

Lab 4: AWS Security and Server Infrastructure

A demo of the basic setup of this lab is at: <https://youtu.be/GaMd8MaqBXA>

A Outline

In previous labs we have set up a range of architectures with vSphere. This is a private cloud environment and creates infrastructure as a service. Increasingly we use the public cloud to build our information systems, and which reduces the investment in data centre costs, while providing the opportunity to quickly scale our server, network and data infrastructure. It is generally a pay-as-you-go model, and where we pay for CPU time, network bandwidth and data costs. The most popular cloud provider is AWS (Amazon Web Services), and which provides EC2 (for compute), S3 (for data buckets), RDS (for databases) and AWS Network Firewall (for firewalls). Some of these services are outlined in Figure 1.

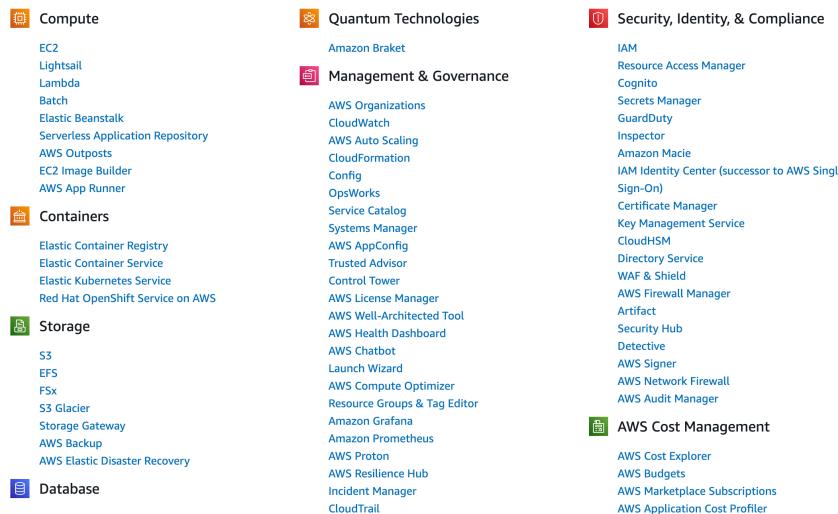


Figure 1: AWS Services

B Enabling your lab

You should have an AWS Academy login, so go to: <https://awsacademy.instructure.com/> and log into the system, and select **AWS Academy Learner Lab** (Figure 2).

AWS Academy Learner Lab - Foundation Services [28224]



This is the Edinburgh Napier University AWS Academy Learner Lab for CSN09112, and which has \$100 of credit for you to learn AWS Services. If you exceed your credit, you will lose your environment, so remember to switch your instances off when you are not using them. The lab session is four hours, but you can restart the lab again at any time. All EC2 instances will be shut off after the four hours are up. Do not end the lab, until we have finished with the module. Some resources are:

- Introduction to Cloud Computing: [here](#)
- A demo of setting up EC2 instances is [here](#).

[Click Modules](#) to start the course.

Figure 2: AWS Academy Learner Lab

Next, select “Modules”, and then “Learner Lab - Foundational Services”, and should have the lab environment (Figure 3).

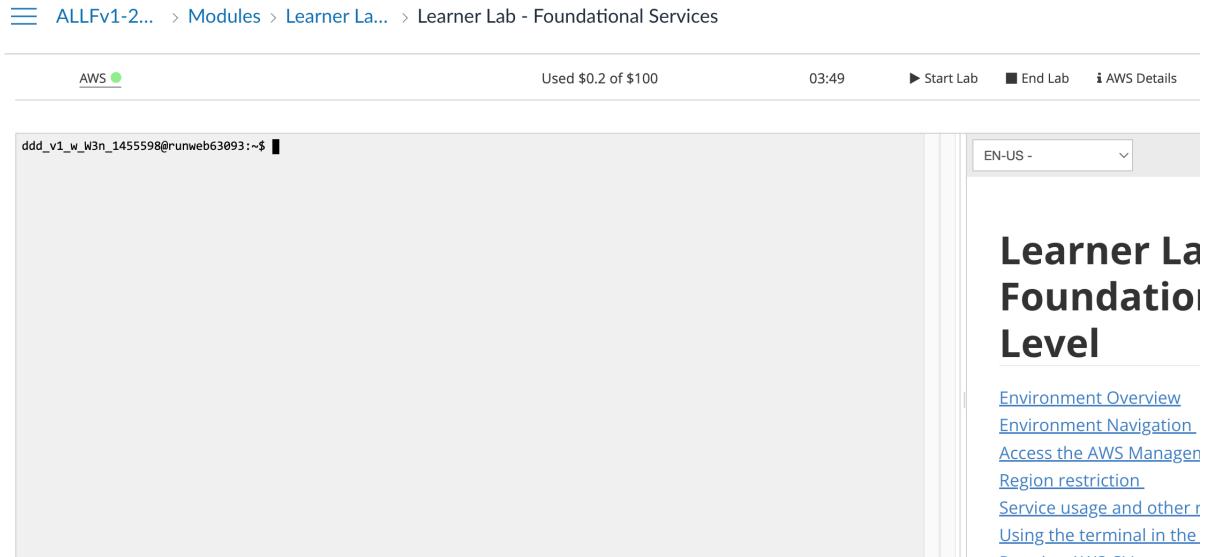


Figure 3: AWS Academy Learner Lab environment

In the console you can interact with your AWS though the console. Now, press the “Start Lab” button, and wait for the AWS light to go green. Once, green, you can click on it, and open up your AWS Management console. After this just select EC2, and you should see your EC2 environment.

