

# **CLOUD COMPUTING**

**UNIVERSITY OF MUMBAI**

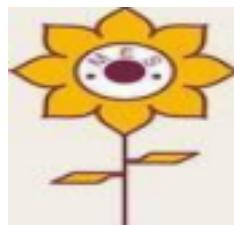
PROJECT ENTITLED

**“Cloud Computing project on AWS”**

SUBMITTED BY  
**SRUSHTI.S.FULPAGAR**

UNDER GUIDANCE OF

Ms. Manasi Khedekar



SUBMITTED FOR THE FULFILLMENT OF THE CURRICULUM  
OF DEGREE  
OF MASTERS OF SCIENCE  
IN DATA ANALYTICS

PILLAI COLLEGE OF ARTS, COMMERCE &  
SCIENCE, NEW PANVEL

2023-2024

# **INDEX**

<b>SR.NO</b>	<b>NAME</b>
1.	Introduction
2.	Technology Used
3.	Project Workflow
4.	Outputs
5.	Case Study
6.	Conclusion

# INTRODUCTION

Amazon Web Services (AWS) comprises over one hundred services, each of which exposes an area of functionality. With Elastic Beanstalk, we can quickly deploy and manage applications in the AWS Cloud without having to learn about the infrastructure that runs those applications. Elastic Beanstalk reduces management complexity without restricting choice or control.

Elastic Beanstalk supports applications developed in Go, Java, .NET, Node.js, PHP, Python, and Ruby. When we deploy your application, Elastic Beanstalk builds the selected supported platform version and provides one or more AWS resources, such as Amazon EC2 instances, to run your application.

Create an application, upload an application version in the form of an application source bundle Elastic Beanstalk, and then provide some information about the application. Elastic Beanstalk automatically launches an environment and creates and configures the AWS resources needed to run your code. After your environment is launched, you can then manage your environment and deploy new application versions.

# TECHNOLOGY USED

- **AWS (Amazon Web Services):** AWS is the leading cloud service provider offering a vast array of cloud computing services including computing power, storage, and databases, along with a suite of tools for deployment, management, and scaling.
- **EC2 (Elastic Compute Cloud):** EC2 is a foundational service within AWS, providing resizable compute capacity in the cloud. It allows users to launch virtual machines (instances) with various configurations to meet their specific needs.
- **Windows Microsoft Platform:** This project leverages the Windows operating system provided by AWS for its EC2 instances. Windows instances are ideal for running applications that require the Microsoft ecosystem, offering compatibility and support for a wide range of enterprise software.
- **Security Groups:** Security groups act as virtual firewalls for EC2 instances, controlling inbound and outbound traffic. In this project, security groups are configured to allow HTTPS and HTTP traffic, ensuring secure communication with the virtual machine.
- **Key Pairs:** Key pairs are used for secure access to EC2 instances. A public key is used to encrypt data, while the corresponding private key is required for decryption. Users generate and upload key pairs to securely connect to their instances.
- **RDP (Remote Desktop Protocol):** RDP is a proprietary protocol developed by Microsoft for remote access to Windows-based systems. In this project, RDP is utilized to connect to the Windows EC2 instance from a local desktop.
- **XAMPP Server:** XAMPP is an open-source cross-platform web server solution stack package developed by Apache Friends. It includes Apache HTTP Server, MySQL database, PHP, and Perl programming languages. XAMPP is used in this project to set up a local web server environment for testing purposes.

# PROJECT WORKFLOW

- 1. AWS Account Sign-In:** The project begins with signing in to the AWS Management Console.
- 2. EC2 Instance Creation:** Within the EC2 dashboard, a new instance is created, selecting the Windows Microsoft platform.
- 3. Security Group Configuration:** Security groups are configured to allow HTTPS and HTTP traffic to the instance.
- 4. Key Pair Creation:** A new key pair is generated for secure access to the instance.
- 5. Instance Connection:** After the instance is created, it is accessed via RDP client using the generated key pair.
- 6. Desktop Integration:** The virtual machine's desktop is accessed from the local desktop using RDP.
- 7. Firewall Configuration:** Windows firewall settings are adjusted to allow inbound connections for the desired services.
- 8. XAMPP Server Setup:** XAMPP server is downloaded and installed on the virtual machine.
- 9. Database Testing:** The functionality of the database is verified by accessing it through localhost, and test records are added to ensure proper operation.

By following this workflow and leveraging the technology stack mentioned above, this project demonstrates the seamless deployment and management of Windows-based EC2 instances on AWS, along with the setup and testing of a local web server environment.

# OUTPUTS

## 1) Sign in to your AWS account:



Sign in

Root user  
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

IAM user  
User within an account that performs daily tasks. [Learn more](#)

Root user email address

jsrushti23mda@student.mes.ac.in

Next

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

New to AWS?

Create a new AWS account



## 2) Select EC2 option from computer:

Services  X

CloudSearch

Services (13)

Features (57)

Resources (New)

Documentation (34,632)

Knowledge Articles (609)

Marketplace (3,439)

Blogs (2,141)

Events (30)

Tutorials (21)

Search results for 'ec2'

Services

See all 13 results ▾

- EC2 ★ Virtual Servers in the Cloud
- EC2 Image Builder ★ A managed service to automate build, customize and deploy OS images
- Recycle Bin Protect resources from accidental deletion
- Amazon Inspector ★ Continual vulnerability management at scale

### 3) Click on instances(running):

The screenshot shows the AWS EC2 Free Tier dashboard. At the top, it displays the number of EC2 resources in the US East (Ohio) Region: Instances (running) = 0, Auto Scaling Groups = 0, Dedicated Hosts = 0, Elastic IPs = 0, Instances = 0, Key pairs = 0, Load balancers = 0, Placement groups = 0, Security groups = 1, Snapshots = 0, and Volumes = 0. To the right, there's a section titled "EC2 Free Tier" which says "Offers for all AWS Regions." It indicates 2 EC2 free tier offers in use. Below this, it shows the end of month forecast and any offers forecasted to exceed the free tier limit. There's also a link to "View Global EC2 resources". On the left, there's a "Launch instance" section with a "Launch instance" button and a "Migrate a server" link. Below this, a note says "Note: Your instances will launch in the US East (Ohio) Region". To the right, there's a "Service health" section showing the region as US East (Ohio) and the status as "This service is operating normally". Below this, there's a "Offer usage (monthly)" section for Linux EC2 Instances and Storage space on EBS, both at 0% usage.

