

PROJECT 1:

VPC Peering

Screenshot 1: VPCs list

Your VPCs (2/2)

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
project1_vpc1_ganeshs	vpc-0d0b40da3ce9e9d5f	Available	172.19.0.0/16	-
project1_vpc2_ganeshs	vpc-050930dfc7d11b151	Available	172.16.0.0/16	-

⇒ 2 VPC's Created: - “**project1_vpc1_ganeshs**” / “**project1_vpc2_ganeshs**”.

Your VPCs (1/2)

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
project1_vpc1_ganeshs	vpc-0d0b40da3ce9e9d5f	Available	172.19.0.0/16	-

Details

VPC ID vpc-0d0b40da3ce9e9d5f	State Available	DNS hostnames Disabled	DNS resolution Enabled
Tenancy Default	DHCP options set dopt-077efa6c	Route table rtb-029e86ee709992129 / project1_route1_ganeshs	Network ACL acl-078624a2acf976b1e
Default VPC No	IPv4 CIDR 172.19.0.0/16	IPv6 pool -	IPv6 CIDR -
Owner ID 355069791090			

⇒ VPC 1 Details: - “**project1_vpc1_ganeshs**”.

The screenshot shows the AWS VPC Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and user information (Ganesh_Subramanian, Ohio, Support). Below the header, a main content area displays a table titled "Your VPCs (1/2)". The table has one row, which is highlighted with a purple border. This row contains the following columns: a checkbox, the VPC name "project1_vpc2_ganeshs", the VPC ID "vpc-050930dfc7d11b151", the state "Available", and the IPv4 CIDR "172.16.0.0/16". To the right of the table is a "Details" panel, also outlined in purple, containing the following configuration details:

Details			
VPC ID vpc-050930dfc7d11b151	State Available	DNS hostnames Disabled	DNS resolution Enabled
Tenancy Default	DHCP options set dopt-077efa6c	Route table rtb-0b61d18453f3fb356 / project1_route2_ganeshs	Network ACL acl-06eb1c948ed2a12ca
Default VPC No	IPv4 CIDR 172.16.0.0/16	IPv6 pool -	IPv6 CIDR -
Owner ID 555069791090			

At the bottom of the page, there are links for Feedback, English (US), and a copyright notice: © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. There are also links for Privacy Policy and Terms of Use.

⇒ VPC 2 Details: - “**project1_vpc2_ganeshs**”.

Screenshot 2: IGW list

The screenshot shows the AWS VPC Management console with the 'Internet gateways' list. There are two entries:

Name	Internet gateway ID	State	VPC ID	Owner
project1_igw1_ganeshs	igw-0b7b3f8d9ea5d5087	Attached	vpc-0d0b40da3ce9e9d5f project1_vpc1_ganeshs	355069791090
project1_igw2_ganeshs	igw-0cce96790cfde806a	Attached	vpc-050930dfc7d11b151 project1_vpc2_ganeshs	355069791090

⇒ 2 IGW's Created: - “**project1_igw1_ganeshs**” / “**project1_igw2_ganeshs**”.

Screenshot 3: Edit Route List

The screenshot shows the AWS VPC Management console with the 'Route Tables' list. There are four entries:

Name	Route Table ID	Expl	Edge	Main	VPC ID	Owner
project1_route2_ganeshs	rtb-0b61d18453f3fb356	-	-	Yes	vpc-050930dfc7d11b151 project1_vpc2_ganeshs	355069791090
project1_route2_ganeshs	rtb-0200230c0edd0aab4	-	-	No	vpc-050930dfc7d11b151 project1_vpc2_ganeshs	355069791090
project1_route1_ganeshs	rtb-080e86ee709992129	-	-	No	vpc-0d0b40da3ce9e9d5f project1_vpc1_ganeshs	355069791090
project1_route1_ganeshs	rtb-029e86ee709992129	-	-	Yes	vpc-0d0b40da3ce9e9d5f project1_vpc1_ganeshs	355069791090

⇒ 2 Route Table Created: - “**project1_route1_ganeshs**” / “**project1_route2_ganeshs**”.

The screenshot shows the AWS VPC Management console with the URL <https://us-east-2.console.aws.amazon.com/vpc/home?region=us-east-2#RouteTables:search=project1:sort=vpclid>. The search bar contains "project1". The main table lists a single route table named "project1_route1_ganeshs" with the ID "rtb-029e86ee709992129". The "Routes" tab is selected, showing two routes:

Destination	Target	Status	Propagated
172.19.0.0/16	local	active	No
0.0.0.0/0	igw-0b7b3f8d9ea5d5087	active	No

⇒ Route Table 1 Details: - "**project1_route1_ganeshs**".

The screenshot shows the AWS VPC Management console with the URL <https://us-east-2.console.aws.amazon.com/vpc/home?region=us-east-2#RouteTables:search=project1:sort=vpclid>. The search bar contains "project1". The main table lists a single route table named "project1_route2_ganeshs" with the ID "rtb-0b61d18453f3fb356". The "Routes" tab is selected, showing two routes:

Destination	Target	Status	Propagated
172.16.0.0/16	local	active	No
0.0.0.0/0	igw-0cce96790cfde806a	active	No

⇒ Route Table 2 Details: - "**project1_route2_ganeshs**".

Screenshot 4: Subnet list

The screenshot shows the AWS VPC Management Console with the URL <https://us-east-2.console.aws.amazon.com/vpc/home?region=us-east-2#subnets:sort=desc:tag:Name>. The 'Create subnet' button is highlighted. A search bar at the top says 'Filter by tags and attributes or search by keyword'. Below it is a table with columns: Name, Subnet ID, State, VPC, IPv4 CIDR, Availability Zone, and Availability. Two subnets are listed:

Name	Subnet ID	State	VPC	IPv4 CIDR	Availability Zone	Availability
project1_subnet2_ganeshs	subnet-001f7bde433b2b895	available	vpc-050930dfc7d11b151 project1_vpc2_ganeshs	172.16.16.0/24	250 - us-east-2a	use2-a
project1_subnet1_ganeshs	subnet-0185501ca6ef52791	available	vpc-0d0b40da3ce9e9d5f project1_vpc1_ganeshs	172.19.19.0/24	250 - us-east-2b	use2-a

Subnets: subnet-001f7bde433b2b895, subnet-0185501ca6ef52791

⇒ 2 Subnet's Created: - “**project1_subnet1_ganeshs**” / “**project1_subnet2_ganeshs**”.