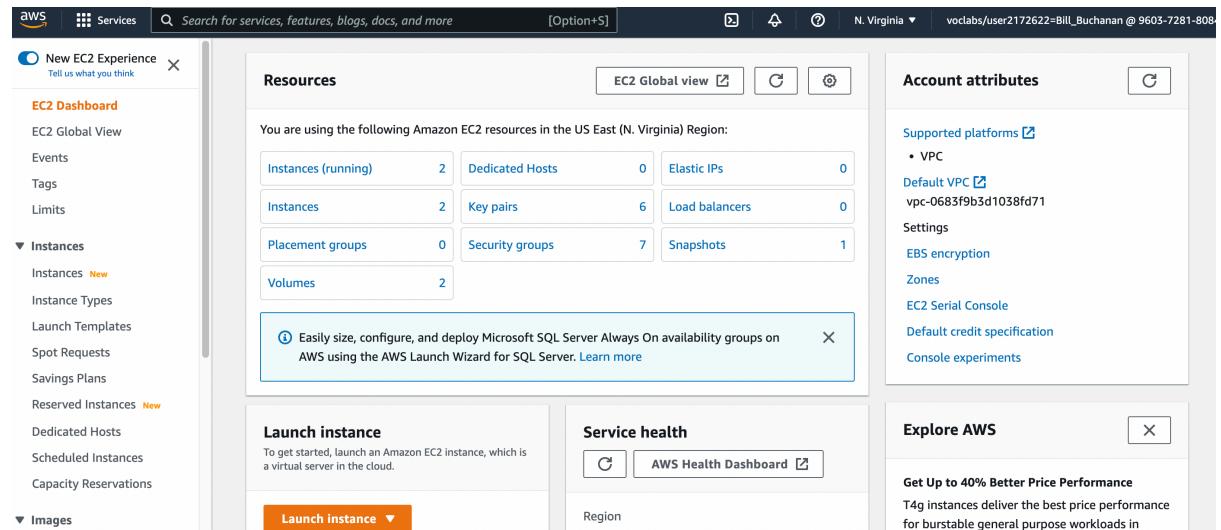


Figure 4: AWS Management Console

C Creating and Securing a Linux Server

We will now create a Linux Server, and which should be accessible from the Internet. For this select “Launch Instance”, and then give it a name (such as “My Linux Server”) and select the Amazon Linux instance for the AMI (Amazon Machine Instance) – as shown in Figure 5.

Name

Add additional tags

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type

Free tier eligible

Figure 5: Creating Amazon Linux instance

Now select **t2.micro** for the instance type.

How many vCPUs will the instance have?

How much memory will it have?

How much will it cost per day to run?

If you selected, t2.medium, how much would it cost per day?

If you selected, t2.large, how much would it cost per day?

Now create a new key pair and save it to your local drive. This file contains your private key, and which you will need to connect to your instance. Accept all the other defaults.

Observe the firewall group that will be applied.

Which firewall ports are open on the instance?

What is the main issue with this firewall setting?

How would you change it, once you have created the instance?

Observe the disk storage setting for the instance.

What type of disk will be used? [HDD/SSD]

What is the advantage of using SSD?

For disk storage, what is the size of the disk that you will create?

What is the maximum storage size for a free tier storage of the AMI instance we are creating?

C.1 Creating the instance

Go ahead and create the instance. Go back to the Management Console, and find your instance. Wait for it to set its state to running. Now we will connect to it. For this we need to create an SSH connection, and use the private key we have generated. The public key will be stored on the instance, and will authenticate our access. We do not need a username or password to access the instance, as this is often insecure. Our PEM file will give us access.

Now, we will examine the details of our instance (Figure 6). On the instance summary, determine the following:

The public IP address:

The private IP address:

The instance type:

The public IPv4 DNS:

From your local host, can you ping the public IP address? [Yes/No]

Why can't you successfully ping your instance?

Which region of the world is your instance running in?

C.2 Enabling ICMP on firewall

Now we will enable ICMP on the instance. First click on the Security tab of the instance summary, and then on the security group.

What is the firewall rule that is applied to the instance?

[SSH/Telnet/FTP/HTTP/HTTPs] for [0.0.0.0/0 or 0.0.0.0/8 or 0.0.0.0/16 or 0.0.0.0/32]

What does 0.0.0.0/0 represent?

Now go ahead and add an ICMP rule for all hosts (Figure 7).

Can you now successfully ping your instance? [Yes/No]

Now, lock your ICMP rule down to just your IP address (you need to use a /32 address for this). Can you still successfully ping the instance? [Yes/No]

Ask your neighbour or one of the lab tutors to ping your instance. Can they successfully ping it? [Yes/No]

What is the advantage of putting the firewall in AWS, rather than in the instance?

EC2 > Instances > i-07b0512e24e263766

Instance summary for i-07b0512e24e263766 (MyLinuxServer) [Info](#)

Updated less than a minute ago

Instance ID	Public IPv4 address	Private IPv4 addresses
i-07b0512e24e263766 (MyLinuxServer)	52.90.3.121 open address	172.31.16.186
IPv6 address	Instance state	Public IPv4 DNS
-	Pending	ec2-52-90-3-121.compute-1.amazonaws.com open address
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-172-31-16-186.ec2.internal	ip-172-31-16-186.ec2.internal	-
Answer private resource DNS name	Instance type	AWS Compute Optimizer finding
IPv4 (A)	t2.micro	Opt-in to AWS Compute Optimizer for recommendations.
Auto-assigned IP address	VPC ID	Learn more
52.90.3.121 [Public IP]	vpc-0683f9b3d1038fd71	
IAM Role	Subnet ID	Auto Scaling Group name
-	subnet-00bdb3e7927760f46	-

[Details](#) [Security](#) [Networking](#) [Storage](#) [Status checks](#) [Monitoring](#) [Tags](#)

► Instance details [Info](#)

Figure 6: Details of instance

Inbound rules | Outbound rules | Tags

ⓘ You can now check network connectivity with Reachability Analyzer [Run Reachability Analyzer](#) X

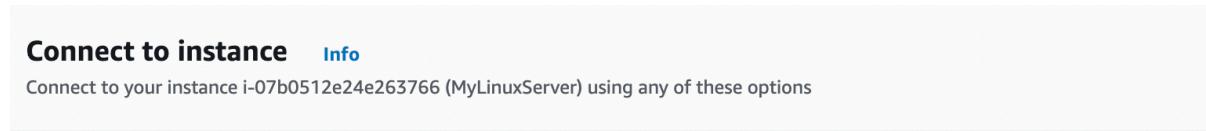
Inbound rules (2)

<input type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port
<input type="checkbox"/>	-	sgr-0ed01ab1ba175fe5b	IPv4	SSH	TCP	22
<input type="checkbox"/>	-	sgr-04b533407d759a...	IPv4	All ICMP - IPv4	ICMP	All

Figure 7: Enable ICMP

C.3 Accessing your instance

Now we will connect to our instance. For this you need SSH. This may be installed on the host you are using (such as in vSoC 2), or from Apps Anywhere. Once you have SSH, press Connect on your instance summary, and you should have tabs for Connect to instance (Figure 8). Next select the SSH client tab, and you will see the details of connecting to your instance with SSH.