### 4) Create an instance:

The screenshot shows the "Launch an instance" wizard. The top navigation bar shows "EC2 > Instances > Launch an instance". The main section is titled "Launch an instance" with an "Info" link. It says "Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below." Below this, there's a "Name and tags" section where the name "Srushti" is entered. There's also a link to "Add additional tags". The next section is titled "Application and OS Images (Amazon Machine Image)" with an "Info" link. It explains what an AMI is and provides a search bar to find AMIs. To the right, there's a "Summary" section that lists the configuration: Number of instances (1), Software Image (AMI) (Amazon Linux 2023.3.2...), Virtual server type (instance type) (t2.micro), Firewall (security group) (New security group), Storage (volumes) (1 volume(s) - 8 GiB), and a note about the Free tier (in the first year, it includes 750 hours of t2.micro or t3.micro instances in regions where t2.micro is unavailable).

5)Select Windows Microsoft platform and also check whether it is in the free tier or not .

The screenshot shows the AWS Lambda console interface. At the top, there's a search bar with placeholder text: "Search our full catalog including 1000s of application and OS images". Below the search bar is a "Quick Start" section featuring logos for various operating systems: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux. A "Browse more AMIs" link is available, along with a note about including AMIs from AWS Marketplace and the Community. On the left, under "Amazon Machine Image (AMI)", a specific item is listed: "Microsoft Windows Server 2022 Base" (ami-0e6aa5f69f06ffa91). This item is marked as "Free tier eligible". Below this, detailed information is provided: "Virtualization: hvm", "ENA enabled: true", and "Root device type: ebs". The "Description" section notes that it's a "Microsoft Windows Server 2022 Full Locale English AMI provided by Amazon". Under "Architecture", it specifies "64-bit (x86)" and "AMI ID: ami-0e6aa5f69f06ffa91", with a "Verified provider" badge. To the right, configuration options include "Number of instances" set to 1, "Software Image (AMI)" set to Microsoft Windows Server 2022, "Virtual server type (instance type)" set to t2.micro, "Firewall (security group)" set to "New security group", and "Storage (volumes)" showing 1 volume(s) - 30 GiB. A callout box highlights the "Free tier" benefit, stating: "Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance". At the bottom right are "Cancel" and "Launch instance" buttons.

6)Then Allow HTTPS and HTTP options from the security group.

The screenshot shows the "Network settings" configuration page. It includes sections for "Network" (vpc-0086fc43da7130383), "Subnet" (No preference), "Auto-assign public IP" (Enable), and "Firewall (security groups)". The "Create security group" option is selected. Below, a note states: "We'll create a new security group called 'launch-wizard-1' with the following rules:". Three rules are listed: "Allow RDP traffic from Anywhere (0.0.0.0/0)", "Allow HTTPS traffic from the internet", and "Allow HTTP traffic from the internet". Each rule has a descriptive subtitle.

## 7)Create new key pair:

The screenshot shows the 'Key pair (login)' configuration page. It includes a note about using a key pair for secure connection, a required key pair name input field containing 'srush', and a link to 'Create new key pair'.

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

srush

Create new key pair

For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance.

## 8)Click on view instances:

The screenshot shows the EC2 Instances page after launching an instance. A green success message indicates the instance was successfully launched. Below it, there's a 'Launch log' link and a 'Next Steps' section with a search bar.

EC2 > Instances > Launch an instance

Success  
Successfully initiated launch of instance (i-01a3b9576fcb02a5f)

▶ Launch log

Next Steps

What would you like to do next with this instance, for example "create alarm" or "create backup"

## 9)Select an instance and click on connect:

The screenshot shows the EC2 Instances list page. An instance named 'Srushti' (ID: i-01a3b9576fcb02a5f) is selected. The 'Actions' dropdown menu is open, showing options like 'Connect', 'Instance state', 'Actions', and 'Launch instances'.

Instances (1/1) Info

Find Instance by attribute or tag (case-sensitive)

Name Instance ID Instance state Instance type Status check Alarm status Availability Zone Public IPv4

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
Srushti	i-01a3b9576fcb02a5f	Running	t2.micro	2/2 checks passed	View alarms	us-east-2a	ec2-18-216-1

## 10) Select RDP client to access the virtual machine on our desktop:

Session Manager | **RDP client** | EC2 serial console

Instance ID  
 i-01a3b9576fcb02a5f (Srushti)

Connection Type

Connect using RDP client  
Download a file to use with your RDP client and retrieve your password.

Connect using Fleet Manager  
To connect to the instance using Fleet Manager Remote Desktop, the SSM Agent must be installed and running on the instance. For more information, see [Working with SSM Agent](#)

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

## 11) Click on upload to upload the key pair file which we have created earlier.

### Get Windows password Info

Use your private key to retrieve and decrypt the initial Windows administrator password for this instance.

Instance ID  
 i-01a3b9576fcb02a5f (Srushti)

Key pair associated with this instance  
 srush

Private key  
Either upload your private key file or copy and paste its contents into the field below.

srush.ppk  
1.423KB

Private key contents - *optional*

```
PuTTY-User-Key-File-2: ssh-rsa
Encryption: none
Comment: srush
Public-Lines: 6
AAAAB3NzaC1yc2EAAAQABAAQCNsAkWcDKVXGp6crG8RgzDyY13tpGEmsRN
3MfkRBpjcdDeJ8G5X+ju+4r8VK5nN+AlytG72vOdFQBfljOWKZkTbZvid6tgQw5/
XxyKYYAwUzfdyVOvyW6G5pG/9F0YF3nlWpxLza4psyizlI17FK9LtsUPUpuDuWre
ah5h0+2la1oY9zJJhPep9p6WXmmqJTIDMKKzObNuRNEfD6UU6fdqCr5grXhmzHxT
```

## 12) In the windows search bar search for remote desktop connection.

(rushti)

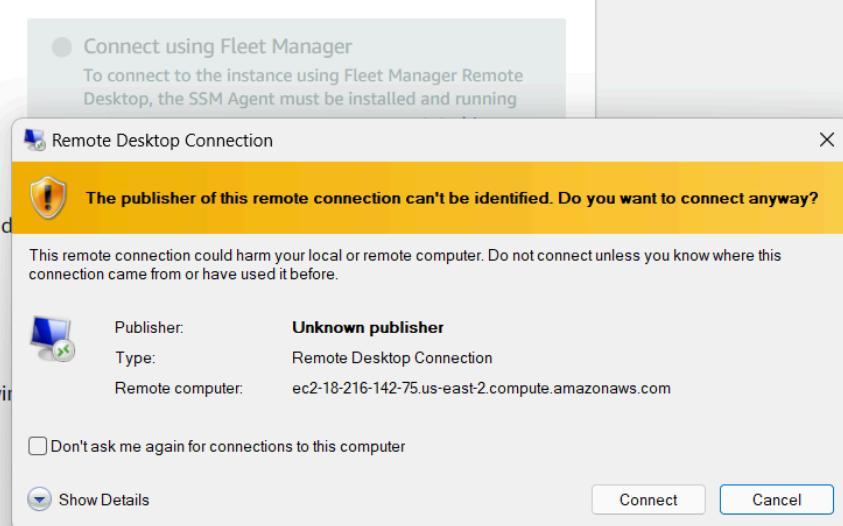
ent  
th your RDP client and retrieve

nindows instance using a remote d  
le below:

