

Task – 16

NACL and Security group

Creation of VPC

The screenshot shows the AWS Management Console VPC dashboard. The left sidebar contains navigation links for VPC, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections, and Route servers. The main content area displays 'Your VPCs (1/2)' with a table listing VPCs. The 'my_vpc' is selected, and its details are shown below.

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
-	vpc-00aa32b792ae81dee	Available	Off	172.31.0.0/16	-
my_vpc	vpc-088da4e83bda295b1	Available	Off	10.0.0.0/16	-

vpc-088da4e83bda295b1 / my_vpc

Details

- VPC ID: vpc-088da4e83bda295b1
- State: Available
- Block Public Access: Off
- DNS hostnames: Disabled
- DNS resolution: Enabled
- Tenancy: default
- DHCP option set: dopt-0cdfba8a3425f7e40
- Main route table: rtb-03a98e700cfc2678
- Main network ACL: acl-0a2b5472e4fdd5999
- Default VPC: No
- IPv4 CIDR: 10.0.0.0/16
- IPv6 pool: -

Creation of Subnet

The screenshot shows the AWS Management Console Subnets dashboard. The left sidebar contains navigation links for VPC, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections, and Route servers. The main content area displays 'Subnets (1/4)' with a table listing subnets. The 'my_pub_subnet' is selected, and its details are shown below.

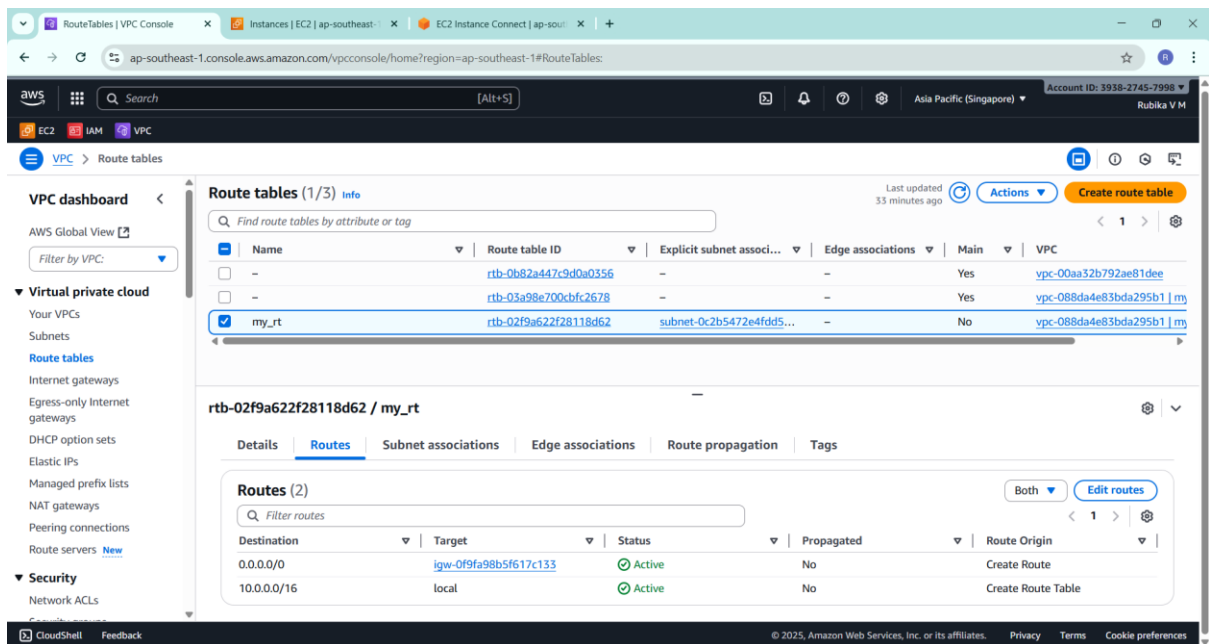
Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-00b1afc1e5e29345	Available	vpc-00aa32b792ae81dee	Off	172.31.32.0
-	subnet-09cd94db8c5f77451	Available	vpc-00aa32b792ae81dee	Off	172.31.0.0/16
-	subnet-03776fe5f5968f808	Available	vpc-00aa32b792ae81dee	Off	172.31.16.0
my_pub_subnet	subnet-0c2b5472e4fdd5999	Available	vpc-088da4e83bda295b1 my_vpc	Off	10.0.1.0/24

subnet-0c2b5472e4fdd5999 / my_pub_subnet

Details

- Subnet ID: subnet-0c2b5472e4fdd5999
- Subnet ARN: arn:aws:ec2:ap-southeast-1:393827457998:subnet/subnet-0c2b5472e4fdd5999
- State: Available
- Block Public Access: Off
- IPv4 CIDR: 10.0.1.0/24
- IPv6 CIDR: -
- Availability Zone: ap-southeast-1a
- Available IPv4 addresses: 250
- VPC: vpc-088da4e83bda295b1 | my_vpc
- Route table: rtb-02f9a622f28118d62 | my_rt