Instance ID

[i-07b0512e24e263766](#) (MyLinuxServer)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is mynewkeypair.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 chmod 400 mynewkeypair.pem
4. Connect to your instance using its Public DNS:
 ec2-52-90-3-121.compute-1.amazonaws.com

Example:

ssh -i "mynewkeypair.pem" ec2-user@ec2-52-90-3-121.compute-1.amazonaws.com

Figure 8: Connect to instance

Now find your PEM file, and protect it with:

chmod myfile.pem

What protection does this put on your private key?

Next use the SSH connect with the name of your PEM file and with the DNS (or IP address) for your instance. For example, in the case in Figure 8, we have:

ssh -i "mynewkeypair.pem" ec2-user@ec2-52-90-3-121.compute-1.amazonaws.com

What is the name of the user that logs in?

An example of connecting is:

```
% ssh -i "mynewkeypair.pem" ec2-user@ec2-52-90-3-121.compute-1.amazonaws.com
The authenticity of host 'ec2-52-90-3-121.compute-1.amazonaws.com (52.90.3.121)' can't be established.
ED25519 key fingerprint is SHA256:/c5UOK6gprKL19XCptNQ1brb9MpYR5wEeqhd/6t+/wk.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:48: ec2-3-90-189-201.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-52-90-3-121.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Last login: Fri Sep 30 17:07:00 2022 from ec2-18-206-107-27.compute-1.amazonaws.com
```

```
__| | C __|_ / Amazon Linux 2 AMI  
__| \__|_|_  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-16-186 ~]$
```

Have you managed to connect? [Yes/No]

Is there a folder named .ssh? [Yes/No]

What is the purpose of the file contained in .ssh?

By using “ip addr show” or “ifconfig” in your instance, what is the private IP address of it?

Can you ping 8.8.8.8 from your instance? [Yes/No]

Now create a folder in the top level named “mytestfolder”, and put a new file in there named “mytext.txt”.

Now go to the EC2 Instance Connect, and press on the Connect button. You should now get a console terminal in the browser (Figure 9).

Verify that your file has been created. Has it been created in the instance? [Yes/No]

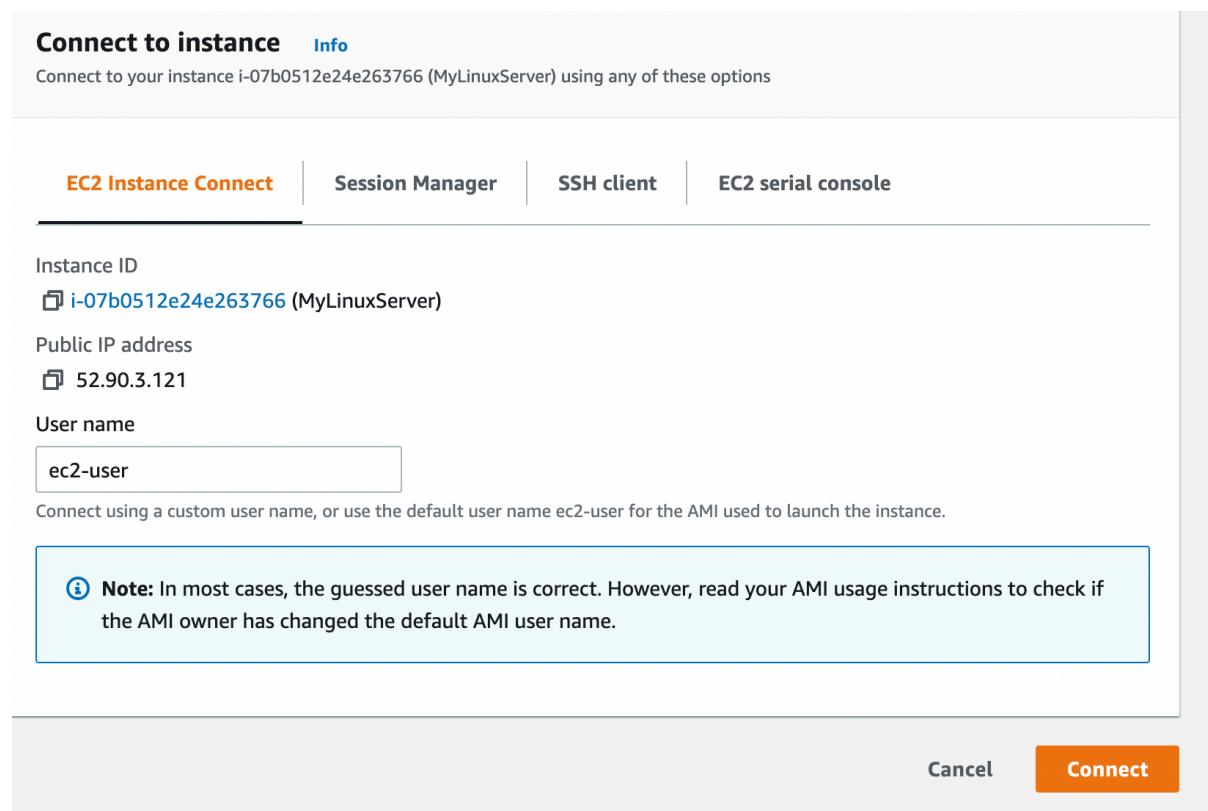


Figure 8: EC2 Instance Connect

```
Last login: Sun Oct  2 11:39:05 2022 from host86-131-160-187.range86-131.btcentralplus.com
[ec2-user@ip-172-31-16-186 ~]$ cd mytestfolder/
[ec2-user@ip-172-31-16-186 mytestfolder]$ ls
mytext.txt
[ec2-user@ip-172-31-16-186 mytestfolder]$ cat mytext.txt
Test
[ec2-user@ip-172-31-16-186 mytestfolder]$
```

Figure 9: EC2 Instance Connect terminal

Now examine the running services on the instance with:

```
$ netstat -i | grep tcp
$ netstat -i | grep udp
```

Which of the main services are running:

C.4 Installing a Web server

Now we will install a Web server on the instance with:

```
sudo yum update -y
sudo yum install -y httpd.x86_64
sudo systemctl start httpd.service
sudo systemctl enable httpd.service
```

Next open up a browser on your host and access your instance for Web access.

