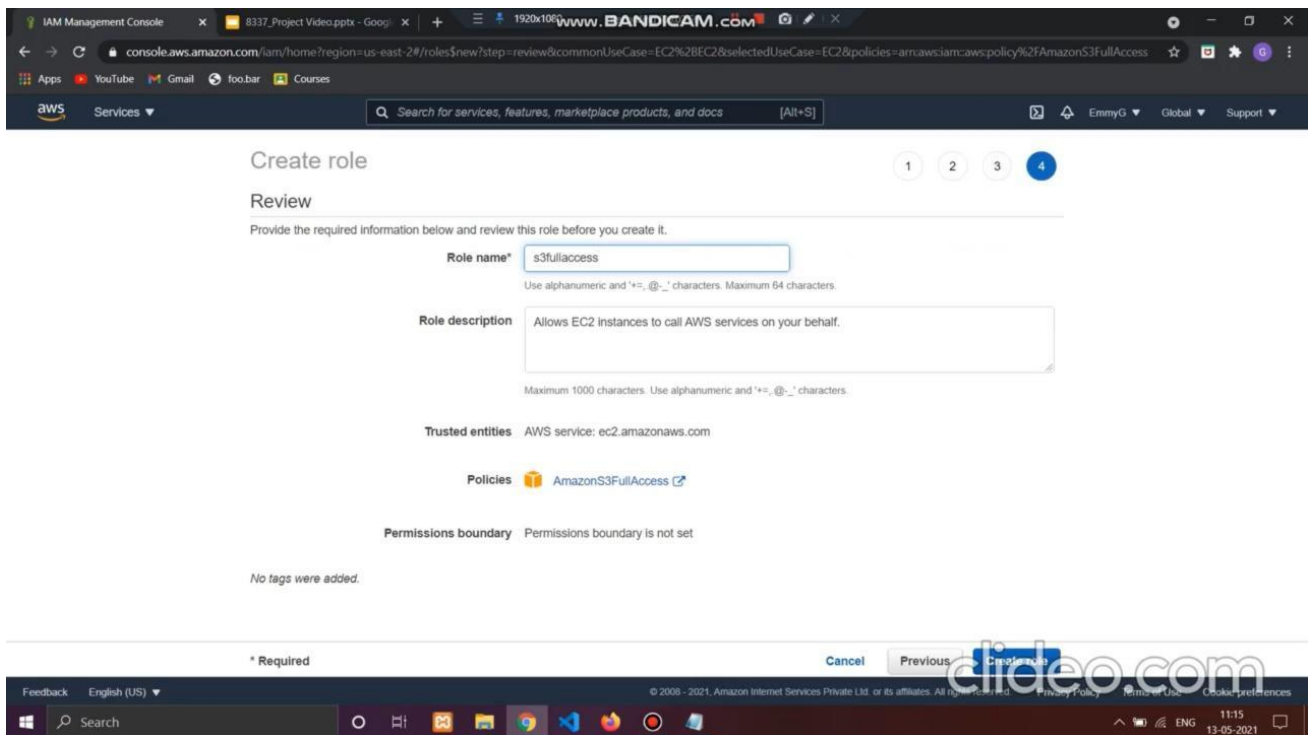
**Step 1: Create a S3 Bucket with a unique name and add your files to it. We can add our css, html, js images to it.**



**Step 2: Create an IAM Role. An IAM Role is needed to access our S3 Stored files through EC2 instance**

**2.1 Go to IAM, then Roles**

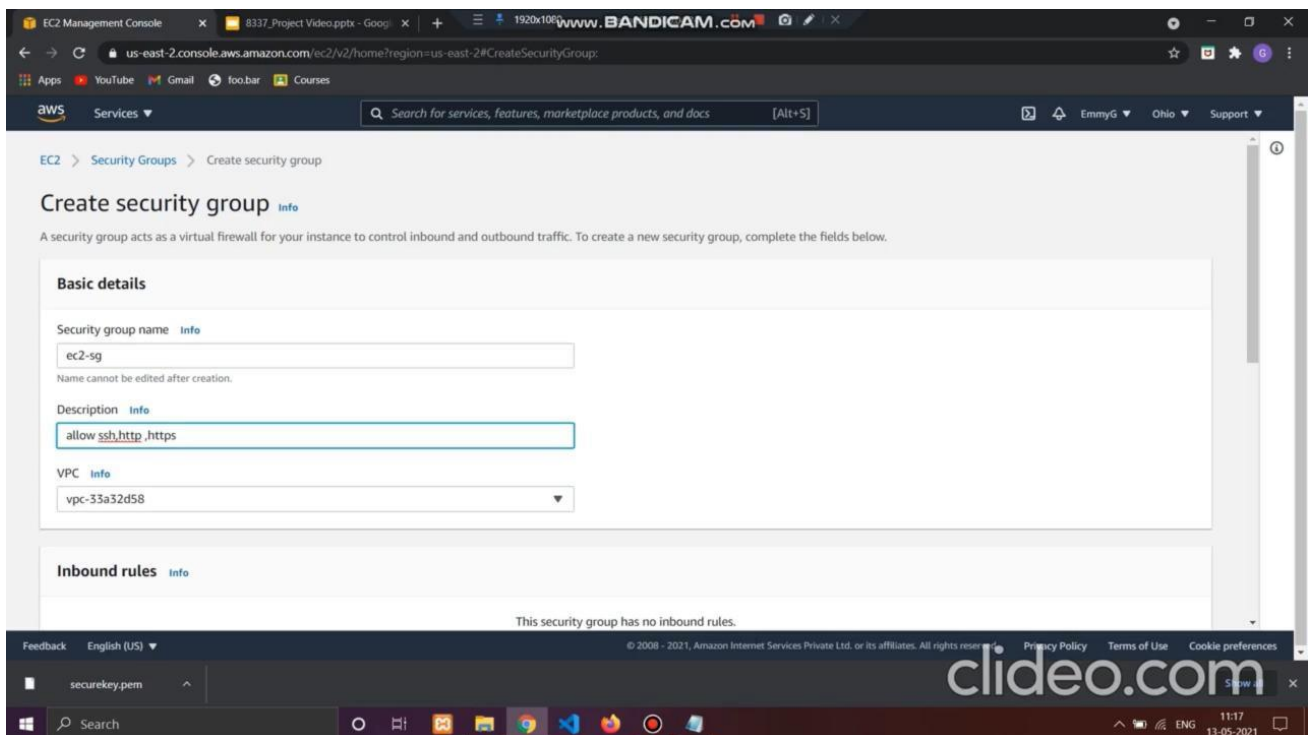
**2.2 Create Policies Select Amazon s3 Full access policy give a name for the role and create**