Creation of Route table



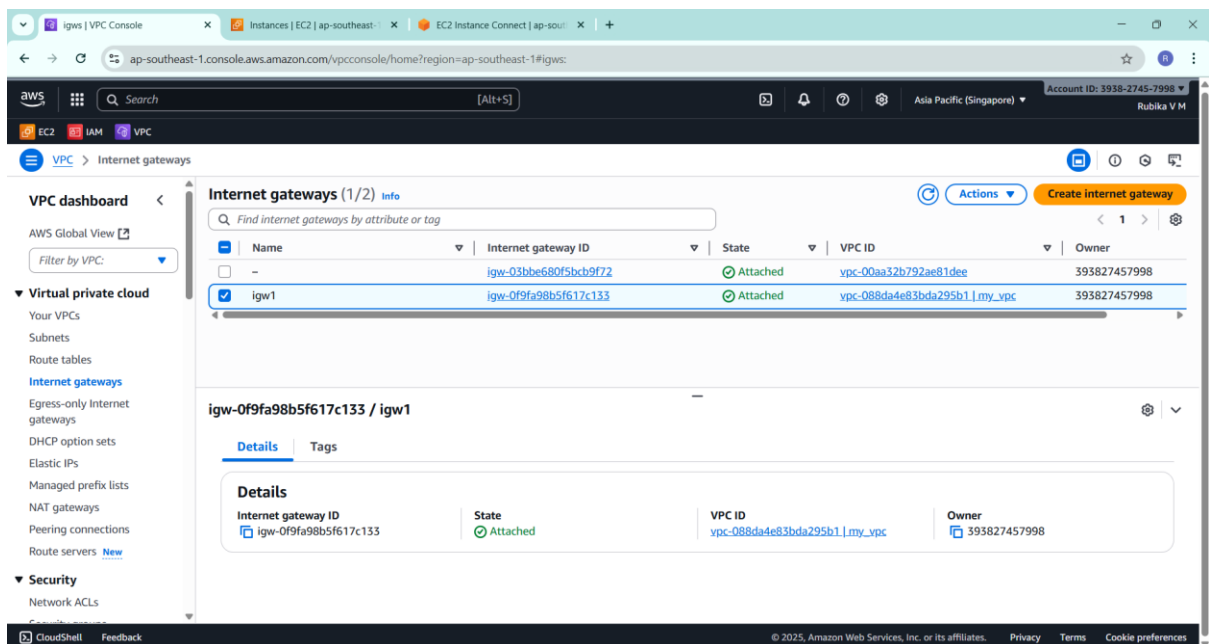
The screenshot shows the AWS VPC console interface. The left sidebar contains the 'VPC dashboard' with links to 'Virtual private cloud' and 'Security'. The main content area is titled 'Route tables (1/3)' and shows a list of route tables. The 'my_rt' route table is selected, and its details are shown below. The 'Routes' tab is active, displaying a table of routes.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
-	rtb-0b82a447c9d0a0356	-	-	Yes	vpc-00aa32b792ae81dee
-	rtb-03a98e700cbfc2678	-	-	Yes	vpc-088da4e83bda295b1 my_vpc
my_rt	rtb-02f9a622f28118d62	subnet-0c2b5472e4fdd5...	-	No	vpc-088da4e83bda295b1 my_vpc

Below the table, the details for 'rtb-02f9a622f28118d62 / my_rt' are shown. The 'Routes' tab is active, displaying a table of routes.

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	igw-0f9fa98b5f617c133	Active	No	Create Route
10.0.0.0/16	local	Active	No	Create Route Table

Creation of Internet gateway



The screenshot shows the AWS VPC console interface. The left sidebar contains the 'VPC dashboard' with links to 'Virtual private cloud' and 'Security'. The main content area is titled 'Internet gateways (1/2)' and shows a list of internet gateways. The 'igw1' internet gateway is selected, and its details are shown below. The 'Details' tab is active, displaying a table of details.