The screenshot shows the AWS VPC Management Console with the URL <https://us-east-2.console.aws.amazon.com/vpc/home?region=us-east-2#subnets:sort=desc:tag:Name>. The 'Create subnet' button is highlighted. A search bar at the top says 'Filter by tags and attributes or search by keyword'. Below it is a table with columns: Name, Subnet ID, State, VPC, IPv4 CIDR, Availability Zone, and Availability. One subnet is listed:

Name	Subnet ID	State	VPC	IPv4 CIDR	Availability Zone	Availability
project1_subnet1_ganeshs	subnet-0185501ca6ef52791	available	vpc-0d0b40da3ce9e9d5f project1_vpc1_ganeshs	172.19.19.0/24	250 - us-east-2b	use2-a

Subnet: subnet-0185501ca6ef52791

The 'Description' tab is selected. A detailed table shows the subnet configuration:

Setting	Value
Subnet ID	subnet-0185501ca6ef52791
VPC	vpc-0d0b40da3ce9e9d5f project1_vpc1_ganeshs
Available IPv4 Addresses	250
Availability Zone	us-east-2b (use2-a2)
Network ACL	acl-078624a2acf976b1e
Auto-assign public IPv4 address	Yes
Customer-owned IPv4 pool	-
Outpost ID	-
State	available
IPv4 CIDR	172.19.19.0/24
IPv6 CIDR	-
Route Table	rtb-029e86ee709992129 project1_route1_ganeshs
Default subnet	No
Auto-assign customer-owned IPv4 address	No
Auto-assign IPv6 address	No
Owner	355069791090

⇒ Subnet 1 Details: - “**project1_subnet1_ganeshs**”.

The screenshot shows the AWS VPC Management Console with the Subnets page. A specific subnet, "project1_subnet2_ganeshs", is selected and highlighted with a purple border. The subnet details are displayed in a modal window below the table.

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6	Availability Zone	Available
project1_subnet2_ganeshs	subnet-001f7bde433b2b895	available	vpc-050930dfc7d11b151 project1_vpc2_ganeshs	172.16.16.0/24	-	us-east-2a	use2-a

Subnet: subnet-001f7bde433b2b895

Description (selected tab) **Flow Logs** **Route Table** **Network ACL** **Tags** **Sharing**

Subnet ID: subnet-001f7bde433b2b895	State: available
VPC: vpc-050930dfc7d11b151 project1_vpc2_ganeshs	IPv4 CIDR: 172.16.16.0/24
Available IPv4 Addresses: 250	IPv6 CIDR: -
Availability Zone: us-east-2a (use2-az1)	Route Table: rtb-0b61d18453f3fb356 project1_route2_ganeshs
Network ACL: acl-06eb1c948ed2a12ca	Default subnet: No
Auto-assign public IPv4 address: Yes	Auto-assign customer-owned: No
Customer-owned IPv4 pool: -	IPV4 address: No
Outpost ID: -	Auto-assign IPv6 address: No
	Owner: 355069791090

⇒ Subnet 2 Details: - “**project1_subnet2_ganeshs**”.

Screenshot 5: Instance details

The screenshot shows the AWS EC2 Management Console Instances page. A purple box highlights the 'Instances (2)' link in the left sidebar. Below it, a table lists two EC2 instances:

Name	Instance ID	Instance state	Status check	Availability zone	Public IPv4
project1_instance2_ganeshs	i-0b1fd9ce09696e879	Running	2/2 checks passed	us-east-2a	3.17.184.173
project1_instance1_ganeshs	i-00ea42176b4982c60	Running	2/2 checks passed	us-east-2b	3.21.156.12

⇒ 2 EC2 Instance's: - “**project1_instance1_ganeshs**” / “**project1_instance2_ganeshs**”.

The screenshot shows the AWS EC2 Management Console Instances page. A purple box highlights the 'Instances (1/2)' link in the left sidebar. Below it, a table lists one EC2 instance, which is then selected (indicated by a checked checkbox). The 'Instance summary' section is expanded, showing detailed information:

Instance ID	Public IPv4 address	Private IPv4 addresses
i-00ea42176b4982c60 (project1_instance1_ganeshs)	3.21.156.12 open address	172.19.19.229
Instance state	Public IPv4 DNS	Private IPv4 DNS
Running	-	ip-172-19-19-229.us-east-2.compute.internal
Instance type	Elastic IP addresses	VPC ID
t2.micro	-	vpc-0d0b40da3ce9e9d5f (project1_vpc1_ganeshs)
IAM Role	Subnet ID	
-	subnet-0185501ca6ef52791 (project1_subnet1_ganeshs)	

⇒ EC2 Instance 1 Details: - “**project1_instance1_ganeshs**”.

The screenshot shows the AWS EC2 Management Console Instances page. A purple box highlights the 'Instances (1/2)' header. Another purple box highlights the first instance in the list, which is selected. Below the table, a detailed view of the instance is shown in a modal window with a purple border. This modal contains the 'Instance summary' section with the following details:

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0b1fd9ce09696e879 (project1_instance2_ganeshs)	3.17.184.173 open address	172.16.16.153
Instance state	Public IPv4 DNS	Private IPv4 DNS
Running	-	ip-172-16-16-153.us-east-2.compute.internal
Instance type	Elastic IP addresses	VPC ID
t2.micro	-	vpc-050930dfc7d11b151 (project1_vpc2_ganeshs)
IAM Role	Subnet ID	
-	subnet-001f7bde433b2b895 (project1_subnet2_ganeshs)	

⇒ EC2 Instance 2 Details: - “**project1_instance2_ganeshs**”.

Screenshot 6: Success Public

```
C:\Users\Administrator>
C:\Users\Administrator> ping 3.21.156.12
Ping statistics for 3.21.156.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 299ms, Maximum = 368ms, Average = 325ms
C:\Users\Administrator>

Administrator: C:\WINDOWS\system32\cmd.exe
Recycle Bin
Administrator: Command Prompt
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix . : us-east-2.compute.internal
    Link-local IPv6 Address . . . . . : fe80::d132:cb9c:589e:1ee6%5
    IPv4 Address . . . . . : 172.19.19.129
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 172.19.19.1
C:\Users\Administrator>
```

⇒ Ping from **Base Machine** to RDP1 (“**project1_instance1_ganeshs**”: - “**3.21.156.12**”) (**Public IP**).

```
C:\Users\Administrator>
C:\Users\Administrator> ping 3.17.184.173
Ping statistics for 3.17.184.173:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 284ms, Maximum = 364ms, Average = 312ms
C:\Users\Administrator>

Administrator: C:\WINDOWS\system32\cmd.exe
Recycle Bin
Administrator: Command Prompt
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix . : us-east-2.compute.internal
    Link-local IPv6 Address . . . . . : fe80::98ea:91a1:f09b:90f8%4
    IPv4 Address . . . . . : 172.16.16.153
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 172.16.16.1
C:\Users\Administrator>
```

⇒ Ping from **Base Machine** to RDP2 (“**project1_instance2_ganeshs**”: “**3.17.184.173**”) (**Public IP**).