esktop file

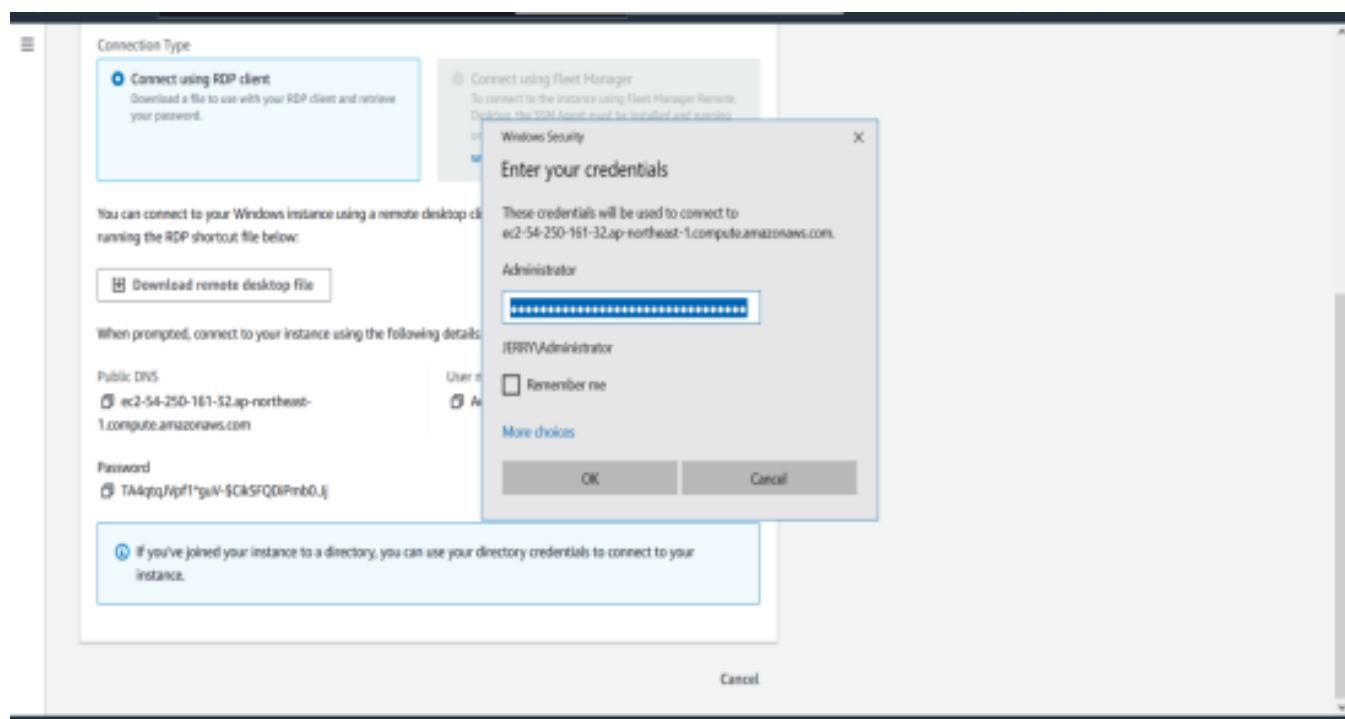
) your instance using the followin

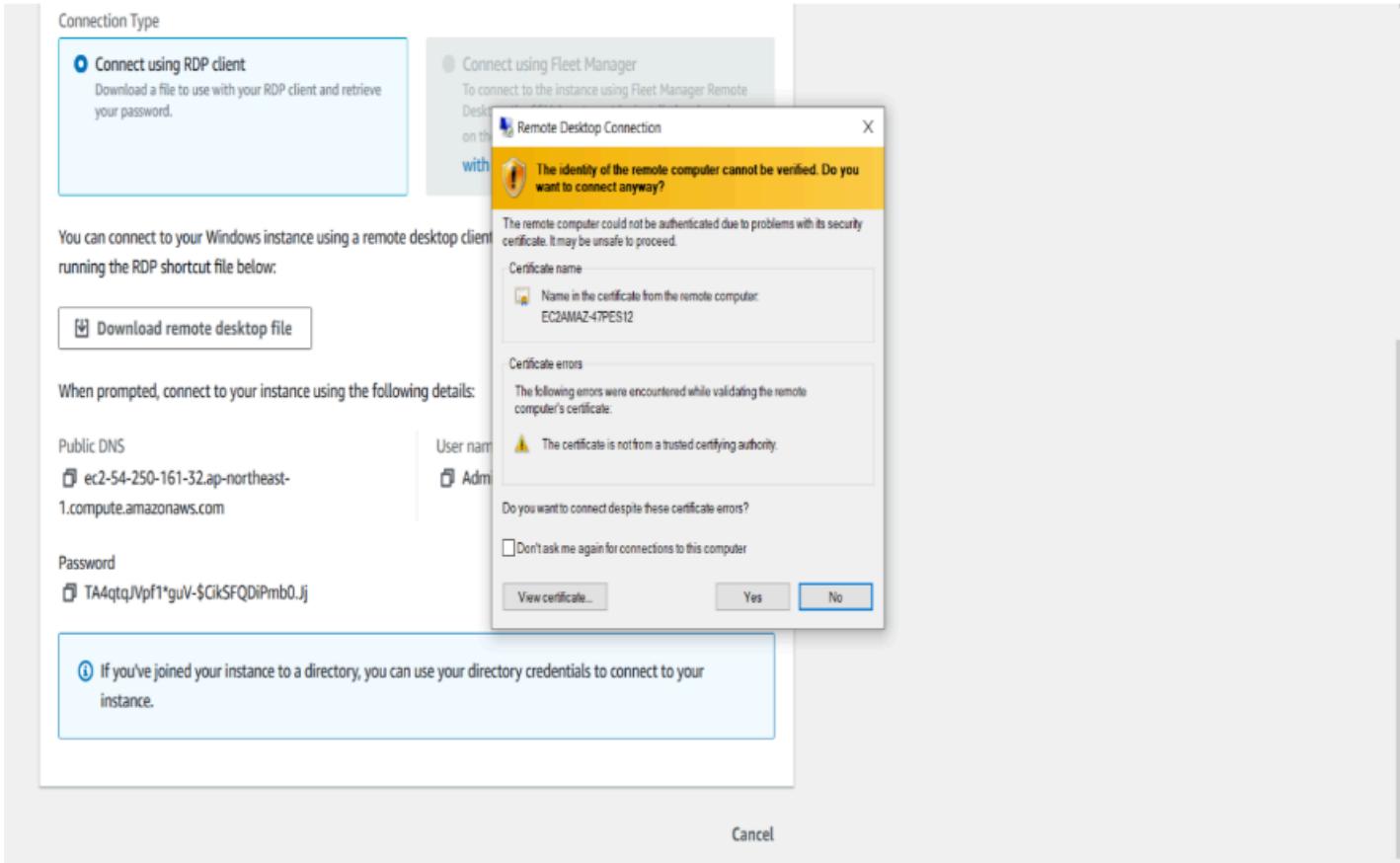