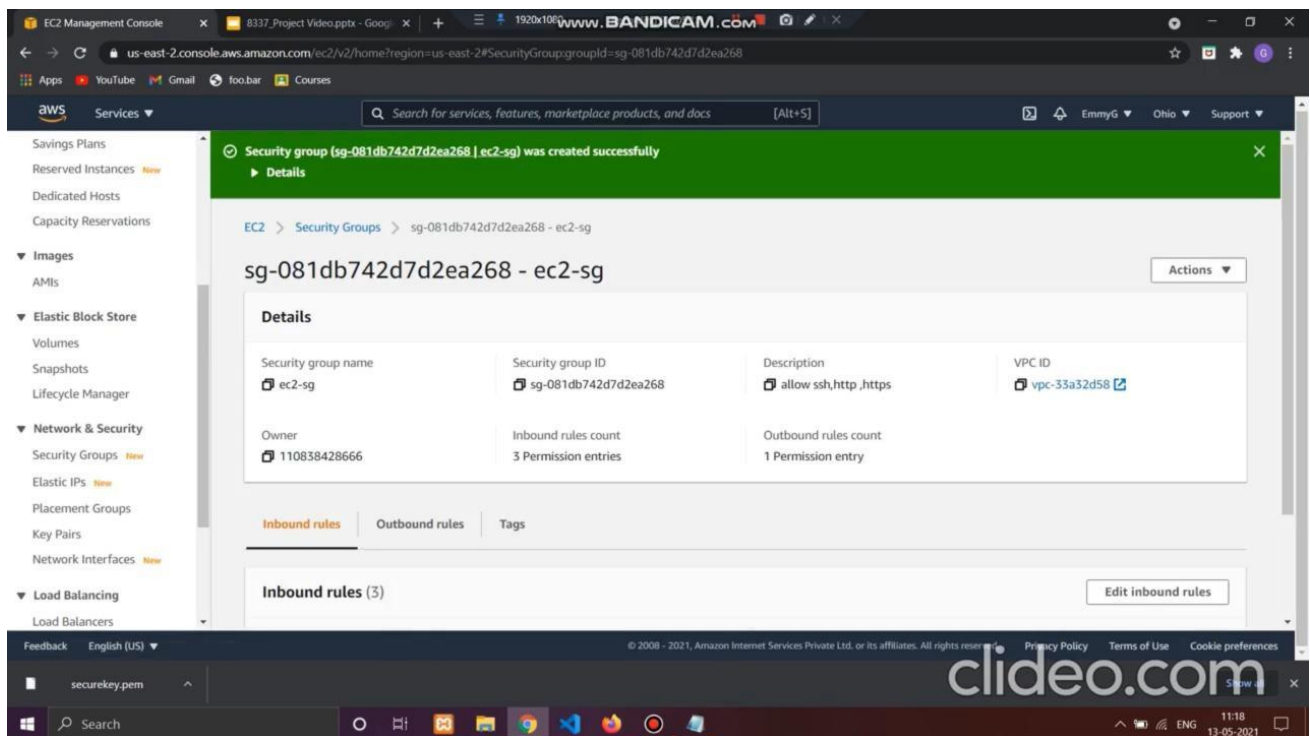
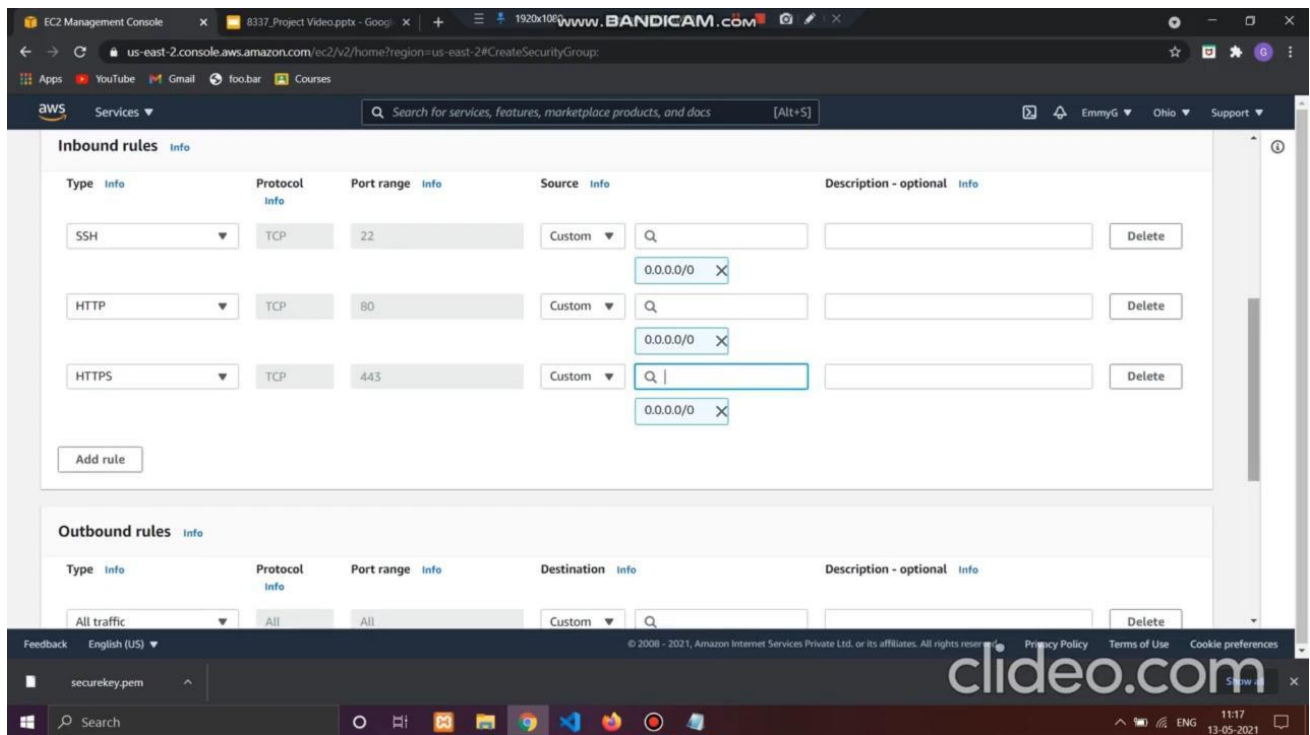