Two windows titled "project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection" and "project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection". Both show Command Prompt windows running ipconfig and ping commands.

Left Window (RDP1):

```
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::d132:cb9c:589e:1ee6%5
  IPv4 Address . . . . . : 172.19.19.29
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.19.19.1

C:\Users\Administrator>ping 3.17.184.173
Pinging 3.17.184.173 with 32 bytes of data:
Reply from 3.17.184.173: bytes=32 time=1ms TTL=127

Ping statistics for 3.17.184.173:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\Administrator>
```

Right Window (RDP2):

```
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::98ea:91a1:f09b:90f8%4
  IPv4 Address . . . . . : 172.16.16.153
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.16.16.1

C:\Users\Administrator>
```

⇒ Ping from RDP1 to RDP2 ("project1_instance2_ganeshs": - "3.17.184.173") (Public IP).

Two windows titled "project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection" and "project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection". Both show Command Prompt windows running ipconfig and ping commands.

Left Window (RDP2):

```
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::d132:cb9c:589e:1ee6%5
  IPv4 Address . . . . . : 172.19.19.29
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.19.19.1

C:\Users\Administrator>
```

Right Window (RDP1):

```
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::98ea:91a1:f09b:90f8%4
  IPv4 Address . . . . . : 172.16.16.153
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.16.16.1

C:\Users\Administrator>ping 3.21.156.12
Pinging 3.21.156.12 with 32 bytes of data:
Reply from 3.21.156.12: bytes=32 time=1ms TTL=127

Ping statistics for 3.21.156.12:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\Administrator>
```

⇒ Ping from RDP2 to RDP1 ("project1_instance1_ganeshs": - "3.21.156.12") (Public IP).

RTO private IP: -

The screenshot shows two windows side-by-side. The left window is a Command Prompt window titled 'Administrator: C:\WINDOWS\system32\cmd.exe'. It displays a series of 'C:\Users\Administrator>' prompts. A yellow box highlights the command 'ping 172.19.19.229' and its output, which shows four 'request timed out' messages. The right window is titled 'project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection'. It contains a Command Prompt window titled 'Administrator: Command Prompt' with the command 'ipconfig'. A yellow box highlights the 'IPv4 Address' entry, which is 172.19.19.229. Below it, the subnet mask is 255.255.255.0 and the default gateway is 172.19.19.1.

⇒ Ping from **Base Machine** to RDP1("project1_instance1_ganeshs": - "172.19.19.229") (**Private IP**).

The screenshot shows two windows side-by-side. The left window is a Command Prompt window titled 'Administrator: C:\WINDOWS\system32\cmd.exe'. It displays a series of 'C:\Users\Administrator>' prompts. A yellow box highlights the command 'ping 172.16.16.153' and its output, which shows four 'request timed out' messages. The right window is titled 'project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection'. It contains a Command Prompt window titled 'Administrator: C:\Windows\system32\cmd.exe' with the command 'ipconfig'. A yellow box highlights the 'IPv4 Address' entry, which is 172.16.16.153. Below it, the subnet mask is 255.255.255.0 and the default gateway is 172.16.16.1.

⇒ Ping from **Base Machine** to RDP2("project1_instance2_ganeshs": - "172.16.16.153") (**Private IP**).

The screenshot shows two separate Remote Desktop sessions. The left window is titled "project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection" and the right window is titled "project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection". Both windows show Command Prompt windows running ipconfig and ping commands.

```

Administrator: Command Prompt
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . fe80::d132:ch9c%589e:1ee6%
  IPv4 Address . . . . . 172.19.19.29
  Subnet Mask . . . . . 255.255.255.0
  Default Gateway . . . . . 172.19.19.1

C:\Users\Administrator>ping 172.16.16.153
Pinging 172.16.16.153 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 172.16.16.153:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\Users\Administrator>

Administrator: Command Prompt
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . fe80::98ea:91a1:f09b:90f8%4
  IPv4 Address . . . . . 172.16.16.153
  Subnet Mask . . . . . 255.255.255.0
  Default Gateway . . . . . 172.16.16.1

C:\Users\Administrator>ping 172.19.19.229
Pinging 172.19.19.229 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 172.19.19.229:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\Users\Administrator>

```

⇒ Ping from **RDP1** to **RDP2** (“**project1_instance2_ganeshs**”) and Vice versa. (**Private IP**).

Screenshot 7: Peering with Requestor and Acceptor

The screenshot shows the "Peering Connections" page in the AWS VPC console. A blue box highlights the "Create Peering Connection" button. Another blue box highlights the table where a new peering connection named "project1_peer_ganeshs" is listed. A third blue box highlights the details of this connection, showing it is active and connecting two VPCs with specific CIDR ranges.

Name	Requester VPC	Acceptor VPC	Requester CIDRs	Acceptor CIDRs
project1_peer_ganeshs	vpc-0d0b40da3ce9e9d5f project1_vpc1_ganeshs	vpc-050930dfc7d11b151 project1_vpc2_ganeshs	172.19.0.0/16	172.16.0.0/16

Peering Connection: pcx-03a4a13b9e4de7c9b

Description	DNS	Route Tables	Tags
Requester VPC owner: 355069791090 Requester VPC ID: vpc-0d0b40da3ce9e9d5f Requester VPC Region: Ohio (us-east-2) Requester VPC CIDRs: 172.19.0.0/16 VPC Peering Connection: pcx-03a4a13b9e4de7c9b Expiration time: -			Acceptor VPC owner: 355069791090 Acceptor VPC ID: vpc-050930dfc7d11b151 Acceptor VPC Region: Ohio (us-east-2) Acceptor VPC CIDRs: 172.16.0.0/16 Peering connection status: Active

⇒ Peering Connection Created: - “**project1_peer_ganeshs**” where the **Requestor** is “**project1_vpc1_ganeshs**” and **Acceptor** is “**project1_vpc2_ganeshs**”.

Screenshot 8: Success for Private

The screenshot shows two separate Remote Desktop sessions. The left session, titled "project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection", has its Command Prompt window open. It runs the command "ipconfig" to show network configuration. The right session, titled "project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection", also has its Command Prompt window open, running the same "ipconfig" command. A yellow arrow points from the highlighted "IPv4 Address" in the left session's output to the highlighted "IPv4 Address" in the right session's output, indicating they are the same private IP address (172.16.16.153). Both sessions then run a "ping" command to the other instance's IP address (172.16.16.153), which succeeds with 0% loss.

```

project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection
Administrator: Command Prompt
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::d132:c9c:589e:1ee6%5
  IPv4 Address . . . . . : 172.19.19.229
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.19.19.1
C:\Users\Administrator>ping 172.16.16.153
Pinging 172.16.16.153 with 32 bytes of data:
Reply from 172.16.16.153: bytes=32 time=1ms TTL=128

Ping statistics for 172.16.16.153:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms
C:\Users\Administrator>

project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection
Administrator: C:\Windows\system32\cmd.exe
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::98ea:91a1:f09b:90f8%4
  IPv4 Address . . . . . : 172.16.16.153
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.16.16.1
C:\Users\Administrator>

```

⇒ Ping from RDP1 to RDP2 ("project1_instance2_ganeshs": - 172.16.16.153) (Private IP).