east-



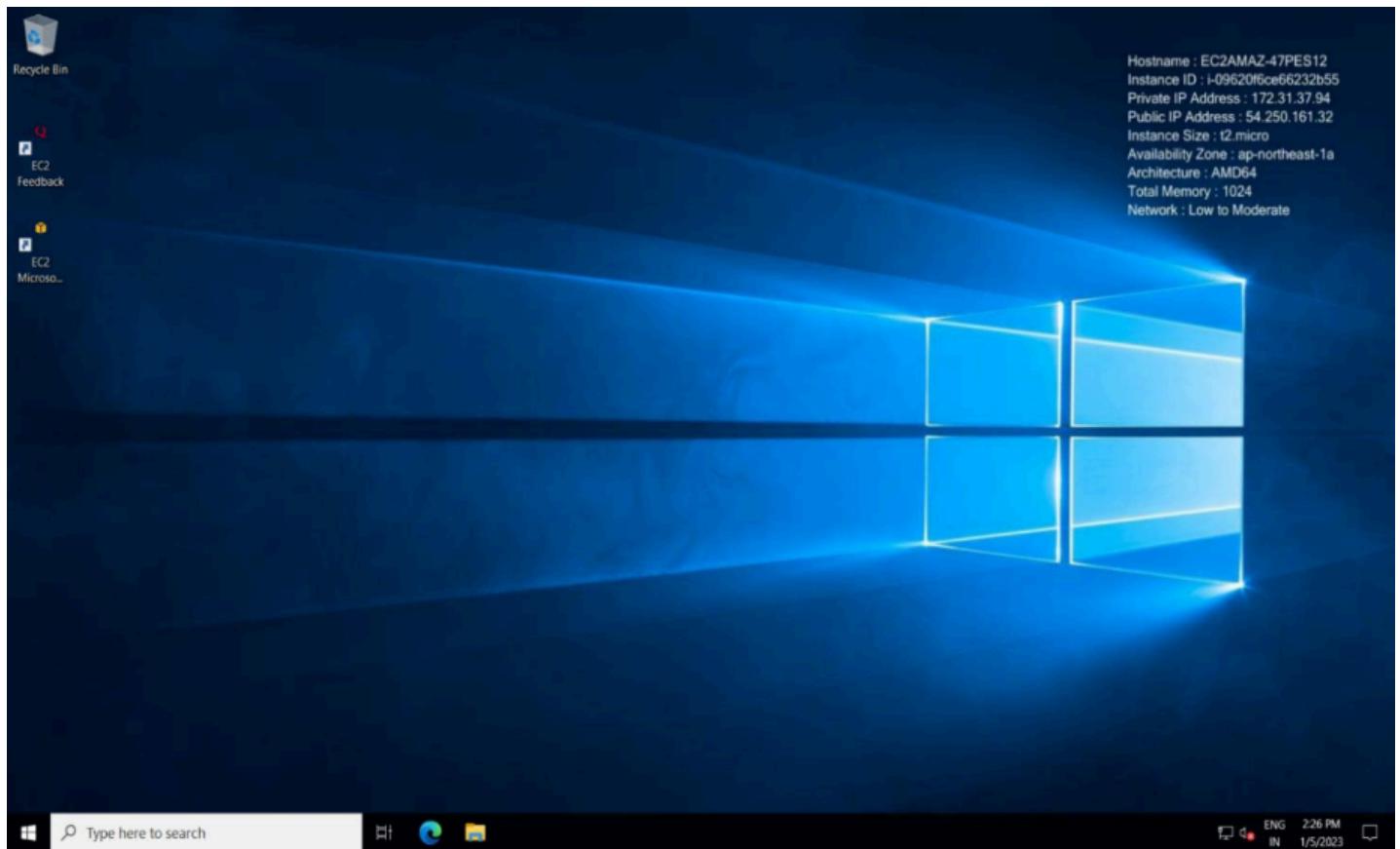
instance to a directory, you can use your directory credentials to connect to your