**Step 3: Create a security group** A security group acts as a firewall to our EC2 instance.

**3.1 Navigate to the security group. Add a name and description.**

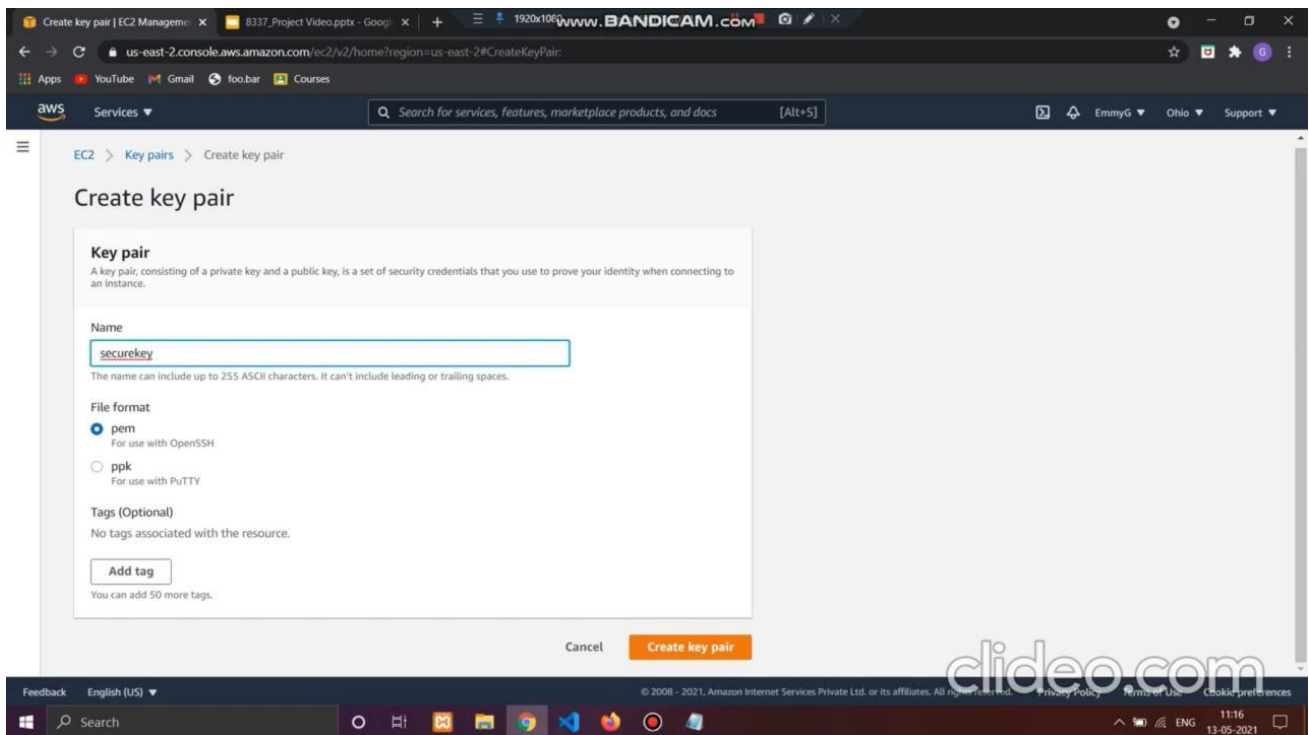
**3.2 We need to set 3 inbound rules. HTTP SSH and HTTPS.**