The screenshot shows two separate Remote Desktop sessions. The left session, titled "project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection", has its Command Prompt window open. It runs the command "ipconfig" to show network configuration. The right session, titled "project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection", also has its Command Prompt window open, running the same "ipconfig" command. A yellow arrow points from the highlighted "IPv4 Address" in the left session's output to the highlighted "IPv4 Address" in the right session's output, indicating they are the same private IP address (172.19.19.229). Both sessions then run a "ping" command to the other instance's IP address (172.19.19.229), which succeeds with 0% loss.

```

project1_instance1_ganeshs - 3.21.156.12 - Remote Desktop Connection
Administrator: Command Prompt
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::d132:c9c:589e:1ee6%5
  IPv4 Address . . . . . : 172.19.19.229
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.19.19.1
C:\Users\Administrator>

project1_instance2_ganeshs - 3.17.184.173 - Remote Desktop Connection
Administrator: C:\Windows\system32\cmd.exe
C:\Users\Administrator>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . : us-east-2.compute.internal
  Link-local IPv6 Address . . . . . : fe80::98ea:91a1:f09b:90f8%4
  IPv4 Address . . . . . : 172.16.16.153
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 172.16.16.1
C:\Users\Administrator>ping 172.19.19.229
Pinging 172.19.19.229 with 32 bytes of data:
Reply from 172.19.19.229: bytes=32 time=1ms TTL=128

Ping statistics for 172.19.19.229:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms
C:\Users\Administrator>

```

⇒ Ping from RDP2 to RDP1 ("project1_instance1_ganeshs": - 172.19.19.229) (Private IP).

PROJECT 2:

IAM

Task 1: Creating users without permissions-IAM password policy check.

Screenshot 1: user summary with all tab information

The screenshot shows the IAM Management Console for a user named 'ganeshsuser'. The 'Permissions' tab is active, displaying a single policy named 'IAMUserChangePassword' which is an 'AWS managed policy'. Other tabs like 'Groups', 'Tags', 'Security credentials', and 'Access Advisor' are also present. The left sidebar lists various IAM management options such as Groups, Roles, Policies, and Groups.

⇒ IAM User Summary “ganeshsuser” with Permission Details.

The screenshot shows the IAM Management Console for the same user 'ganeshsuser'. The 'Groups' tab is active, indicating 'No results'. Other tabs like 'Permissions', 'Tags', 'Security credentials', and 'Access Advisor' are visible. The left sidebar shows the same set of IAM management options as the previous screenshot.

⇒ IAM User Summary “ganeshsuser” with Group Details.

The screenshot shows the AWS IAM Management Console interface. The left sidebar is titled "Identity and Access Management (IAM)" and includes sections for Dashboard, Access management (Groups, Users, Roles, Policies, Identity providers, Account settings), and Access reports (Access analyzer, Archive rules, Analyzers, Settings). The main content area displays the details for a user named "ganeshsuser". The top navigation bar shows the User ARN as "arn:aws:iam::355069791090:user/ganeshsuser", Path as "/", and Creation time as "2020-10-22 06:46 UTC+0530". Below this, tabs for "Permissions", "Groups", "Tags", "Security credentials" (which is selected and highlighted with a purple border), and "Access Advisor" are visible. The "Security credentials" section is expanded, showing "Sign-in credentials" and "Access keys". Under "Sign-in credentials", there is a summary link, a console password status (Enabled, last signed in Today), an assigned MFA device (Not assigned), and signing certificates (None). Under "Access keys", a table lists one key: AKIAVFK6PBNZA5QL6HNT, created on 2020-10-22 06:46 UTC+0530, with N/A for last used, and an active status. A "Create access key" button is also present.

⇒ **IAM User Summary “ganeshsuser” with Security Credentials Details.**

Task 2: Creating users without the IAM password policy.

Screenshot 2: user summary with all tab information

The screenshot shows the AWS IAM Management Console for a user named 'nexususer'. The 'Permissions' tab is active. A callout points to the 'Get started with permissions' section, which says: 'This user doesn't have any permissions yet. Get started by adding the user to a group, copying permissions from another user, or attaching a policy directly.' It includes a 'Learn more' link and 'Add permissions' and 'Add inline policy' buttons. Other tabs visible are 'Groups', 'Tags', 'Security credentials', and 'Access Advisor'. The left sidebar shows navigation options like Dashboard, Access management (Groups, Users, Roles, Policies), Identity providers, Account settings, Access reports (Archive rules, Analyzers, Settings), Credential report, and Organization activity. The top right shows the user's name 'Ganesh_Subramanian' and navigation links for Global, Support, and Help.

⇒ IAM User Summary “**nexususer**” with **Permission Details**.

The screenshot shows the AWS IAM Management Console for the same user 'nexususer'. The 'Groups' tab is active. A callout points to the 'Add user to groups' button. Below it is a table with 'Group name' and 'Attached permissions' columns, both showing 'No results'. Other tabs include 'Permissions', 'Tags', 'Security credentials', and 'Access Advisor'. The left sidebar and top navigation are identical to the previous screenshot.

⇒ IAM User Summary “**nexususer**” with **Group Details**.

User ARN: arn:aws:iam::355069791090:user/nexususer

Path: /

Creation time: 2020-10-22 08:00 UTC+0530

Sign-in credentials

Summary	Console sign-in link: https://355069791090.signin.aws.amazon.com/console
Console password	Enabled (last signed in Today) Manage
Assigned MFA device	Not assigned Manage
Signing certificates	None

Access keys

Use access keys to make secure REST or HTTP Query protocol requests to AWS service APIs. For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation. [Learn more](#)

[Create access key](#)

Access key ID	Created	Last used	Status
AKIAVFK6PBNZEKN5NL40	2020-10-22 08:00 UTC+0530	N/A	Active Make inactive

⇒ **IAM User Summary “nexususer” with Security Credentials Details.**

Task 3: Create a user with S3 full access

Screenshot 3: User summary

The screenshot shows the AWS IAM Management Console interface. The left sidebar is titled 'Identity and Access Management (IAM)' and includes sections for Dashboard, Access management (Groups, Users, Roles, Policies, Identity providers, Account settings), and Access reports (Archive rules, Analyzers, Settings, Credential report, Organization activity). The main content area is titled 'Summary' for the user 'ganeshsS3fullaccess'. It displays the User ARN (arn:aws:iam::355069791090:user/ganeshsS3fullaccess), Path (/), and Creation time (2020-10-22 08:39 UTC+0530). Below this, the 'Permissions' tab is selected, showing a table with one row: 'Attached directly' with the policy 'AmazonS3FullAccess' (AWS managed policy). Other tabs include Groups, Tags, Security credentials, and Access Advisor. A yellow box highlights the user name 'Ganesh_Subramanian' in the top right corner. A purple box highlights the 'Permissions' tab and the attached policy table.