## 13) Fill the username and password and click on OK.

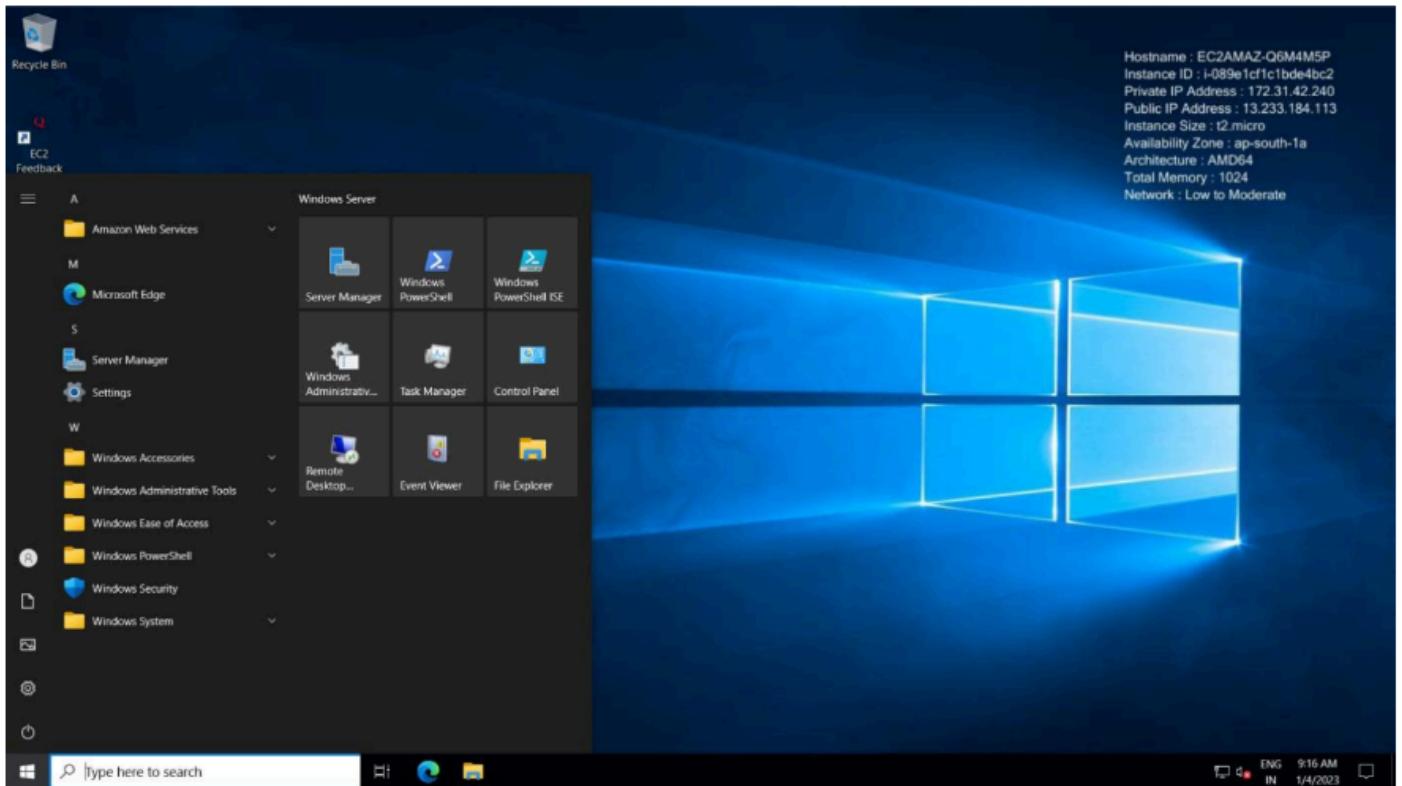




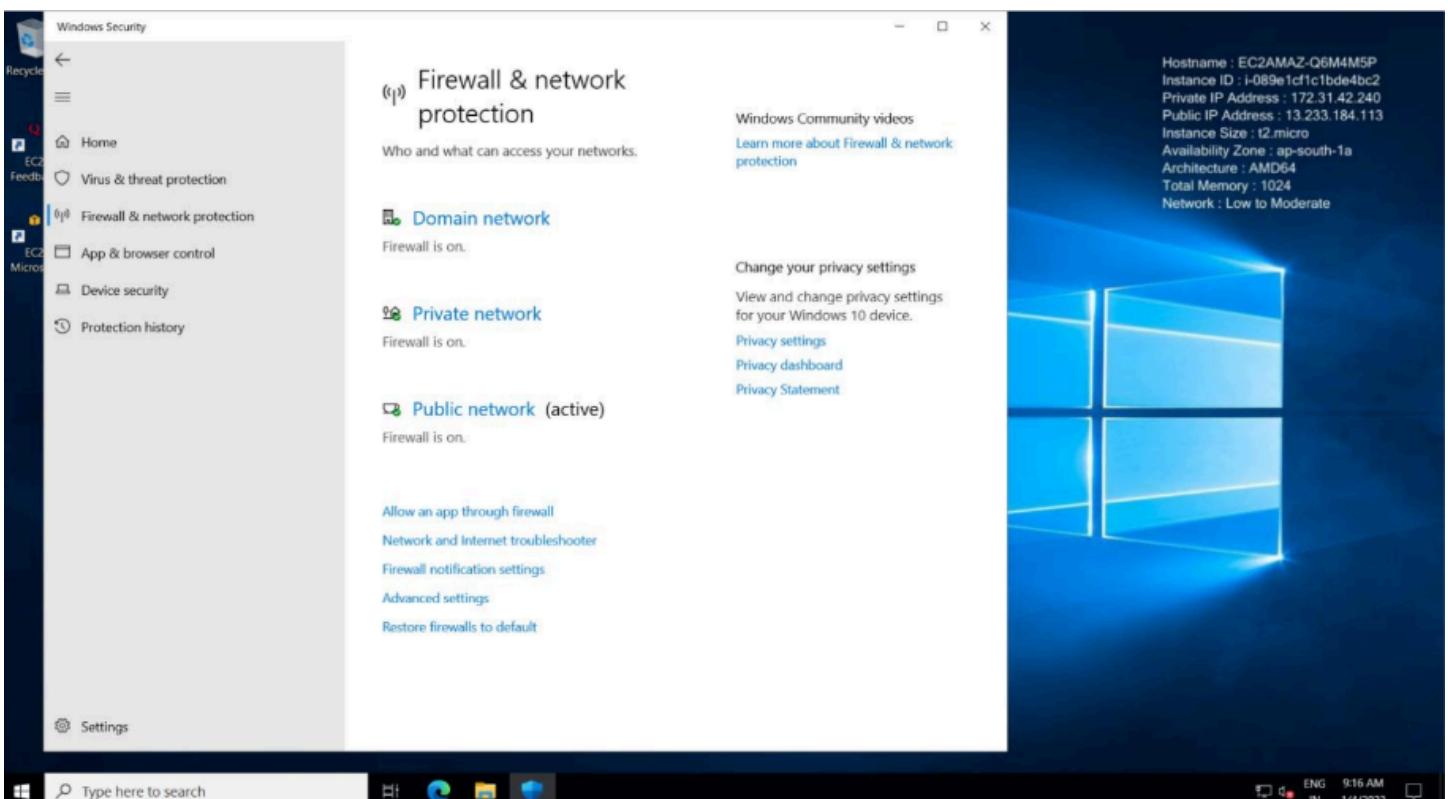
14)Now the virtual machine screen will open on our desktop.