Can you connect to it? [Yes/No]

Why can't you connect to it?

Now enable a firewall rule on Port 80 and Port 443, and allow access (see Figure 10).

Inbound rules Info						
Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
sgr-0ed01ab1ba175fe5b	SSH	TCP	22	Custom ▾ <input type="text" value="0.0.0.0/0"/> X		Delete
sgr-04b533407d759a286	All ICMP - IPv4	ICMP	All	Anywh... ▾ <input type="text" value="0.0.0.0/0"/> X	Ping	Delete
-	HTTPS	TCP	443	Anywh... ▾ <input type="text" value="0.0.0.0/0"/> X	All access to Web server	Delete
-	HTTP	TCP	80	Anywh... ▾ <input type="text" value="0.0.0.0/0"/> X	All access to Web server	Delete

Figure 10: Enable HTTP and HTTPS rules

Can you now connect to your Web site? [Yes/No] (see Figure 11)

The screenshot shows a web browser window with the address bar displaying 'Not Secure | 52.90.3.121'. The main content area has a red header bar with the text 'Test Page'. Below the header, the page content reads: 'This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.' On the right side, there are two sections: 'If you are a member of the general public:' and 'If you are the website administrator:'. The 'general public' section contains instructions for reporting problems. The 'administrator' section contains instructions for adding content and preventing the page from being used. At the bottom right, there is a 'Powered by APACHE 2.4' logo.

Figure 11: Sample access to Web site

Now go into the /var/www folder, and create a file named “index.html”, and add:

```
<h1>Main web site</h1>
<p>Hello to you</p>
```

And then save the file.

Has it changed the welcome? [Yes/No]

C.6 Auditing

The main logging output is in the /var/log folder. Identify the contents of the following files:

What are the likely contents of the “secure” file?

What are the likely contents of the “boot.log” file?

List the log/httpd/access_log file. What are its contents? Can you identify your browser access? (see Figure 12). Which browser type accessed your Web server?

Now try with another browser type, and re-examine the log/httpd/access_log file. Did it detect the new browser type?

Now access a file that does not exist in your site (such as http://AWSIP/test.htm). Now re-examine the log/httpd/access_log file. What is the status code returned for the access?

```
e/105.0.0.0 Safari/537.36"
187 - - [02/Oct/2022:11:56:24 +0000] "GET /icons/apache_pb2.gif HTTP/1.1" 200 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36"
187 - - [02/Oct/2022:11:56:24 +0000] "GET /favicon.ico HTTP/1.1" 404 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36"
187 - - [02/Oct/2022:11:58:16 +0000] "-" 408 "-" "-"
187 - - [02/Oct/2022:12:12:22 +0000] "GET / HTTP/1.1" 200 13 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36"
187 - - [02/Oct/2022:12:19:53 +0000] "GET / HTTP/1.1" 200 13 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36"
187 - - [02/Oct/2022:12:19:53 +0000] "GET /favicon.ico HTTP/1.1" 404 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36"
187 - - [02/Oct/2022:12:19:54 +0000] "GET / HTTP/1.1" 304 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36"
187 - - [02/Oct/2022:12:19:56 +0000] "GET / HTTP/1.1" 304 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36"
```

Figure 12: Sample list of log/httpd/access_log

C.7 Accessing from AWS prompt

We can also access our instance from the AWS prompt. For this return to your AWS Academy console, and enter the command (Figure 13):

```
$ aws ec2 describe-instances
```

From the results, can you identify:

Instance type:

Public IP address:

Private IP address:

State:

```

ddd_v1_w_W3n_1455598@runweb63093:~$ aws ec2 describe-instances
{
    "Reservations": [
        {
            "Groups": [],
            "Instances": [
                {
                    "AmiLaunchIndex": 0,
                    "ImageId": "ami-026b57f3c383c2ec",
                    "InstanceId": "i-07b0512e24e263766",
                    "InstanceType": "t2.micro",
                    "KeyName": "mynewkeypair",
                    "LaunchTime": "2022-10-02T11:14:16+00:00",
                    "Monitoring": {
                        "State": "disabled"
                    },
                    "Placement": {
                        "AvailabilityZone": "us-east-1b",
                        "GroupName": "",
                        "Tenancy": "default"
                    },
                    "PrivateDnsName": "ip-172-31-16-186.ec2.internal",
                    "PrivateIpAddress": "172.31.16.186",
                    "ProductCodes": [],
                    "PublicDnsName": "ec2-52-90-3-121.compute-1.amazonaws.com",
                    "PublicIpAddress": "52.90.3.121",
                    "State": {
                        "Code": 16,
                        "Name": "running"
                    },
                    "StateTransitionReason": "",
                    "SubnetId": "subnet-00bdb2e7927760f46",
                    "VpcId": "vpc-0683f9b3d1038fd71",
                    "Architecture": "x86_64"
                }
            ]
        }
    ]
}

```

Lear Four Leve

[Environme](#)
[Environme](#)
[Access the](#)
[Region res](#)
[Service usa](#)
[Using the t](#)
[Running A\)](#)
[Using the /](#)
[Preservin](#)
[Accessing J](#)

Figure 13: Accessing instances

Now try we will stop our instance using an AWS EC2 command. Run the following with your instance ID (see Figure 14):

```
aws ec2 stop-instances --instance-ids [My-instance-ID]
```

From the AWS Management Console, has your instance stopped? [Yes/No]

```

ddd_v1_w_W3n_1455598@runweb62964:~$ aws ec2 stop-instances --instance-ids i-07b0512e24e263766
{
    "StoppingInstances": [
        {
            "CurrentState": {
                "Code": 64,
                "Name": "stopping"
            },
            "InstanceId": "i-07b0512e24e263766",
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
ddd_v1_w_W3n_1455598@runweb62964:~$ █

```

Figure 14: Stopping an instance

Now we will restart the instance, with:

```
aws ec2 start-instances --instance-ids [My-instance-ID]
```

Has the instance re-started? [Yes/No]

Now we will change the instance type from t3.micro to t3.small. To do this, run the following commands:

```
aws ec2 stop-instances --instance-ids [My-instance-ID]
aws ec2 wait instance-stopped --instance-ids [My-instance-ID]
aws ec2 modify-instance-attribute --instance-id [My-instance-ID] --instance-type "{\"Value\": \"t3.small\"}"
aws ec2 start-instances --instance-ids [My-instance-ID]
```

Did it change the instance type? [Yes/No]

Can you still get access to your instance?