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-03bbe680f5bcb9f72	Attached	vpc-00aa32b792ae81dee	393827457998
igw1	igw-0f9fa98b5f617c133	Attached	vpc-088da4e83bda295b1 my_vpc	393827457998

Below the table, the details for 'igw-0f9fa98b5f617c133 / igw1' are shown. The 'Details' tab is active, displaying a table of details.

Internet gateway ID	State	VPC ID	Owner
igw-0f9fa98b5f617c133	Attached	vpc-088da4e83bda295b1 my_vpc	393827457998

Creation of security group

The screenshot shows the AWS Management Console with the 'Security Groups' page. A green notification banner at the top states: 'Security group (sg-0596412164b45d7fc | my_sec_grp) was created successfully'. Below this, the 'Security Groups (1/12)' table lists the newly created group. The 'Details' tab for 'sg-0596412164b45d7fc - my_sec_grp' is active, showing its configuration.

Name	Security group ID	Security group name	VPC ID	Description
my_sec_grp	sg-0596412164b45d7fc	my_sec_grp	vpc-088da4e83bda295b1	sg
launch-wizard-1	sg-044168cc1028c7e84	launch-wizard-1	vpc-00aa32b792ae81dee	launch-wizard-

sg-0596412164b45d7fc - my_sec_grp Details:

- Security group name: my_sec_grp
- Security group ID: sg-0596412164b45d7fc
- Description: sg
- VPC ID: vpc-088da4e83bda295b1
- Owner: 393827457998
- Inbound rules count: 2 Permission entries
- Outbound rules count: 1 Permission entry

Creation of NACL

Here it has the inbound rule as (all traffic: allow) that's why the web hosting happened

The screenshot shows the AWS Management Console with the 'Network ACLs' page. The 'acl-0de10ef26a5726fcf' Network ACL is selected. The 'Inbound rules' tab is active, showing a list of rules.