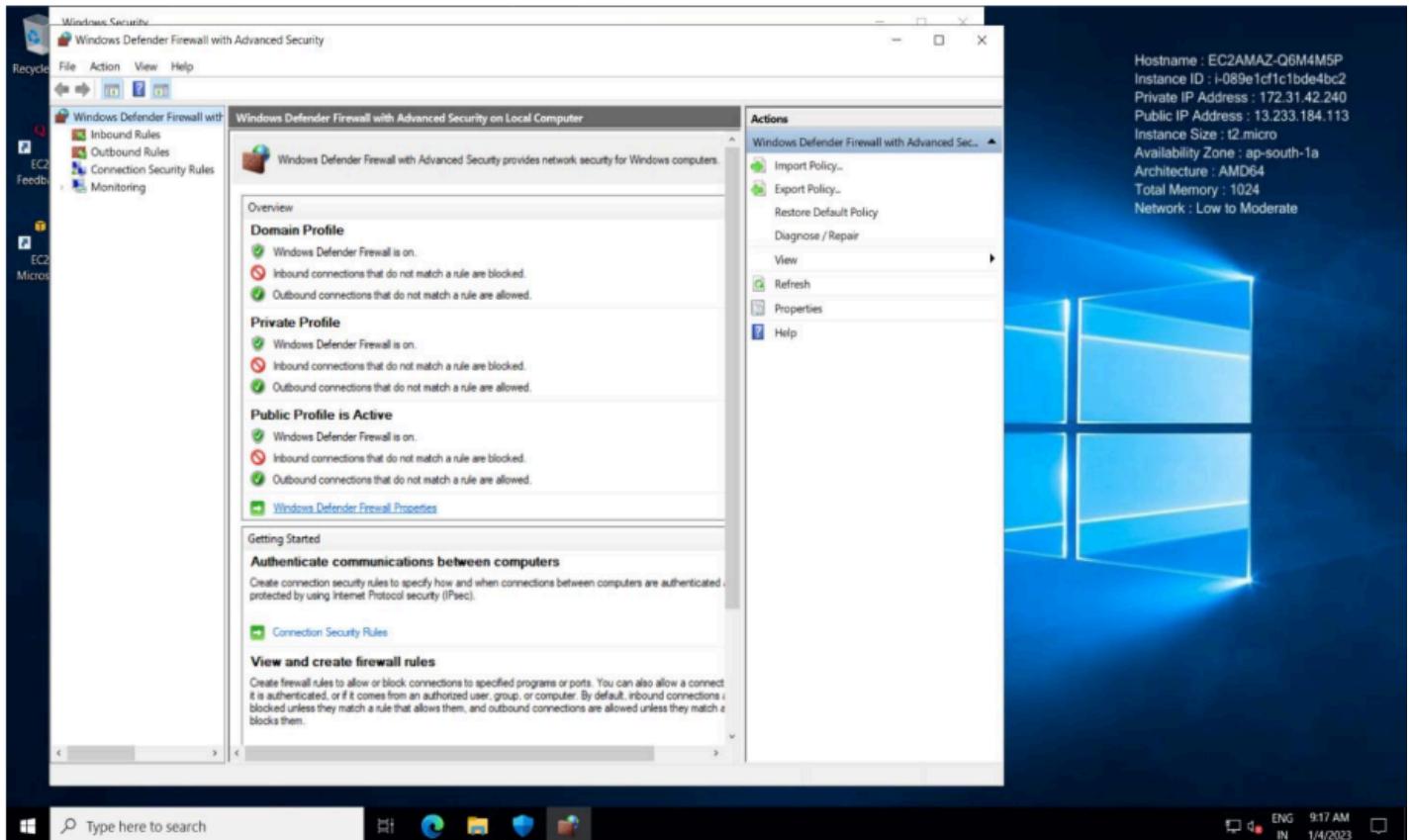
15)click on windows security:



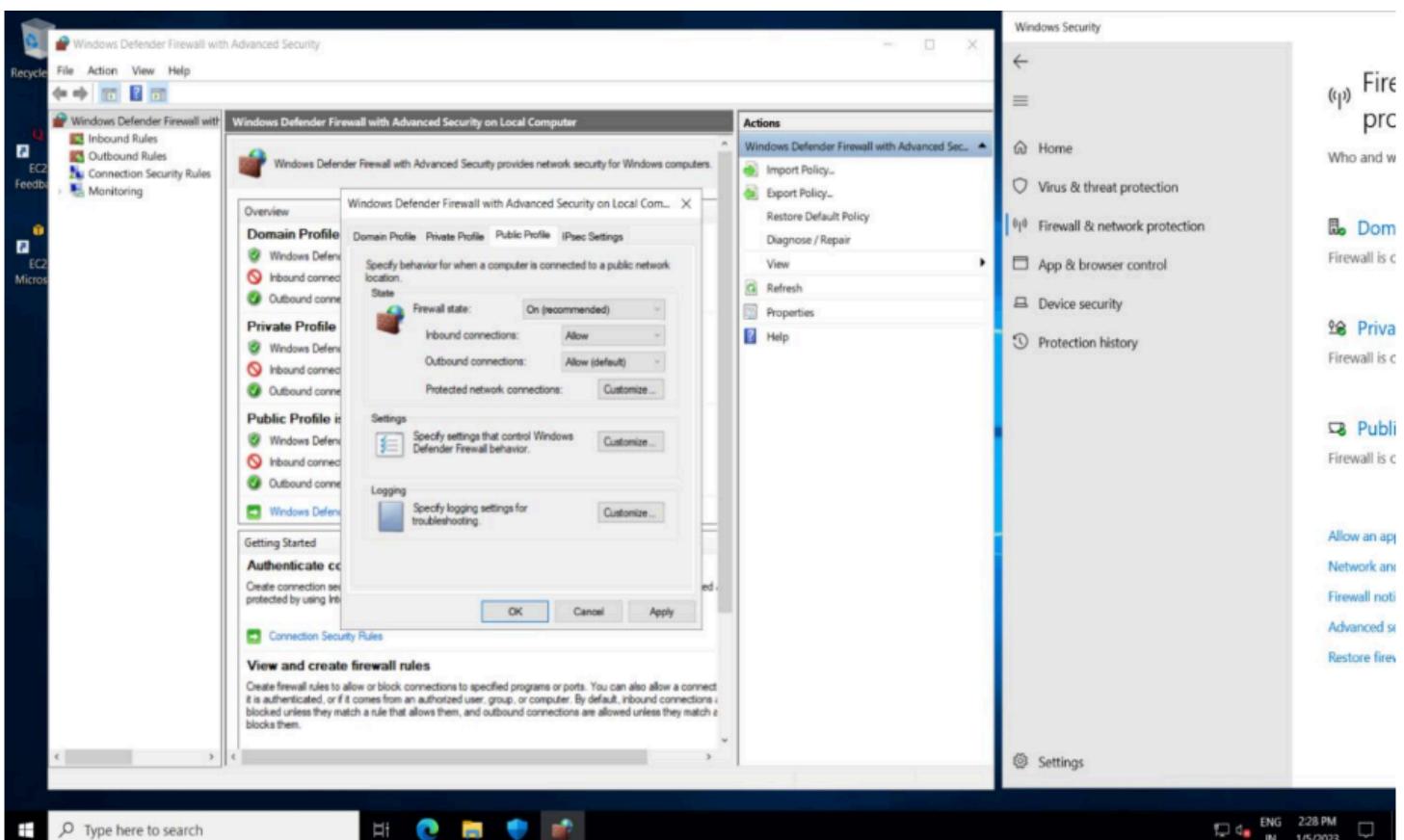
16)Click on firewall and network protection:



17) Click on advanced settings and click on windows defender firewall properties.