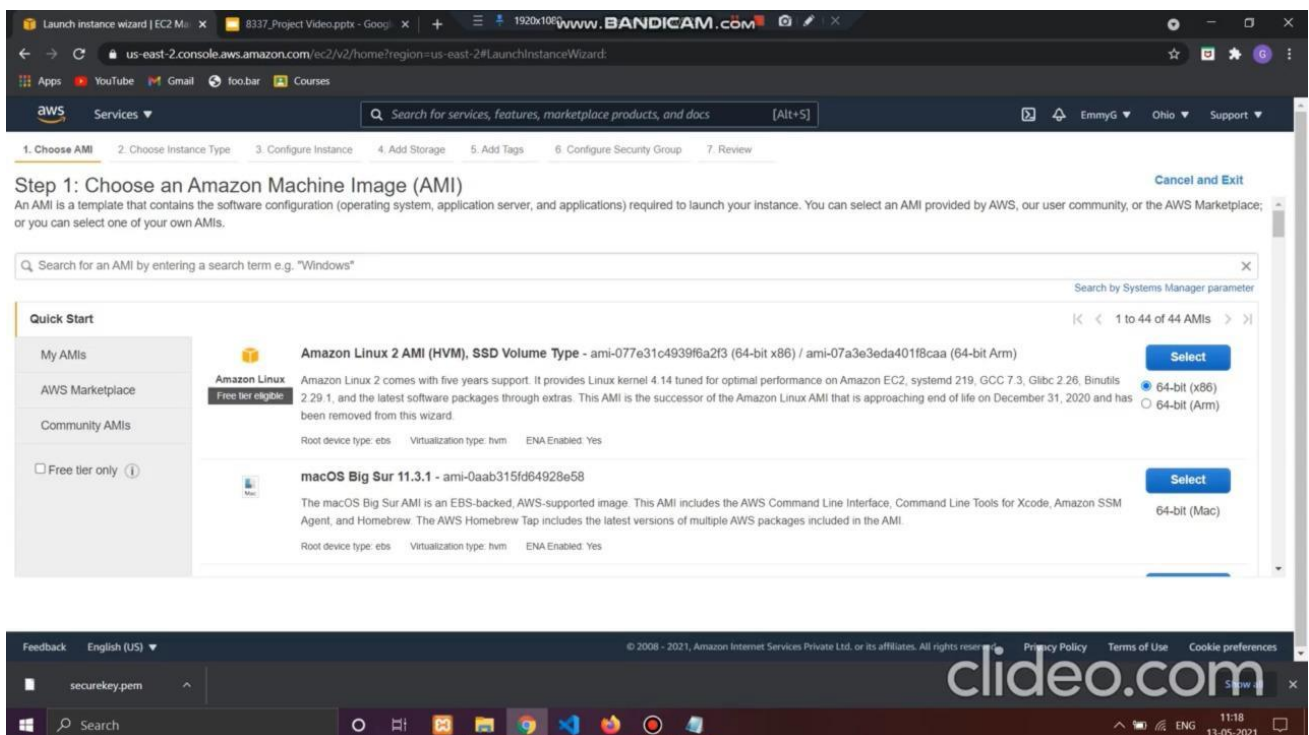
**Step 4: We need to create a key pair. A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance. Amazon EC2 stores the public key, and you store the private key**

**4.1 Navigate to key pair. Give a name to your key pair. And download in pem format.**



Step 5: Now we need to create an ec2 instance

## 5.1 Choose the free tier eligible AMI Machine



## 5.2 Choose t2.micro instance



Launch instance wizard | EC2 M... | 8337\_Project Video.pptx - Google... | 1920x1080 | www.BANDICAM.com

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard

Services Search for services, features, marketplace products, and docs [Alt+S]

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance families Current generation Show/Hide Columns

Currently selected: t2.micro (~ ECUs, 1 vCPUs, 2.5 GHz, ~, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	t2.micro	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

securekey.pem clideo.com 11:18 13-05-2021

### 5.3 In configure part make sure to choose the IAM Role

Launch instance wizard | EC2 M... | 8337\_Project Video.pptx - Google... | 1920x1080 | www.BANDICAM.com

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard

Services Search for services, features, marketplace products, and docs [Alt+S]

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 3: Configure Instance Details

Network vpc-33a32d58 (default) Create new VPC

Subnet No preference (default subnet in any Availability Zone) Create new subnet

Auto-assign Public IP ☒ Yes

Placement group ☐ No

Capacity Reservation ☐ No

Domain join directory ☐ No directory

IAM role ☒ None ☐ s3fullaccess ☐ stop

Shutdown behavior ☐ Stop

Stop - Hibernate behavior ☐ Enable hibernation as an additional stop behavior

Enable termination protection ☐ Protect against accidental termination

Monitoring ☐ Enable CloudWatch detailed monitoring

Cancel Previous Review and Launch Next: Add Storage

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

securekey.pem clideo.com 11:18 13-05-2021

### 5.4 Next leave as default but in configure security group chose the security group created.



**Step 6: Configure Security Group**

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more about Amazon EC2 security groups.](#)

Assign a security group: ☐ Create a new security group ☒ Select an existing security group