⇒ IAM User Summary “ganeshsS3fullaccess” with Permission Details.

Task4: Create a group with ec2 full access

Screenshot 4: group summary

The screenshot shows the AWS IAM Management Console interface. The left sidebar navigation includes 'Identity and Access Management (IAM)' with 'Groups' selected, 'Users', 'Roles', 'Policies', 'Identity providers', and 'Account settings'. Below that is 'Access reports' with 'Access analyzer', 'Archive rules', 'Analyzers', and 'Settings'. At the bottom of the sidebar are 'Feedback', 'English (US)', 'Privacy Policy', and 'Terms of Use'. The main content area is titled 'IAM > Groups > Testing'. It displays the 'Summary' section with the Group ARN as 'arn:aws:iam::355069791090:group/Testing', 2 users in the group, a path of '/', and a creation time of '2020-10-22 09:05 UTC+0530'. Below this is a tab bar with 'Users' (selected), 'Permissions' (highlighted with a purple box), and 'Access Advisor'. The 'Permissions' tab leads to the 'Managed Policies' section, which lists the attached policy 'AmazonEC2FullAccess'. A blue 'Attach Policy' button is visible above the table. The table has columns for 'Policy Name' and 'Actions', showing 'AmazonEC2FullAccess' with options 'Show Policy', 'Detach Policy', and 'Simulate Policy'. A purple arrow points from the 'Permissions' tab in the sidebar to the 'Managed Policies' section in the main content area.

⇒ IAM Group “Testing” with Permission Details.

Task 5: Add user to a group and check if user policy and the group policy is reflecting on the user

Screenshot 5: user summary with permissions

The screenshot shows the IAM Management Console interface. The left sidebar is titled 'Identity and Access Management (IAM)' and includes sections for Dashboard, Access management (Groups, Users, Roles, Policies), Access reports (Access analyzer, Archive rules, Analyzers, Settings), and Credential report/Organization activity. The 'Groups' section is currently selected. The main content area shows the 'Testing' group details. The 'Summary' tab is selected, showing the Group ARN (arn:aws:iam::355069791090:group/Testing), Users (in this group) count (2), Path (/), and Creation Time (2020-10-22 09:05 UTC+0530). Below this, the 'Users' tab is selected, showing a table with two entries: 'nexususer' and 'ganeshsS3fullaccess'. Each entry has a 'Remove User from Group' action button. Other tabs include 'Permissions' and 'Access Advisor'.

⇒ IAM Group “Testing” with 2 Users Attached Details (“**nexususer**”, “**ganeshsS3fullaccess**”).

The screenshot shows the same IAM Management Console interface as the previous one, but the 'Permissions' tab is now selected under the 'Testing' group summary. The 'Managed Policies' section shows a single policy named 'AmazonEC2FullAccess' attached to the group. There are buttons for 'Attach Policy' (disabled), 'Show Policy', 'Detach Policy', and 'Simulate Policy'. The 'Inline Policies' section is also visible below.

⇒ IAM Group “Testing” with Permission Details.

Screenshot 6: login as this user show that this policy is in effect

IAM Management Console - https://console.aws.amazon.com/iam/home?region=us-east-2#/groups/Testing

Identity and Access Management (IAM)

- Dashboard
- Access management
 - Groups** (highlighted)
 - Users
 - Roles
 - Policies
 - Identity providers
 - Account settings
- Access reports
 - Access analyzer
 - Archive rules
 - Analyzers
 - Settings
- Credential report
- Organization activity

IAM > Groups > Testing

Summary

Group ARN: arn:aws:iam::355069791090:group/Testing
Users (in this group): 2
Path: /
Creation Time: 2020-10-22 09:05 UTC+0530

Users (selected) **Permissions** **Access Advisor**

This view shows all users in this group: 2 Users

User **Actions**

nexususer	Remove User from Group
ganeshsS3fullaccess	Remove User from Group

Feedback English (US) ▾ © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

⇒ IAM Group “Testing (EC2 Full access)” with 2 Users Attached Details.

- “nexususer” (User 1: - No Access at User Level).
- “ganeshsS3fullaccess” (User 2: - S3 Full Access at User Level).

IAM Management Console - https://console.aws.amazon.com/iam/home?region=us-east-2#/groups/Testing

Identity and Access Management (IAM)

- Dashboard
- Access management
 - Groups** (highlighted)
 - Users
 - Roles
 - Policies
 - Identity providers
 - Account settings
- Access reports
 - Access analyzer
 - Archive rules
 - Analyzers
 - Settings
- Credential report
- Organization activity

IAM > Groups > Testing

Summary

Group ARN: arn:aws:iam::355069791090:group/Testing
Users (in this group): 2
Path: /
Creation Time: 2020-10-22 09:05 UTC+0530

Users **Permissions** (highlighted) **Access Advisor**

Managed Policies

The following managed policies are attached to this group. You can attach up to 10 managed policies.

Attach Policy

Policy Name	Actions
AmazonEC2FullAccess	Show Policy Detach Policy Simulate Policy

Inline Policies

Feedback English (US) ▾ © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

⇒ IAM Group “Testing” with EC2FullAccess Permission Details.

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.

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
General purpose	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

- ⇒ Logged in as User “**ganeshsS3fullaccess**” which has S3 full access at **User Level** & it’s been attached to the Group “**Testing**” which has EC2 full access at the **Group Level**; The User “**ganeshsS3fullaccess**” is able to **Create a New Instance successfully**.

Amazon S3

Buckets

Access analyzer for S3

Block public access (account settings)

Feature spotlight

Create bucket

S3 buckets

Discover the console

Search for buckets

All access types

1 Buckets 1 Regions

Bucket name	Access	Region	Date created
ganeshbuckettest	Bucket and objects not public	US East (Ohio)	Oct 22, 2020 8:50:47 AM GMT+0530

- ⇒ User “**ganeshsS3fullaccess**” which has S3 full access at **User Level** & it’s been attached to the Group “**Testing**” which has EC2 full access at the **Group Level**; The User “**ganeshsS3fullaccess**” is able to **Create a S3 Bucket successfully**.

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.