18) In public profile click on Inbound connection and select allow:

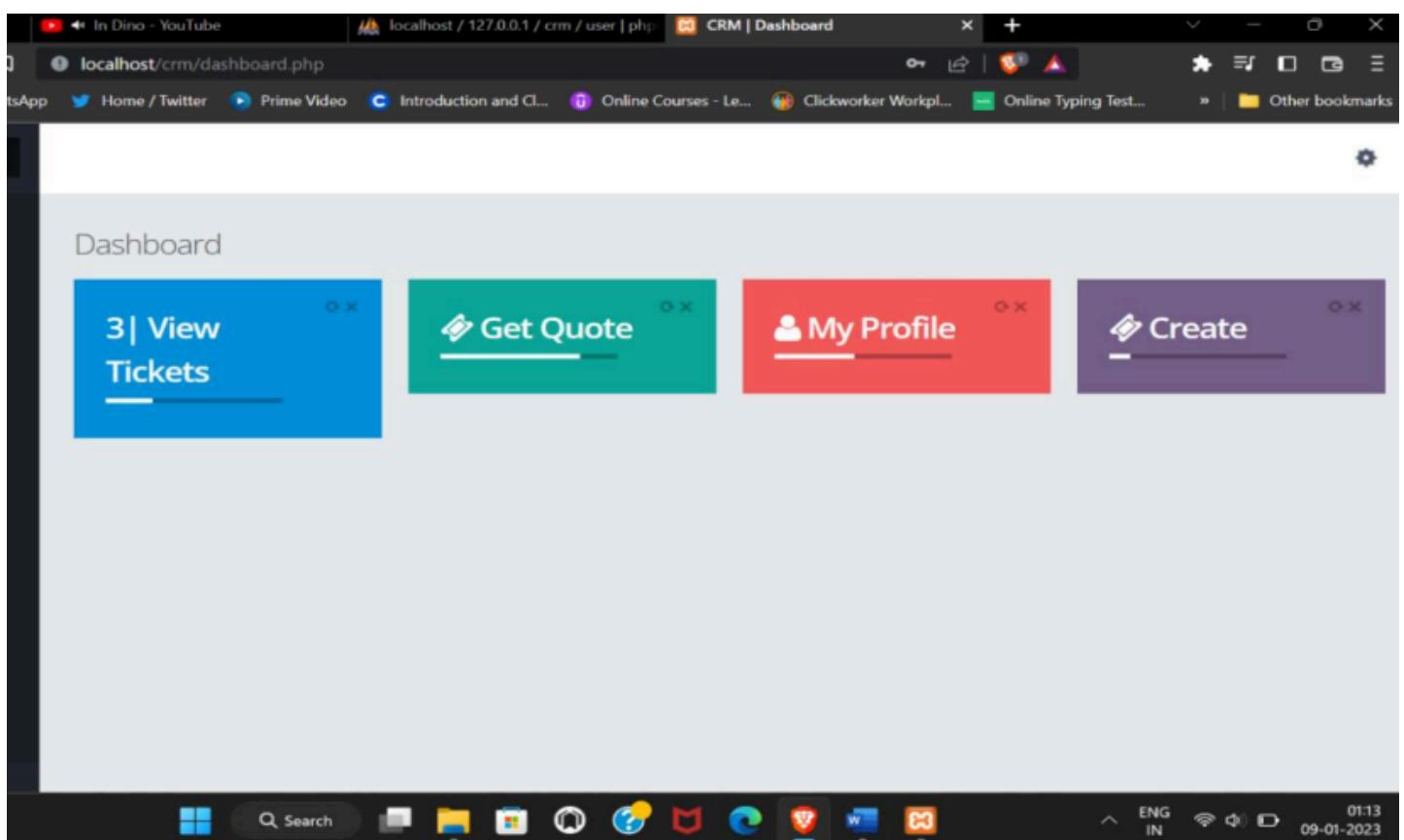
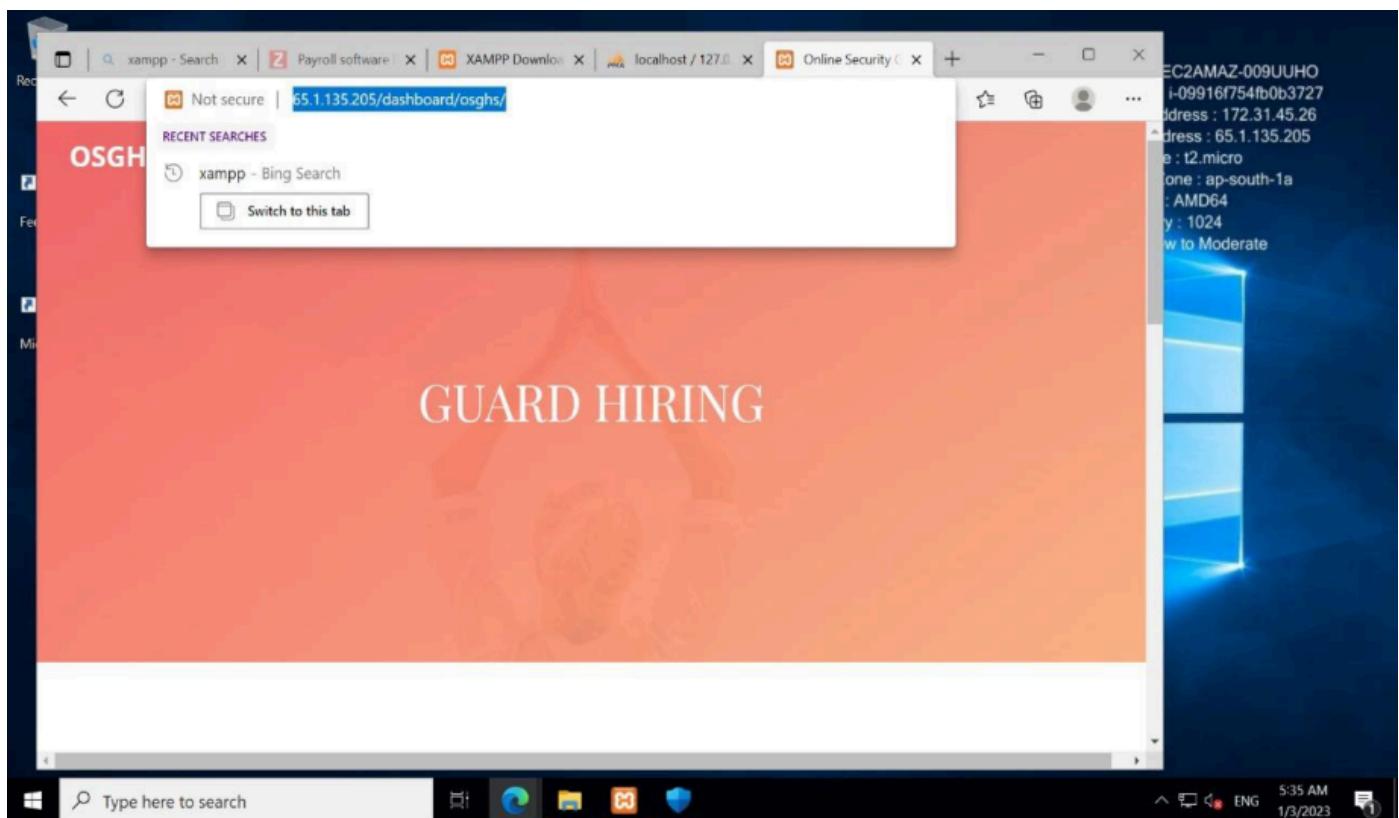


## 19) Next step is download xampp server:

The screenshot shows a Microsoft Edge browser window with the search term "xampp download" entered. The results page for Apache Friends' XAMPP website is displayed. The page includes links for "Download", "Features", "Forums", and "System requirements". It also features a large orange "XAMPP" logo and a brief description of the software. On the left, there are language links for "De" and "Zh CN". Below the main content, there's a section for desktop wallpaper and social media links for "apachefriends.org".