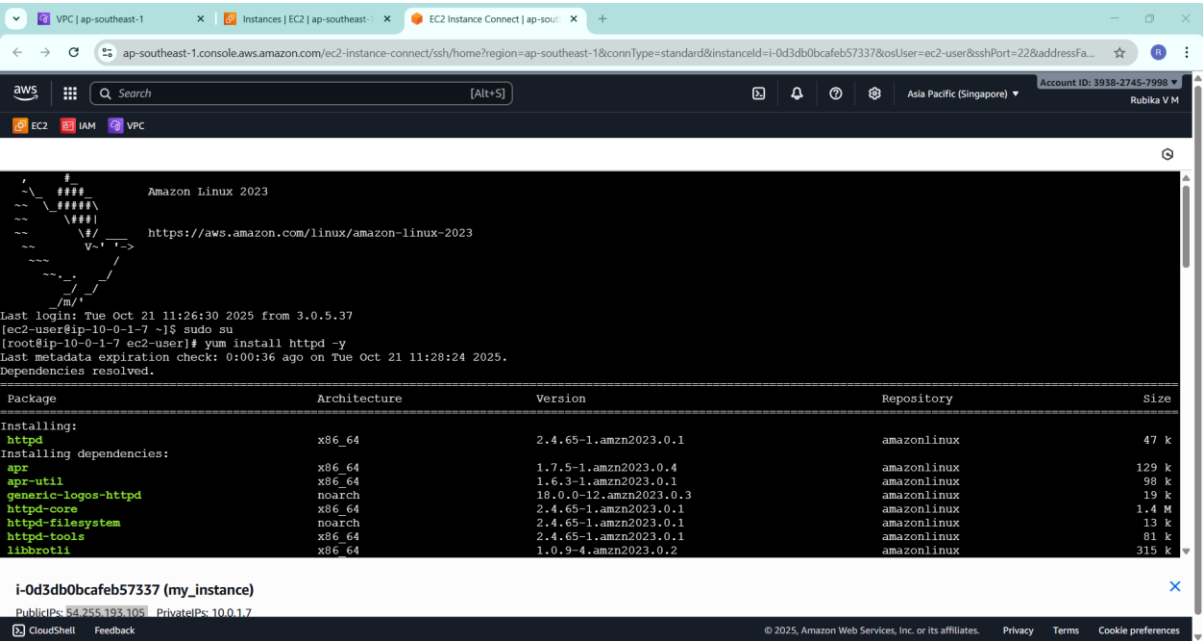
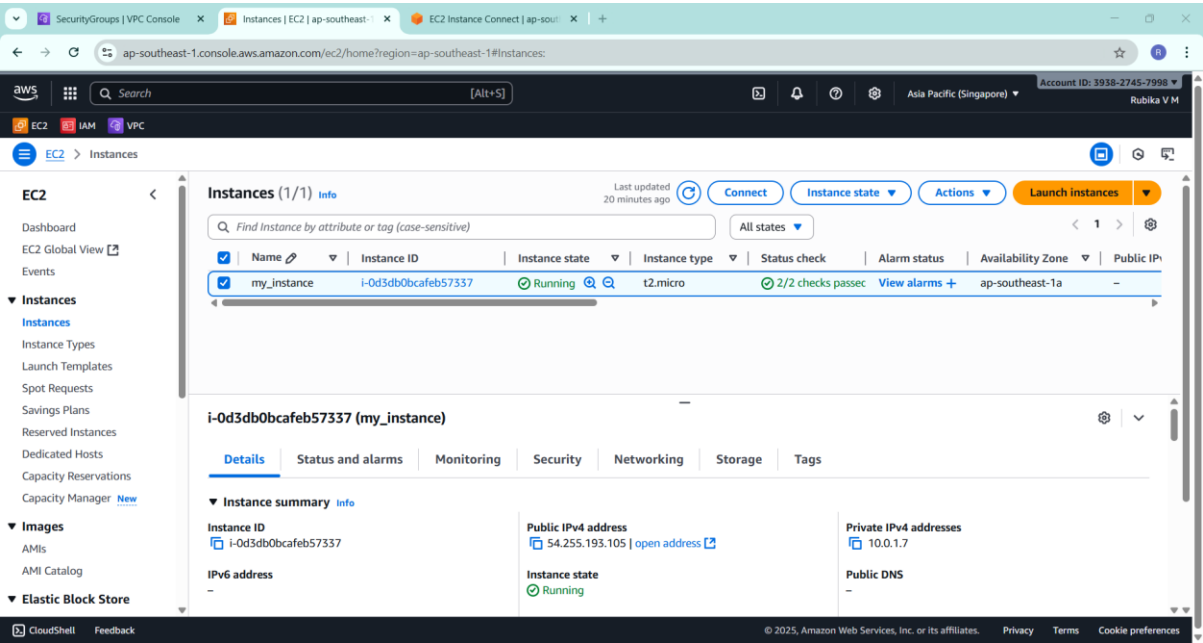
acl-0de10ef26a5726fcf Details:

- Network ACL ID: acl-0de10ef26a5726fcf
- Associated with: subnet-0c2b5472e4fdd5999 / my_pub_subnet
- Default: Yes
- VPC ID: vpc-088da4e83bda295b1 / my_vpc
- Owner: 393827457998

Inbound rules (2):

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	All traffic	All	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Creation of Instance



```
[root@ip-10-0-1-7 ec2-user]# systemctl status httpd
o httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)
[root@ip-10-0-1-7 ec2-user]# systemctl start httpd
[root@ip-10-0-1-7 ec2-user]# systemctl status httpd
• httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Tue 2025-10-21 11:29:37 UTC; 5s ago
     Docs: man:httpd.service(8)
  Main PID: 28694 (httpd)
    Status: "Started, listening on: port 80"
    Tasks: 177 (limit: 1106)
   Memory: 13.0M
        CPU: 61ms
    CGroup: /system.slice/httpd.service
            └─28694 /usr/sbin/httpd -DFOREGROUND
              └─28695 /usr/sbin/httpd -DFOREGROUND
                └─28696 /usr/sbin/httpd -DFOREGROUND
                  └─28697 /usr/sbin/httpd -DFOREGROUND
                    └─28698 /usr/sbin/httpd -DFOREGROUND

Oct 21 11:29:37 ip-10-0-1-7.ap-southeast-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Oct 21 11:29:37 ip-10-0-1-7.ap-southeast-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Oct 21 11:29:37 ip-10-0-1-7.ap-southeast-1.compute.internal httpd[28694]: Server configured, listening on: port 80
[root@ip-10-0-1-7 ec2-user]#
```