Security Group ID	Name	Description	Actions
sg-6efb6d1b	default	default VPC security group	Copy to new
sg-081db742d7d2ea268	ec2-sg	allow ssh,http ,https	Copy to new

Inbound rules for sg-081db742d7d2ea268 (Selected security groups: sg-081db742d7d2ea268)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	
SSH	TCP	22	0.0.0.0/0	

[Cancel](#) [Previous](#) [Review and Launch](#)

**5.5 Review and launch your instance, at that time a keypair needs to be chosen. Chose an existing pair and choose the one we have created.**

**Step 7: Review Instance Launch**

**Select an existing key pair or create a new key pair**

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

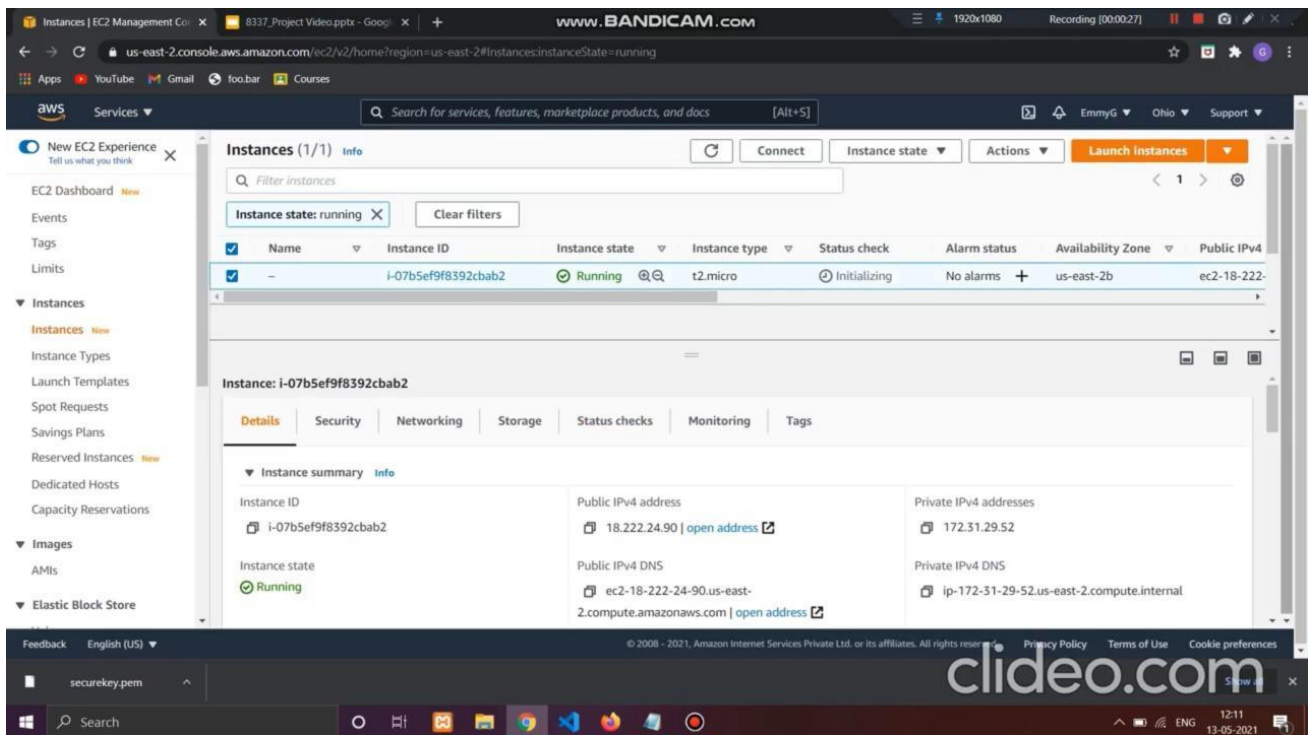
Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Choose an existing key pair  
**Select a key pair**  
securekey