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
General purpose	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

- ⇒ Logged in as User “**nexususer**” which has been attached to the Group “**Testing**” which has EC2 full access at the Group Level; The User “**nexususer**” is able to **Create a New Instance successfully**.

Amazon S3

Buckets Batch operations Access analyzer for S3 Block public access (account settings) Feature spotlight

S3 buckets

We've temporarily re-enabled the previous version of the S3 console while we continue to improve the new S3 console experience. [Switch to the new console](#)

+ Create bucket Edit public access settings Empty Delete Buckets 0 Regions

Error
Access Denied

Bucket name Access Region Date created

- ⇒ User “**nexususer**” which has **DOES NOT** have S3 full access at User Level & it’s been attached to the Group “**Testing**” which has EC2 full access at the Group Level; The User “**nexususer**” is **Unable** to Create a S3 Bucket successfully.

Task 6: Copy policies from the existing user

Screenshot 7: attach user summary of the user from which you create a new user

The screenshot shows the IAM Management Console interface. The left sidebar is collapsed. The main area displays the 'Summary' for a user named 'existingpolicyUser'. The 'Permissions' tab is selected, showing two policies applied:

- Attached directly: AmazonS3FullAccess (AWS managed policy)
- Attached from group: AmazonEC2FullAccess (AWS managed policy from group Testing)

- ⇒ New User “**existingpolicyUser**” which is given the permission Copied from existing user “**ganeshsS3fullaccess**” where the User “**ganeshsS3fullaccess**” has S3 full access at User Level & EC2 full access at Group Level.

The screenshot shows the IAM Management Console interface. The left sidebar is collapsed. The main area displays the 'Summary' for a user named 'existingpolicyUser'. The 'Groups' tab is selected, showing one group attached:

Group name	Attached permissions
Testing	AmazonEC2FullAccess

- ⇒ New User “**existingpolicyUser**” which is given the permission Copied from existing user “**ganeshsS3fullaccess**” where the User “**ganeshsS3fullaccess**” has S3 full access at User Level & EC2 full access at Group Level.

Screenshot 8: login as this user show that this policy is in effect

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 types ▾ Current generation ▾ Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

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

Cancel Previous Review and Launch Next: Configure Instance Details

Feedback English (US) ▾ © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

⇒ Logged in as New User “existingpolicyUser” and it has **EC2 full access**.

Amazon S3

- Buckets
- Batch operations
- Access analyzer for S3
- Block public access (account settings)
- Feature spotlight (2)

Access S3-backed file shares on premises and reduce local storage costs using AWS Storage Gateway. [Learn more](#) » Documentation

We've temporarily re-enabled the previous version of the S3 console while we continue to improve the new S3 console experience. [Switch to the new console](#).

S3 buckets

+ Create bucket Edit public access settings Empty Delete

Bucket name	Access	Region	Date created
ganeshbuckettest	Bucket and objects not public	US East (Ohio)	Oct 22, 2020 8:50:47 AM GMT+0530

Discover the console

All access types

Feedback English (US) ▾ © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

⇒ Logged in as New User “existingpolicyUser” and it has **S3 full access**.

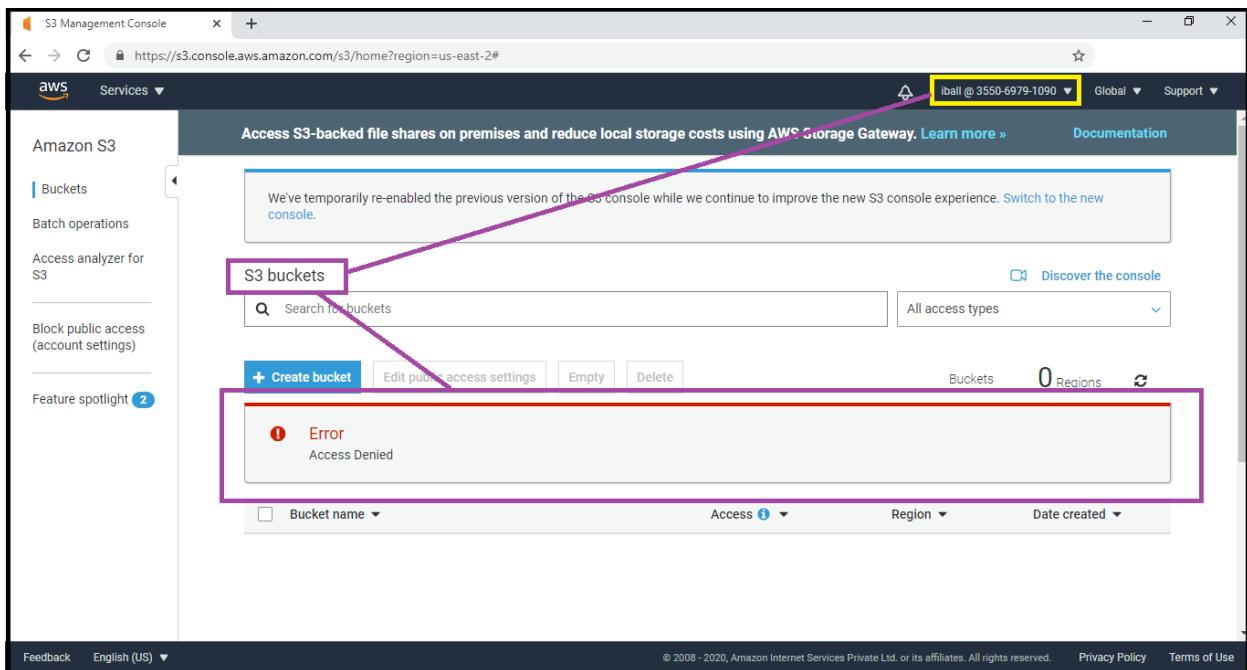
Task 7: Add user to a group in the process of creating a user

The screenshot shows the AWS IAM Management Console. On the left, the navigation menu includes 'Identity and Access Management (IAM)', 'Access management' (with 'Groups' selected), 'Users' (selected), 'Roles', 'Policies', 'Identity providers', 'Account settings', 'Access reports', 'Access analyzer', 'Archive rules', 'Analyzers', 'Settings', 'Credential report', and 'Organization activity'. The main area is titled 'Summary' for the user 'iball'. It displays the User ARN (arn:aws:iam::355069791090:user/iball), Path (/), and Creation time (2020-10-22 11:16 UTC+0530). Below this, there are tabs for 'Permissions', 'Groups (1)' (which is highlighted with a purple box), 'Tags', 'Security credentials', and 'Access Advisor'. A blue button labeled 'Add user to groups' is visible. Under the 'Groups' tab, a table shows one group assignment: 'Testing' with 'Attached permissions' 'AmazonEC2FullAccess'. A purple arrow points from the 'Groups (1)' tab to this table.