By observing the script, and investigating what t3.micro and t3.small are, can you determine what has changed about your instance?

Now, revert the instance back to t3.micro, and suspend the instance.

D Creating and Securing a Windows 2022 Server

In this part of the lab we will create a Windows 2022 with t3.micro instance. First create a new instance, and give it a name, such as “MyWindowsServer” (Figure 15).

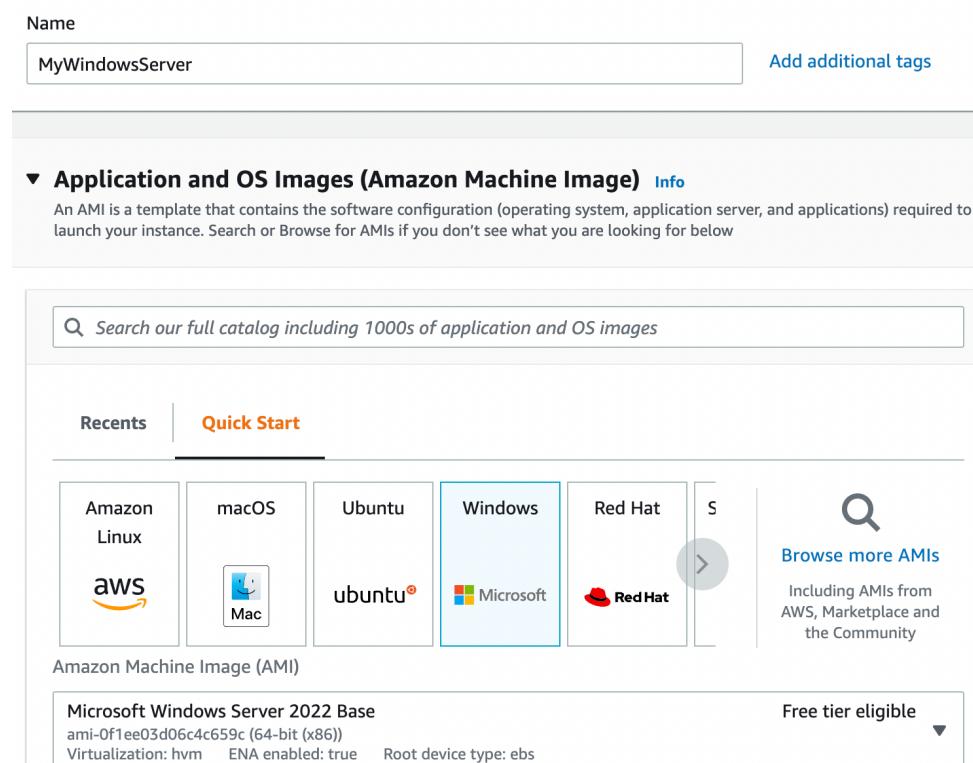


Figure 15: Creating Windows 2022 instance

Now select **t2.micro** for the instance type.

How many vCPUs will the instance have?

How much memory will it have?

How much will it cost per day to run?

If you selected, t2.medium, how much would it cost per day?

If you selected, t2.large, how much would it cost per day?

Now create a new key pair and save it to your local drive. This file contains your private key, and which you will need to connect to your instance. Accept all the other defaults.

Observe the firewall group that will be applied.

Which firewall ports are open on the instance?

What is the main issue with this firewall setting?

How would you change it, once you have created the instance?

Observe the disk storage setting for the instance.

What type of disk will be used? [HDD/SSD]

What is the advantage of using SSD?

For disk storage, what is the size of the disk that you will create?

What is the maximum storage size for a free tier storage of the AMI instance we are creating?

D.1 Creating the instance

Go ahead and create the instance. Go back to the Management Console, and find your instance. Wait for it to set its state to running. Now we will connect to it. For this we need to create an RDP connection, and use the private key we have generated to generate the initial password.

Now, we will examine the details of our instance (Figure 16). On the instance summary, determine the following:

The public IP address:

The private IP address:

The instance type:

The public IPv4 DNS:

From your local host, can you ping the public IP address? [Yes/No]

Why can't you successfully ping your instance?

Which region of the world is your instance running in?

C.2 Enabling ICMP on firewall

Now we will enable ICMP on the instance. First click on the Security tab of the instance summary, and then on the security group.

What is the firewall rule that is applied to the instance?

[SSH/RDP/Telnet/FTP/HTTP/HTTPs] for [0.0.0.0/0 or 0.0.0.0/8 or 0.0.0.0/16 or 0.0.0.0/32]

What does 0.0.0.0/0 represent?

Now go ahead and add an ICMP rule for all hosts (Figure 17).

Can you now successfully ping your instance? [Yes/No]

We will not be able to ping the instance yet, as the firewall on Windows is disabling it.

Instance summary for i-07d723258364f7172 (MyWindowsServer)		Info	Actions
Updated less than a minute ago		Copy	Connect
Instance ID	Public IPv4 address	Private IPv4 addresses	
i-07d723258364f7172 (MyWindowsServer)	54.83.154.0 open address	172.31.85.24	
IPv6 address	Instance state	Public IPv4 DNS	
-	Running	ec2-54-83-154-0.compute-1.amazonaws.com open address	
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses	
IP name: ip-172-31-85-24.ec2.internal	ip-172-31-85-24.ec2.internal	-	
Answer private resource DNS name	Instance type	AWS Compute Optimizer finding	
IPv4 (A)	t2.micro	Opt-in to AWS Compute Optimizer for recommendations.	
Auto-assigned IP address	VPC ID	Learn more	
54.83.154.0 [Public IP]	vpc-0683f9b3d1038fd71		
IAM Role	Subnet ID	Auto Scaling Group name	
-	subnet-07a3be17bcc59528b	-	

Figure 16: Details of instance

Inbound rules Info						
Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
sgr-0101b70fb7c795a97	RDP ▼	TCP	3389	Custom ▼ <input type="text" value="Q"/> <input type="button" value="0.0.0.0/0 X"/>	<input type="button" value="Delete"/>	
-	Custom ICMP - IPv4 ▼	All	All	Anywh... ▼ <input type="text" value="Q"/> <input type="button" value="Ping All"/> <input type="button" value="0.0.0.0 X"/>	<input type="button" value="Delete"/>	

Figure 17: Enable ICMP

D.3 Accessing your instance

Now we will connect to our instance. For this you need RDP. Next Connect to instance (Figure 18). Click on “Get password” and present your PEM file, and it should reveal the password (Figure 19).