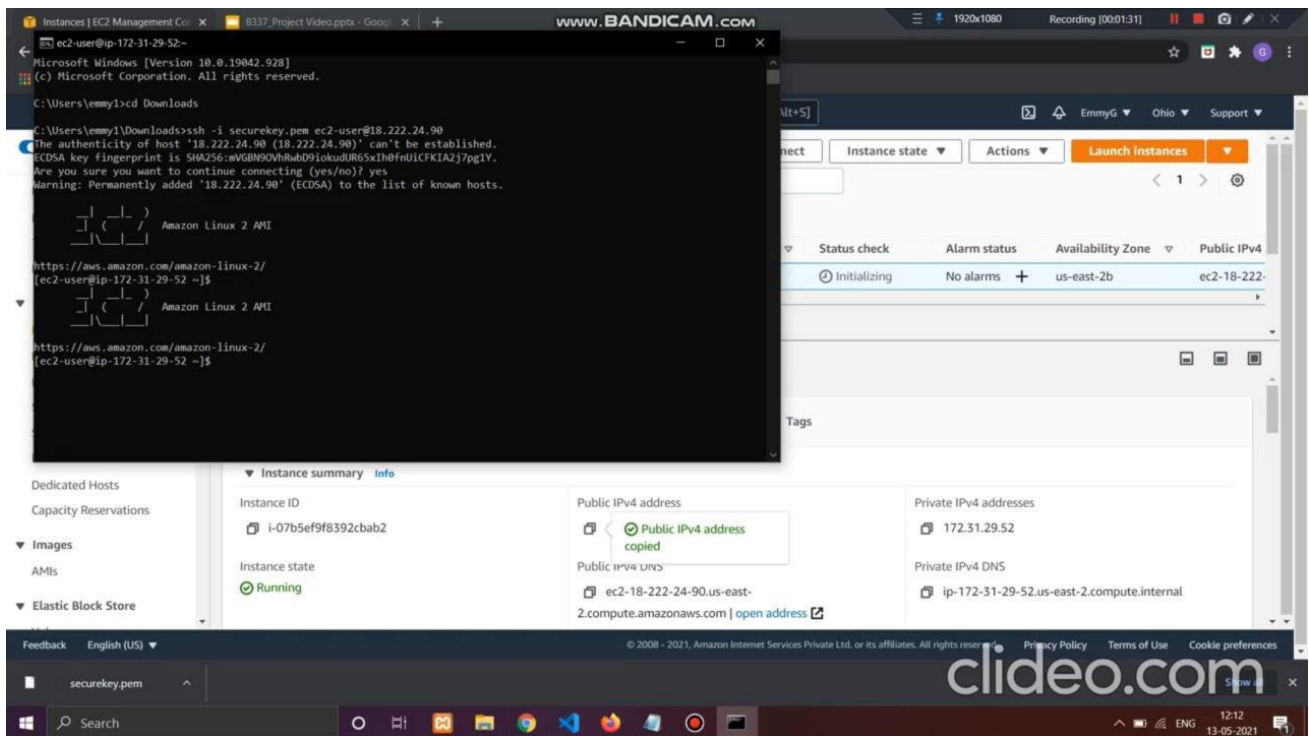
☐ I acknowledge that I have access to the selected private key file (securekey.pem), and that without this file, I won't be able to log into my instance.

[Cancel](#) [Launch Instances](#)

**Step 6: Once the instance is launched and running just copy your Public IP address.**



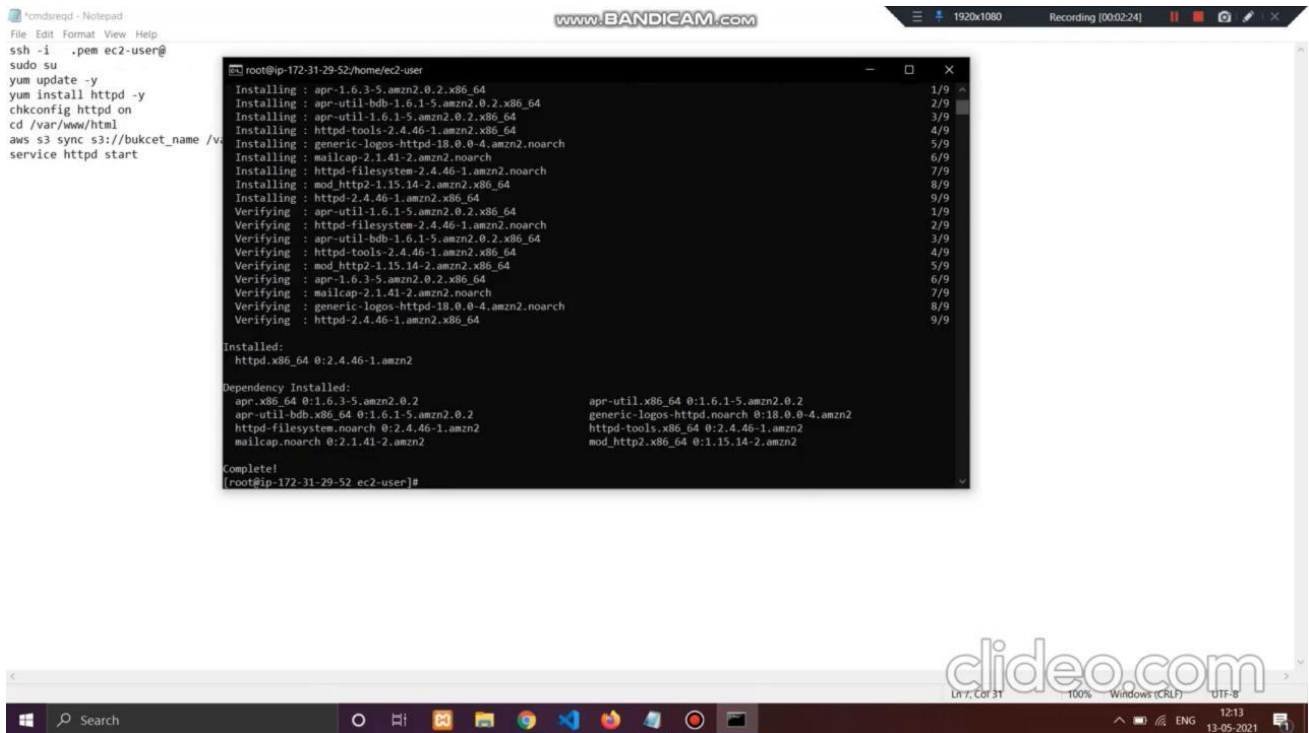
**Step 7:** Next, we need to ssh into our ec2 instance through our terminal. For that open your cmd. And navigate to your downloads folder where our pem file is downloaded. And type the command `ssh -i securitykeypair.pem ec2-user@(public ip address)`