⇒ New User “iball” which is Added to User Group of “Testing” which has EC2 full access.

The screenshot shows the AWS EC2 Launch Instance Wizard at Step 2: Choose an Instance Type. The URL is https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard. The top navigation bar shows 'Services' and the user 'iball @ 3550-6979-1090'. The wizard steps are 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The 'Choose Instance Type' section is highlighted with a purple box. It shows a table of instance types under the heading 'Step 2: Choose an Instance Type'. The table columns are Family, Type, vCPUs, Memory (GiB), Instance Storage (GB), EBS-Optimized Available, Network Performance, and IPv6 Support. The 'Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)' is also highlighted with a purple box. The 't2.micro' row is selected, indicated by a blue checkbox. Other rows include t2.nano, t2.small, t2.medium, t2.large, and t2.xlarge. At the bottom are 'Cancel', 'Previous', 'Review and Launch' (highlighted with a purple box), and 'Next: Configure Instance Details'.

⇒ Logged in as New User “iball” and it has EC2 full access.



- ⇒ Logged in as New User "iball" and it **Does Not have S3 full access**, since "**Testing**" Group has **EC2 Full Access only**.

Task8: setting a password policy

Screenshot 9: password policy screen

The screenshot shows the IAM Management Console with the 'Account settings' section selected. In the center, a message box says 'Password policy updated.' Below it, a box highlights the password policy details:

- Minimum password length is 10 characters
- Require at least one uppercase letter from Latin alphabet (A-Z)
- Require at least one lowercase letter from Latin alphabet (a-z)
- Require at least one number
- Require at least one non-alphanumeric character (! @ # \$ % ^ & * () _ + - [] { } { } !)
- Password expires in 90 day(s)
- Allow users to change their own password
- Remember last 7 password(s) and prevent reuse

At the bottom, there are 'Change password policy' and 'Delete password policy' buttons.

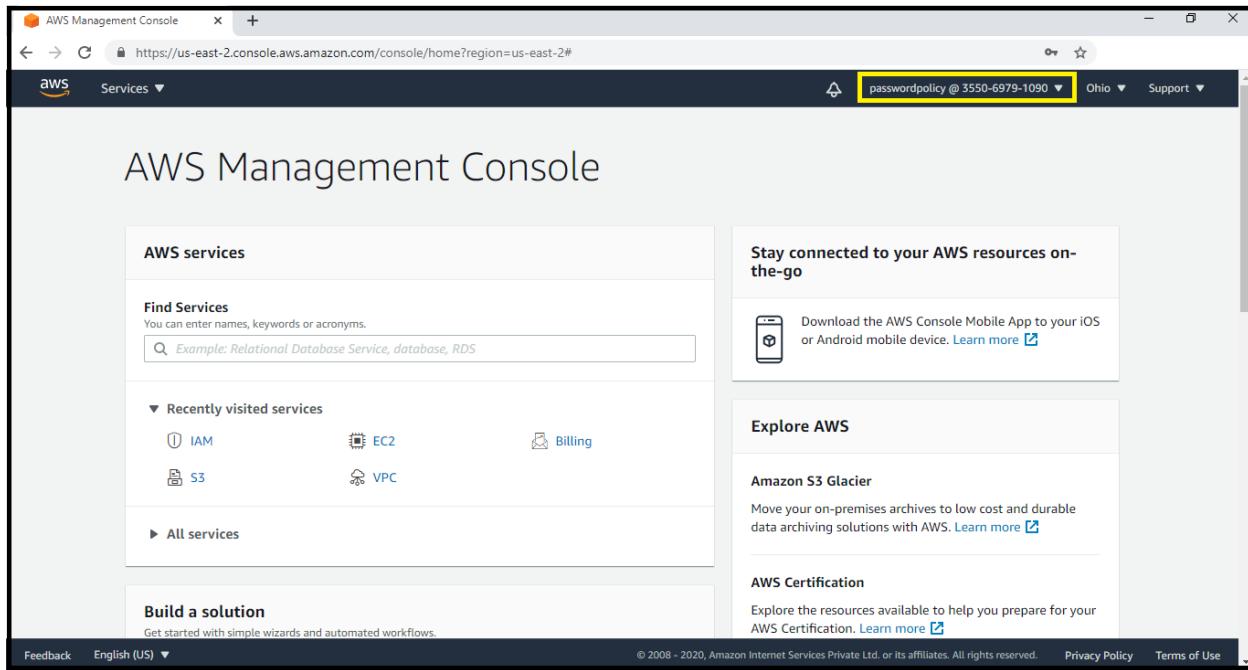
⇒ New Password Policy has been created.

Screenshot 10: login as the user and show password incompatibility error

The screenshot shows the 'Create New User' wizard at the 'Select AWS access type' step. Under 'Access type*', 'Programmatic access' and 'AWS Management Console access' are checked. Under 'Console password*', 'Custom password' is selected and a password is entered. An error message box highlights the password entry:

The password does not conform to the account password policy:
• it must contain at least 10 characters
• it must contain a lower case character, an upper case character, a special character and a digit

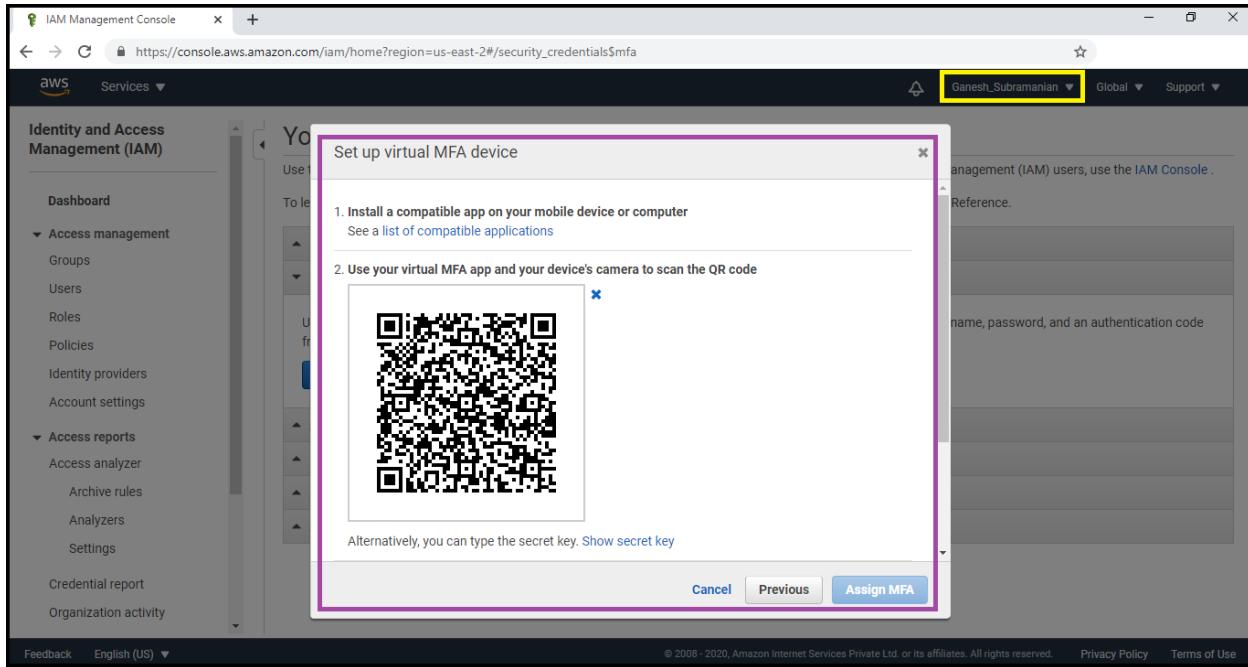
⇒ Password Incompatibility Error.