Session Manager RDP client EC2 serial console

Instance ID
 i-07d723258364f7172 (MyWindowsServer)

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:

[Download remote desktop file](#)

When prompted, connect to your instance using the following details:

Public DNS

ec2-54-83-154-0.compute-1.amazonaws.com

User name

Administrator

Password [Get password](#)

Figure 18: Connect to instance

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:

 [Download remote desktop file](#)

When prompted, connect to your instance using the following details:

Public DNS

 ec2-54-83-154-0.compute-1.amazonaws.com

User name

 Administrator

Password

 7s8l-G?Jzzh;Lb8%KyGdmn6JwqReoRF-

Figure 19: Reveal password

Have you managed to connect? [Yes/No] (Figure 20)

By using “ipconfig” in your instance, what is the private IP address of it?

Can you ping 8.8.8.8 from your instance? [Yes/No]

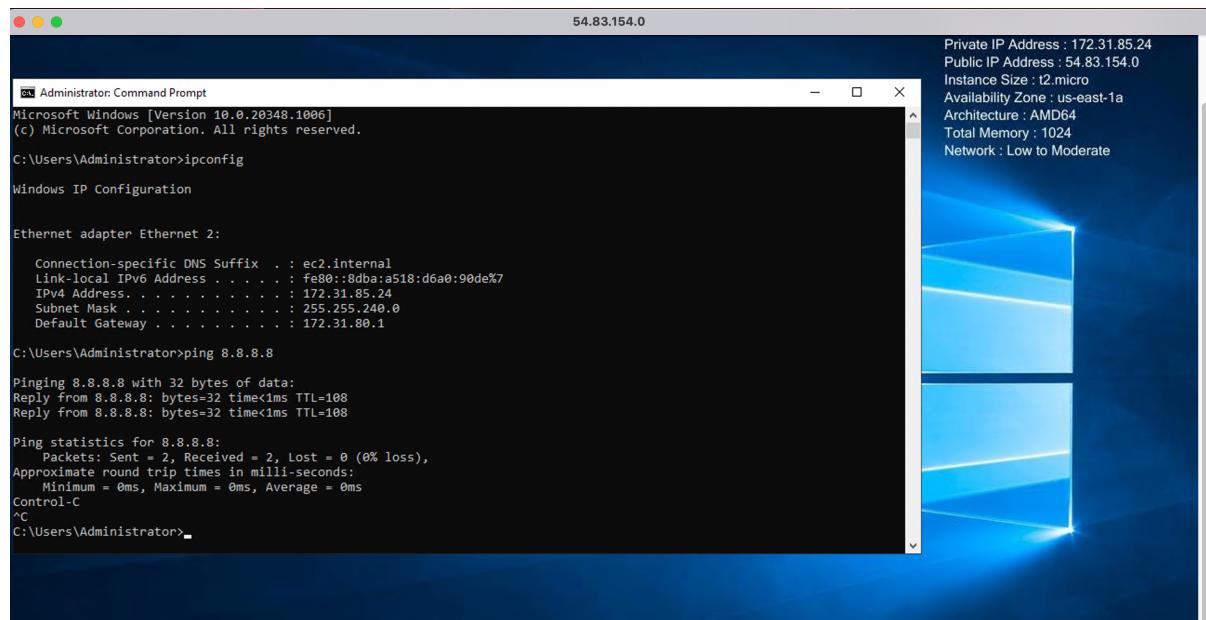


Figure 20: Windows 2022

D.4 Enable ICMP on instance

We have enabled the AWS firewall for ICMP. Now we will open-up ICMP in the instance. For this open-up with Advanced Windows firewall, and enable the rule for “File and Printer Sharing (ICMP-in) – as shown in Figure 21.