**Step 8:** Now we will be into our ec2 instance.to navigate to root user type the command `sudo su`

**8.1** Then update any required files using `yum update -y`

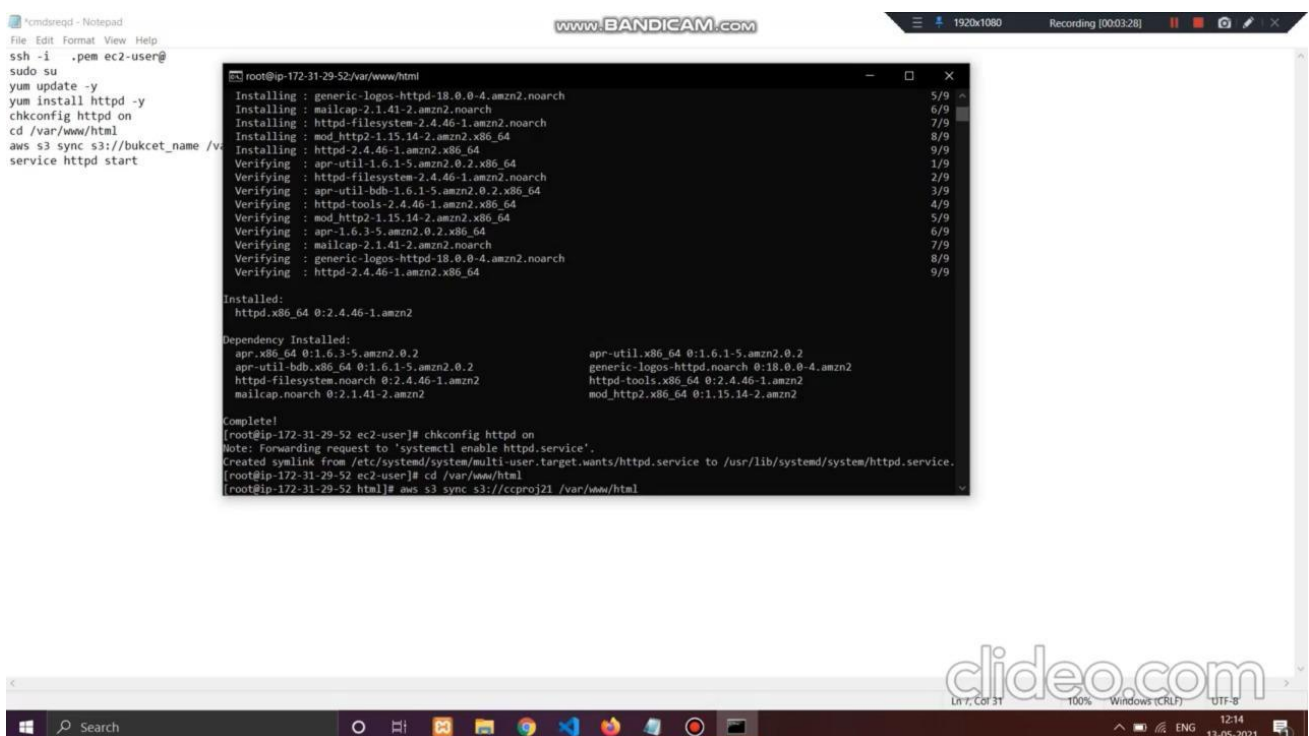
**8.2** Then we need to install apache server using `yum install httpd -y`



**Step 9: We can check whether the server is installed or not using `chkconfig httpd on`**

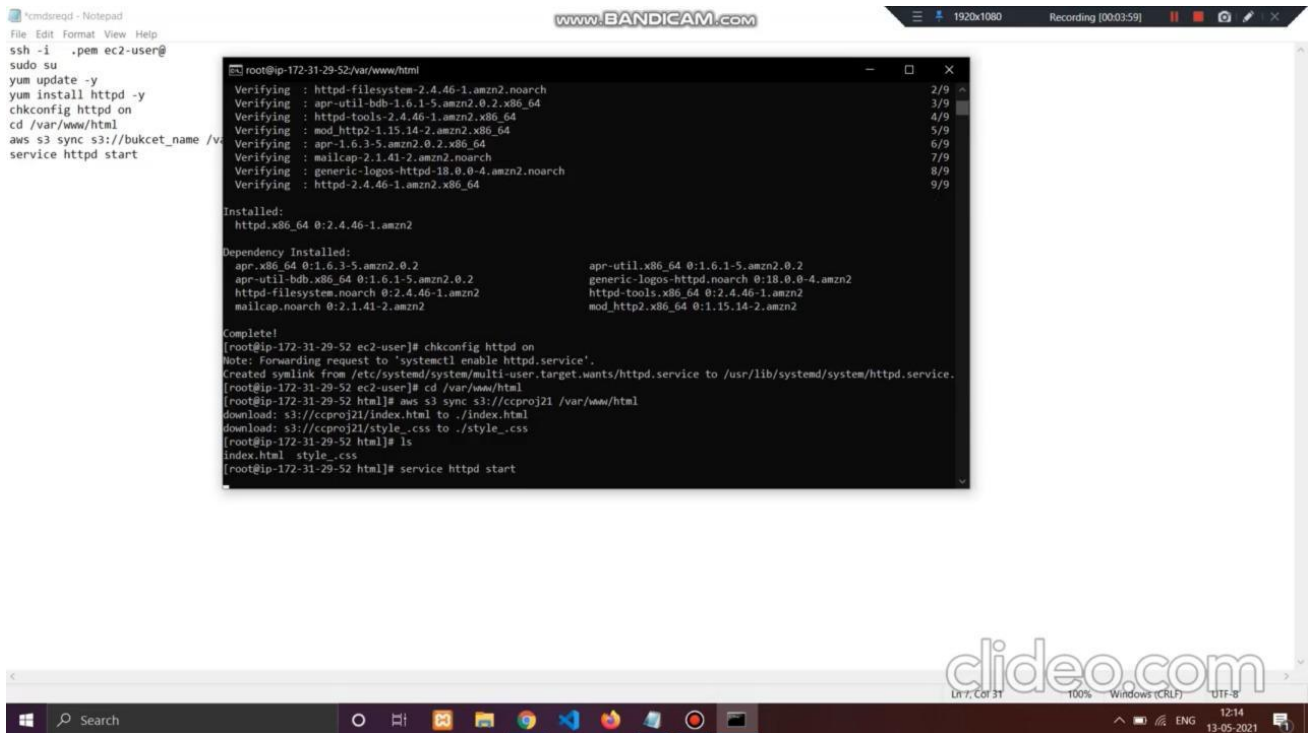
**9.1 Then we need to change to the directory where we will be storing our files. `cd /var/www/html`**