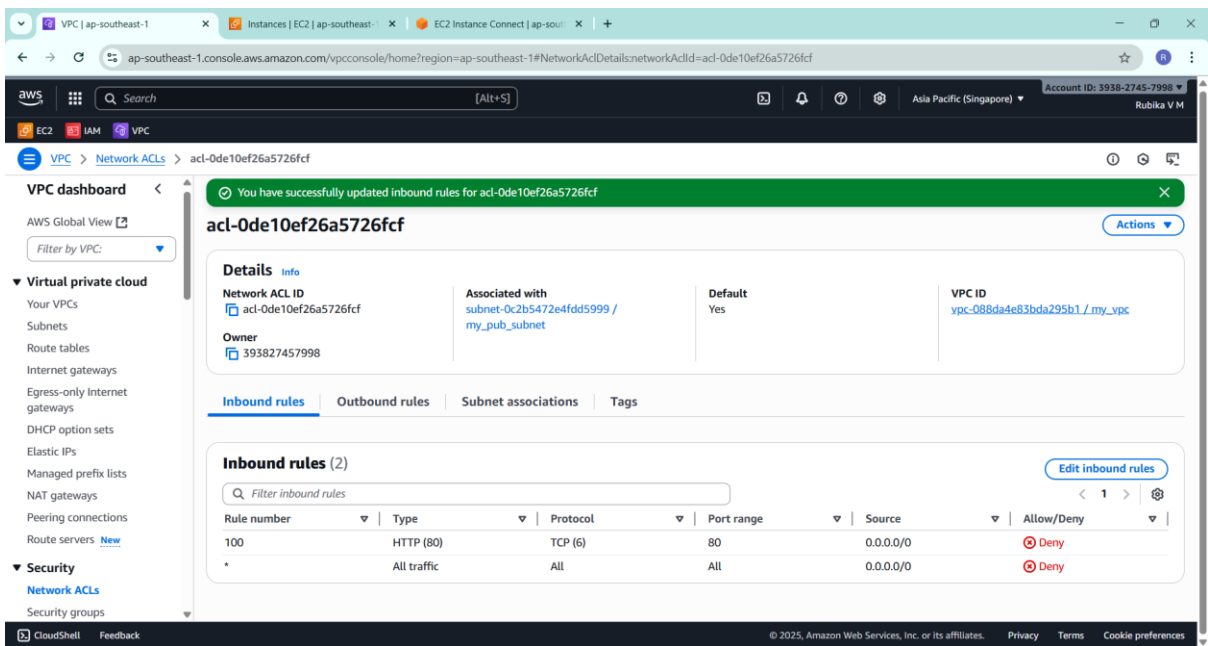
i-0d3db0bcafeb57337 (my_instance)
PublicIPs: 54.255.193.105 PrivateIPs: 10.0.1.7

Here we get the targeted output of the webhosting when the inbound rule's all traffic is allowed.

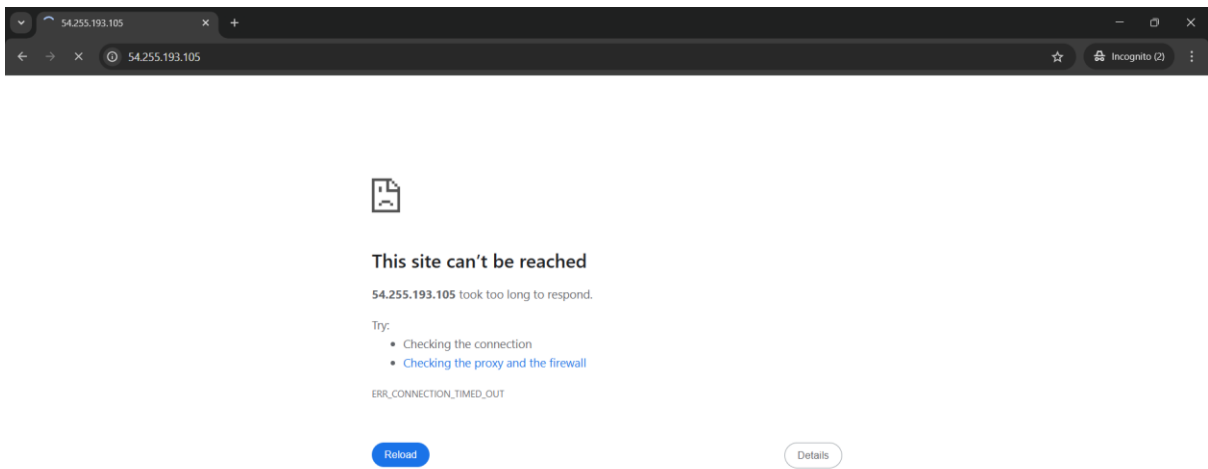


It works!