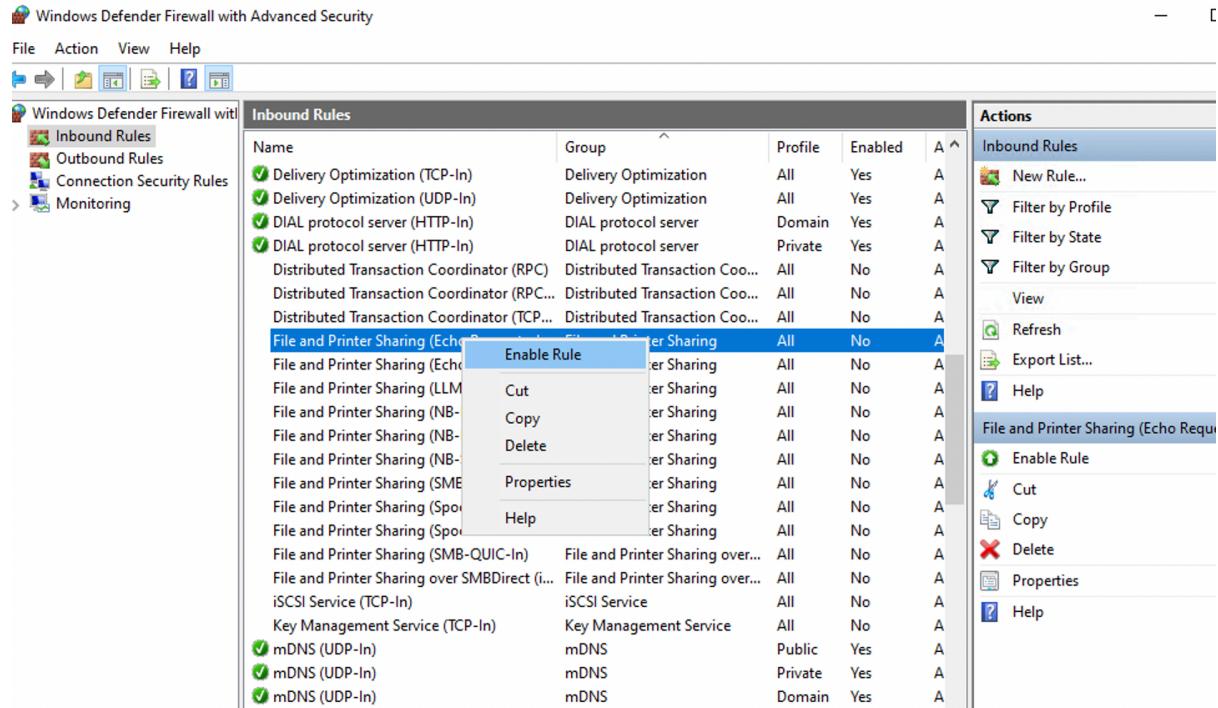


Figure 21: Enable ICMP

Can you successfully ping the instance from your instance? [Yes/No]

D.5 Show running services

Now examine the running services on the instance with:

```
$ netstat -i | grep tcp  
$ netstat -i | grep udp
```

Which of the main services are running:

D.5 Enable Web server

Now select Server Manage, and “Add a Role” for Web Server (IIS) (Figure 22).

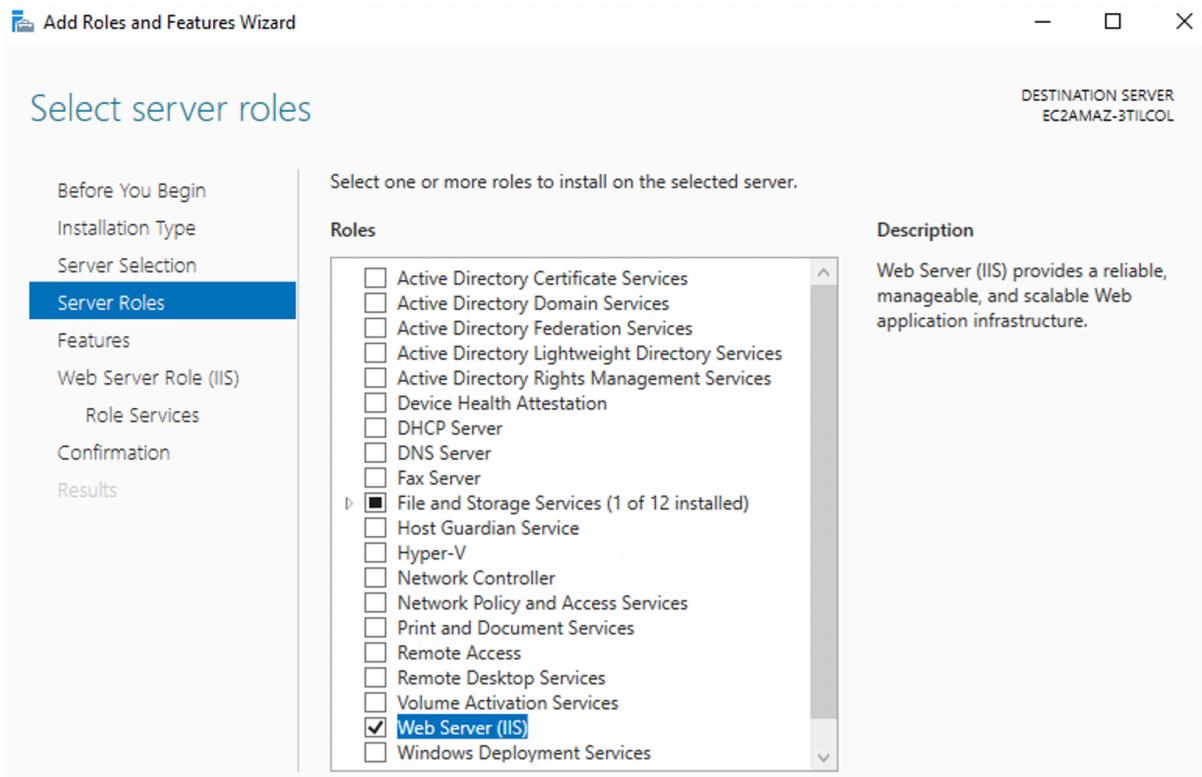


Figure 22: Enable ICMP

Now open a browser on the instance, and access <http://localhost>

Can you connect to the IIS Web server? [Yes/No]

Now open up your AWS firewall for Port 80 (Figure 22).

Inbound rules Info						
Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
sgr-088cf5bfd7b0741f4	All ICMP - IPv4	ICMP	All	Custom ▾	<input type="text"/> Ping All 0.0.0.0/0 X	Delete
sgr-0101b70fb7c795a97	RDP	TCP	3389	Custom ▾	<input type="text"/> 0.0.0.0/0 X	Delete
-	HTTP	TCP	80	Anywhere ▾	<input type="text"/> Web 0.0.0.0/0 X	Delete

Figure 21: Enable HTTP

Now open a browser on the instance, and access [http://\[IP of AWS\]](http://[IP of AWS])

Can you connect to the IIS Web server? [Yes/No] (Figure 23)