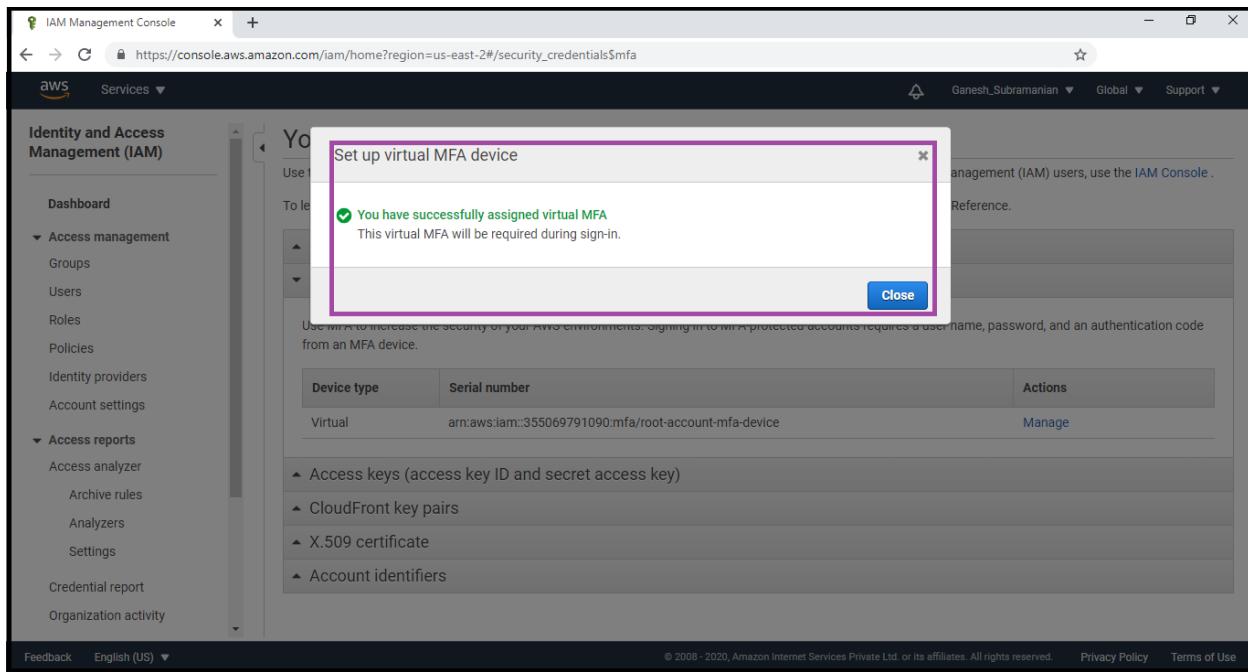
⇒ **Successful login as a New user “passwordpolicy”.**

Task 9: Enabling MFA and using an MFA device

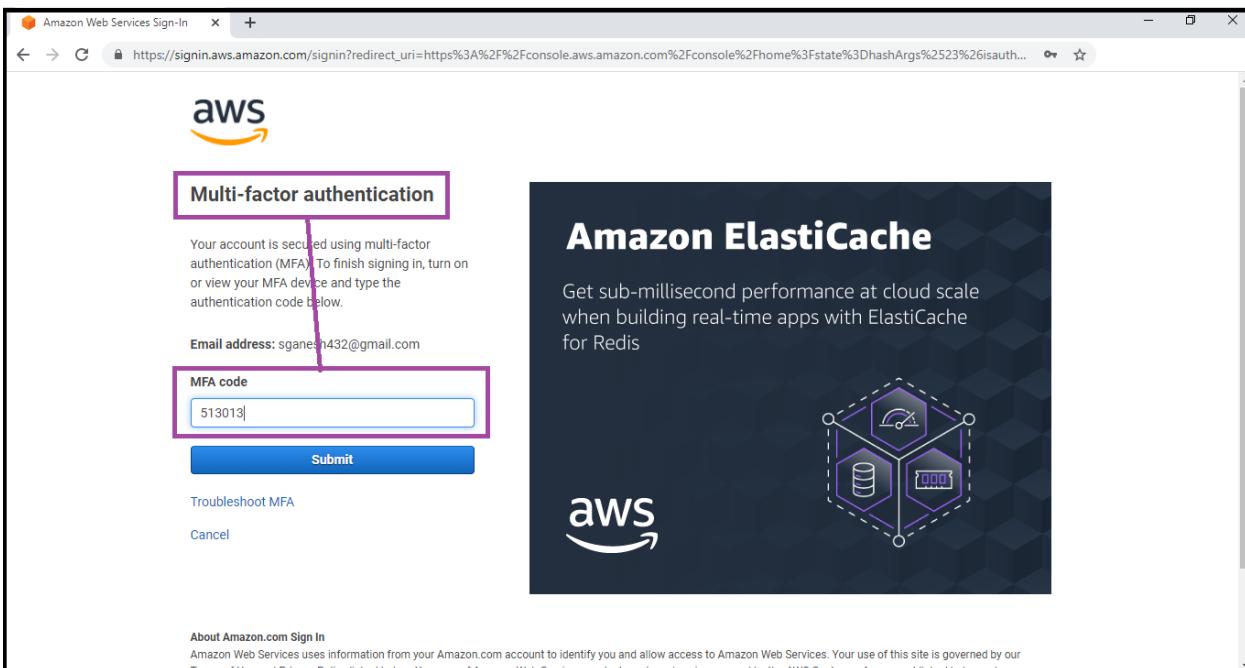
Screenshot 11: Enable MFA



⇒ **MFA is Enabled** and QR Code is requested to be Scanned for further process.



Screenshot 12: login screen for MFA



⇒ **MFA Code** is being Prompted during the Login session after Enabling the MFA option.