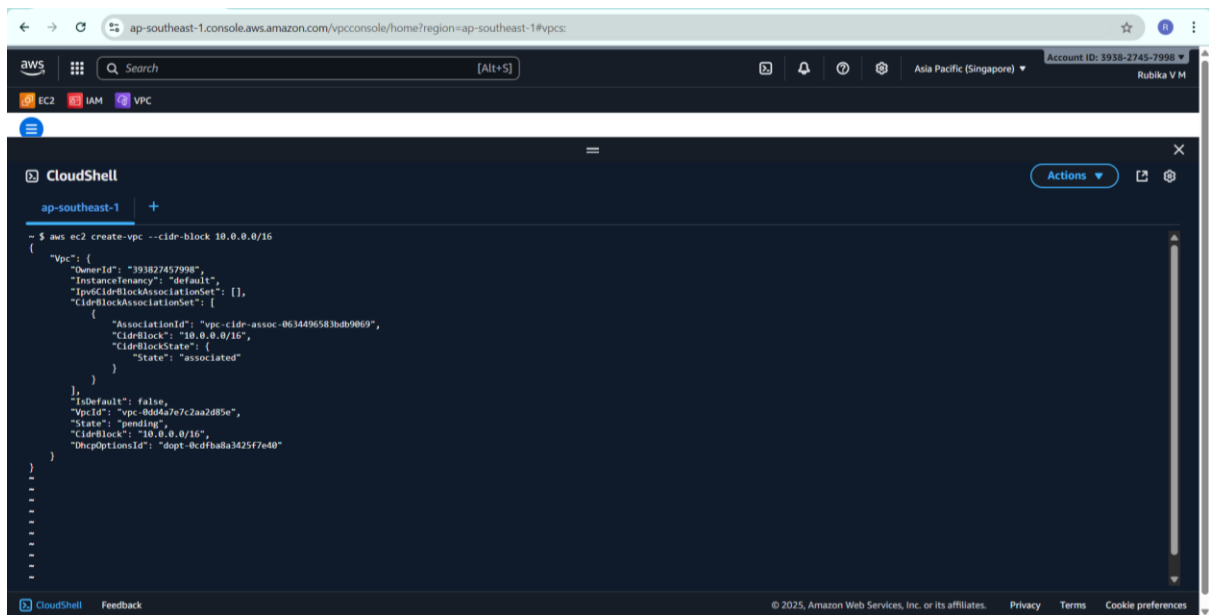
When the inbound rule is set to (http: deny) like the below screenshot



We don't get the desired output of web hosting.