**9.2 Then we need to sync our s3 with aws. using `aws s3 sync s3://bucket name /var/www/html`**

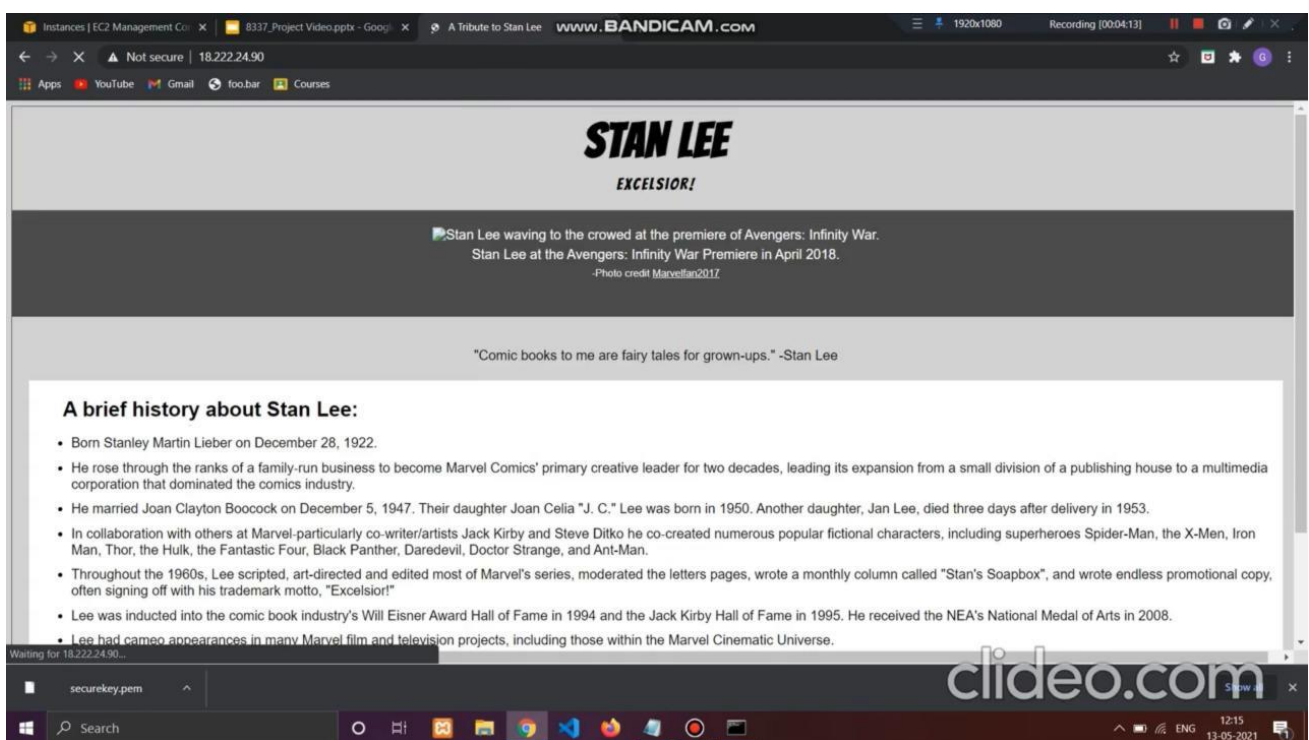


**Step 10: Next we can `ls` and check if our files are uploaded onto the directory**

**10.1 Next we need to start our apache server using `service httpd start`**



Step 11: After this we can navigate to our website using the EC2 public ip we can see our website is hosted



## Conclusion:

Hence, we have learnt how to access files from S3 and deploy a website using EC2 and S3.

## References:

[https://docs.aws.amazon.com/ec2/index.html?nc2=h\\_ql\\_doc\\_ec2](https://docs.aws.amazon.com/ec2/index.html?nc2=h_ql_doc_ec2)

[https://docs.aws.amazon.com/s3/index.html?nc2=h\\_ql\\_doc\\_s3](https://docs.aws.amazon.com/s3/index.html?nc2=h_ql_doc_s3)

[https://docs.aws.amazon.com/cli/index.html?nc2=h\\_ql\\_doc\\_cli](https://docs.aws.amazon.com/cli/index.html?nc2=h_ql_doc_cli)