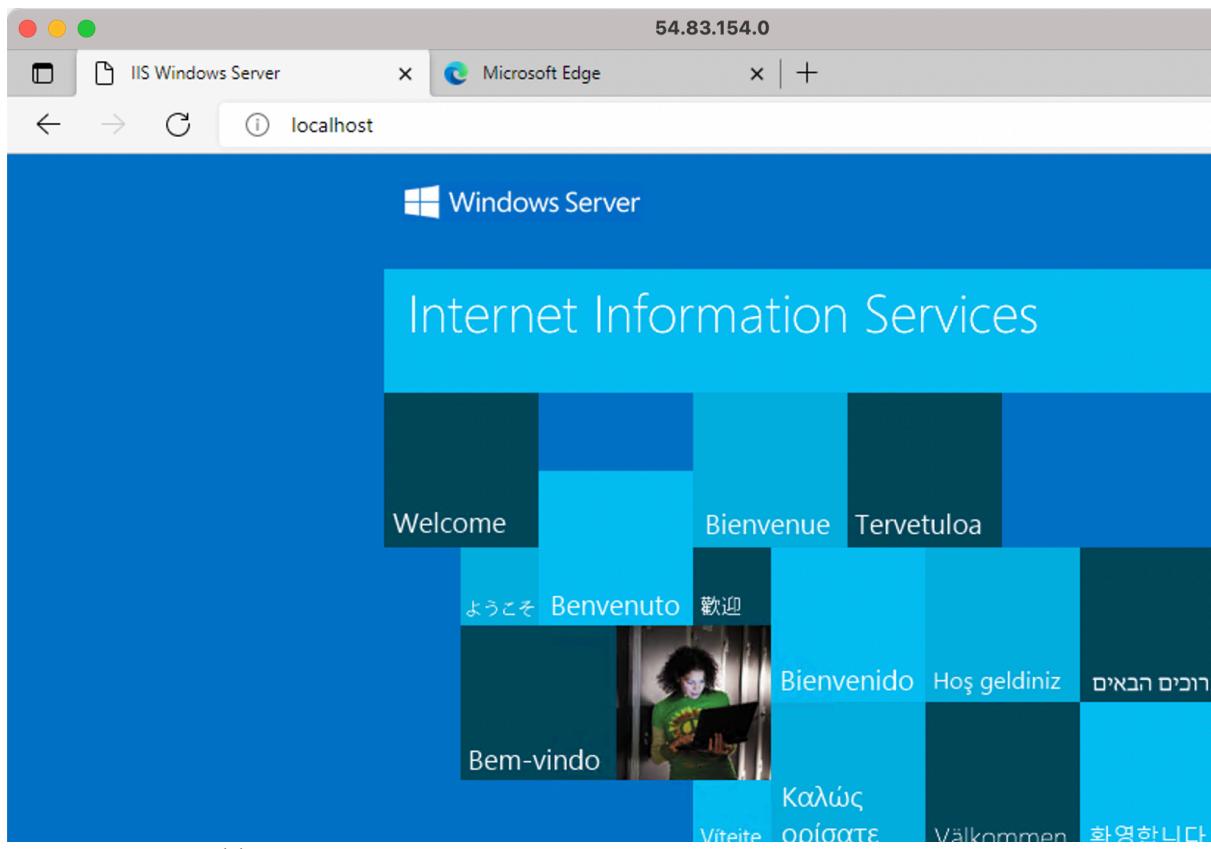


Figure 22: Local host

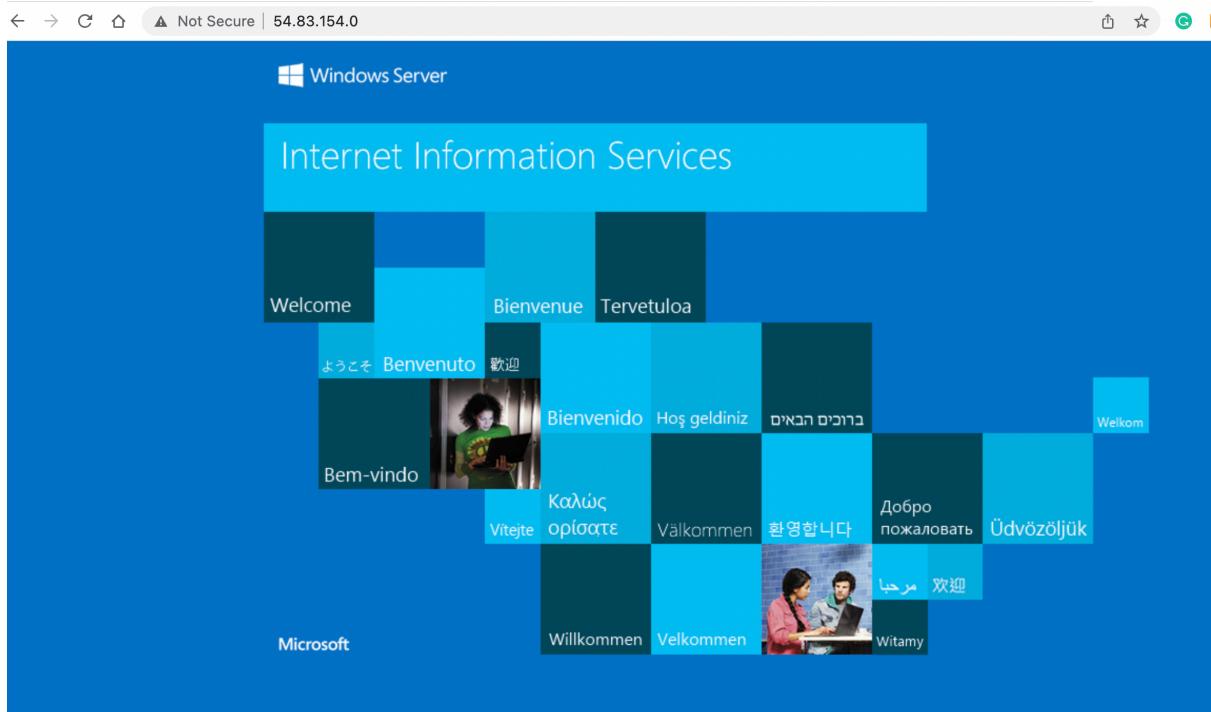


Figure 23: Remote access

Now go into the /inetpub/www folder, and create a file named “iisstart.html”, and add:

```
<h1>Main web site</h1>
<p>Hello to you</p>
```

And then save the file.

Has it changed the welcome? [Yes/No]

D.6 Auditing

The main logging output is in the “C:\inetpub\logs\LogFiles\W3SVC1” folder. Identify the contents of the following files:

Go into the “C:\inetpub\logs\LogFiles\W3SVC1” folder, and list the file in there. What are its contents? Can you identify your browser access? (see Figure 12). Which browser type accessed your Web server?

Now try with another browser type, and re-examine the log/httpd/access_log file. Did it detect the new browser type?

Now access a file that does not exist in your site (such as http://AWSIP/test.htm). Now re-examine the file. What is the status code returned for the access?