The screenshot shows the XAMPP Control Panel v3.3.0 window running on a Windows 10 desktop. The control panel lists several services: Apache (running on port 80, 443), MySQL (running on port 3306), FileZilla, Mercury, and Tomcat. The Apache and MySQL services are highlighted in green, indicating they are active. The control panel interface includes buttons for Stop, Admin, Config, Logs, Shell, Explorer, Services, Help, and Quit. To the right of the control panel, system information is displayed, including the host name, instance ID, private and public IP addresses, instance size, availability zone, architecture, total memory, and network status. The Windows taskbar at the bottom shows the Start button, a search bar, and pinned icons for File Explorer, Edge, and FileZilla. The system tray shows battery level, signal strength, and the date/time.

20)on new tab type localhost/dashboard/osghs/



21)check whether the database is working properly or not and add some record in it.

The screenshot shows the phpMyAdmin interface for the 'agmsdb' database. The left sidebar lists databases and tables, including 'tblenquiry'. The main area shows a query results page for the 'tblenquiry' table. A yellow box highlights the message '1 row affected. (Query took 0.0111 seconds.)' above the table. The table has columns: ID, EnquiryNumber, Artpdid, FullName, Email, MobileNumber, Message, EnquiryDate, Status, AdminRemark, and AdminRemarkdate. One row is displayed with values: ID=3, EnquiryNumber=125915125, Artpdid=3, FullName=sharon, Email=bts123@gmail.com, MobileNumber=3452761345, Message=bts i love you, EnquiryDate=2023-01-05 15:04:31, Status=NULL, AdminRemark=NULL, AdminRemarkdate=NULL. Below the table are 'Query results operations' buttons: Print, Copy to clipboard, Export, Display chart, Create view, and a 'Bookmark this SQL query' button.

ID	EnquiryNumber	Artpdid	FullName	Email	MobileNumber	Message	EnquiryDate	Status	AdminRemark	AdminRemarkdate
3	125915125	3	sharon	bts123@gmail.com	3452761345	bts i love you	2023-01-05 15:04:31	NULL	NULL	NULL

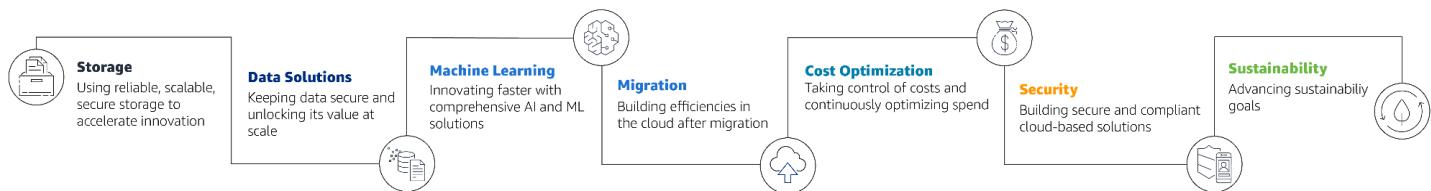
# CASE STUDY



## Pinterest's AWS Migration Journey

### INTRODUCTION

Pinterest, a visual discovery engine, enables users to discover and save creative ideas for various projects and interests. With over 450 million monthly active users, Pinterest faces immense pressure to ensure seamless performance, scalability, and reliability of its platform. Using compute solutions from AWS, Pinterest migrated its iOS continuous integration and continuous delivery (CI/CD) pipeline from on-premises to reduce build failures by over 80 percent. Pinterest also uses machine learning (ML) to power its visual search tool Pinterest Lens that can recognize over 2.5 billion objects and match them to products. Today, Pinterest's exabyte data platform runs entirely on AWS, scaling log search and analytics to over 1.7TB while reducing operations costs by 30%. To meet these demands and enhance its infrastructure, Pinterest embarked on a migration journey to leverage Amazon Web Services (AWS) cloud services.



### CHALLENGES:

Before migrating to AWS, Pinterest encountered several challenges with its existing infrastructure:

- **Scalability:** Pinterest's user base was rapidly expanding, leading to increased demands on its infrastructure. Scaling their on-premises infrastructure to accommodate growth was costly and time-consuming.
- **Resource Efficiency:** Maintaining and managing physical servers required significant resources, both in terms of time and manpower. This hindered Pinterest's ability to focus on innovation and product development.
- **Global Reach:** Pinterest aimed to reach a global audience efficiently, requiring a scalable and geographically distributed infrastructure to reduce latency and ensure optimal user experience worldwide.

## **SOLUTIONS:**

Pinterest opted to migrate to AWS to address these challenges and leverage the benefits of cloud computing:

- 1. Elasticity:** AWS's Elastic Compute Cloud (EC2) enabled Pinterest to dynamically scale its compute capacity based on demand, ensuring optimal performance during peak traffic periods.
- 2. Managed Services:** Pinterest utilized AWS managed services such as Amazon RDS (Relational Database Service) and Amazon DynamoDB for database management, reducing the operational overhead associated with maintaining databases.
- 3. Content Delivery:** Leveraging AWS's Content Delivery Network (CDN) service, Amazon CloudFront, Pinterest accelerated content delivery to users worldwide, reducing latency and improving user experience.
- 4. Security and Compliance:** AWS provided robust security features and compliance certifications, ensuring that Pinterest's data and infrastructure remained secure and compliant with industry standards.
- 5. Cost Optimization:** By leveraging AWS's pay-as-you-go pricing model and utilizing services such as AWS Lambda for serverless computing, Pinterest optimized its infrastructure costs while maintaining performance and scalability.

## **RESULTS:**

The migration to AWS yielded significant benefits for Pinterest:

- 1. Scalability:** Pinterest achieved greater scalability and elasticity, seamlessly handling spikes in user traffic without sacrificing performance or reliability.
- 2. Operational Efficiency:** By offloading infrastructure management to AWS, Pinterest's engineering teams could focus more on product development and innovation, accelerating time-to-market for new features and improvements.
- 3. Global Reach:** With AWS's global infrastructure, Pinterest improved its global reach and reduced latency for users worldwide, enhancing the overall user experience.
- 4. Cost Savings:** Pinterest realized cost savings by optimizing its infrastructure usage, leveraging AWS's cost-effective services, and eliminating the need for upfront hardware investments.

## **CONCLUSION:**

Pinterest's migration to AWS enabled the company to overcome its infrastructure challenges while achieving greater scalability, efficiency, and global reach. By leveraging AWS's comprehensive suite of services, Pinterest enhanced its platform's performance, reliability, and security, ultimately delivering a superior user experience to millions of users worldwide. This case study exemplifies the transformative impact of cloud computing on businesses seeking to innovate and scale in today's digital landscape.