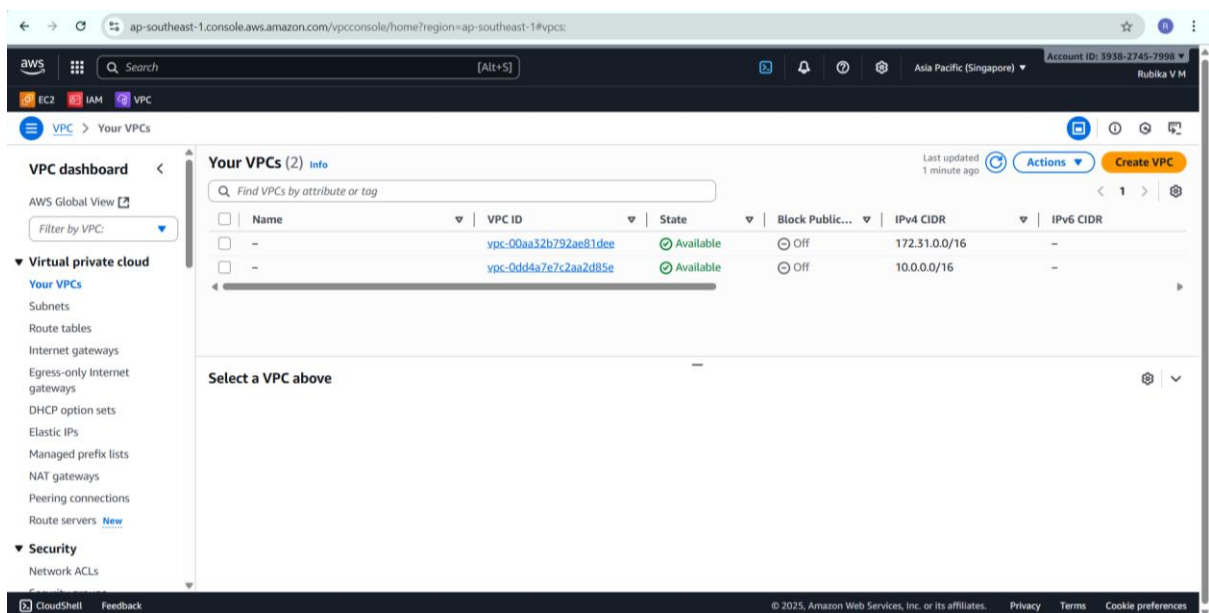


Create VPC through AWS CLI



The screenshot shows the AWS CloudShell interface in the Asia Pacific (Singapore) region. The terminal displays the command `aws ec2 create-vpc --cidr-block 10.0.0.0/16` and its output, which is a JSON object containing VPC details. The VPC ID is `vpc-0dd4a7e7c2aa2d85e` and its state is `pending`.

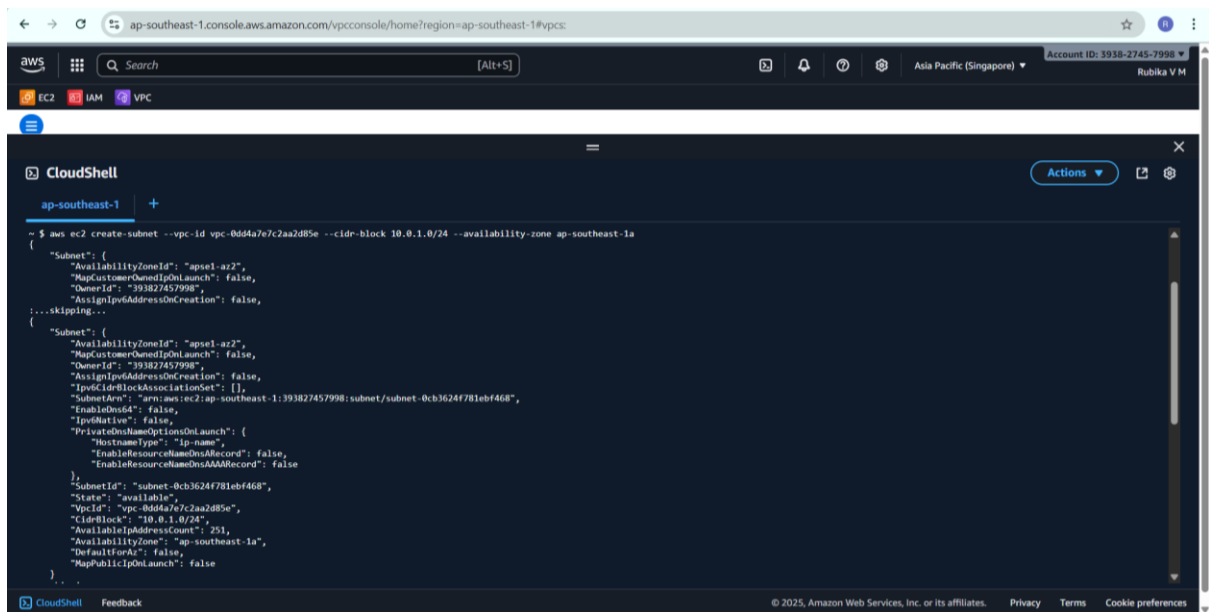
```
$ aws ec2 create-vpc --cidr-block 10.0.0.0/16
{
  "Vpc": {
    "OwnerId": "393827457998",
    "InstanceTenancy": "default",
    "Ipv6CidrBlockAssociationSet": [],
    "CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-0634496583bdb9069",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
          "State": "associated"
        }
      }
    ],
    "IsDefault": false,
    "VpcId": "vpc-0dd4a7e7c2aa2d85e",
    "State": "pending",
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-0cdfba8a3425f7e40"
  }
}
```



The screenshot shows the AWS VPC console. The 'Your VPCs (2)' section displays a table with two VPCs. The first VPC has ID `vpc-00aa32b792ae81dee` and is in an 'Available' state. The second VPC has ID `vpc-0dd4a7e7c2aa2d85e` and is also in an 'Available' state. The table includes columns for Name, VPC ID, State, Block Public..., IPv4 CIDR, and IPv6 CIDR.

	Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
<input type="checkbox"/>	-	vpc-00aa32b792ae81dee	Available	Off	172.31.0.0/16	-
<input type="checkbox"/>	-	vpc-0dd4a7e7c2aa2d85e	Available	Off	10.0.0.0/16	-

Creation of subnet through CLI

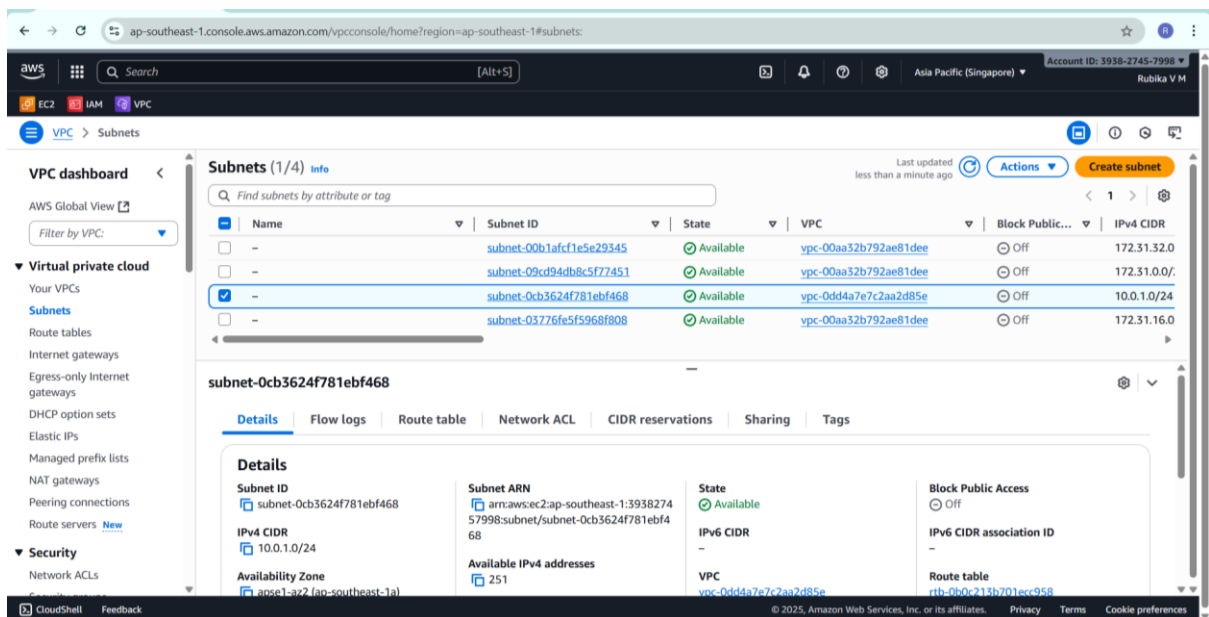


```
ap-southeast-1.console.aws.amazon.com/vpconsole/home?region=ap-southeast-1#vpcs:

AWS [Search] [Alt+S] Asia Pacific (Singapore) Account ID: 3938-2745-7998 Rubika V M

CloudShell ap-southeast-1 Actions

$ aws ec2 create-subnet --vpc-id vpc-0dd4a7e7c2aa2d85e --cidr-block 10.0.1.0/24 --availability-zone ap-southeast-1a
{
  "Subnet": {
    "AvailabilityZoneId": "apse1-ar2",
    "MapCustomerOwnedIpOnLaunch": false,
    "OwnerId": "393827457998",
    "AssignIpv6AddressOnCreation": false,
    "AssignIpv6AddressOnCreation": false,
    "SkipWaiting": true
  },
  "Subnet": {
    "AvailabilityZoneId": "apse1-ar2",
    "MapCustomerOwnedIpOnLaunch": false,
    "OwnerId": "393827457998",
    "AssignIpv6AddressOnCreation": false,
    "AssignIpv6AddressOnCreation": false,
    "SkipWaiting": true,
    "SubnetArn": "arn:aws:ec2:ap-southeast-1:393827457998:subnet/subnet-0cb3624f781ebf468",
    "EnableDns4": false,
    "Ipv6Native": false,
    "PrivateDnsNameOptionsOnLaunch": {
      "HostNameType": "ip-name",
      "EnableResourceNameDnsARecord": false,
      "EnableResourceNameDnsAAAARecord": false
    },
    "SubnetId": "subnet-0cb3624f781ebf468",
    "State": "available",
    "VpcId": "vpc-0dd4a7e7c2aa2d85e",
    "CidrBlock": "10.0.1.0/24",
    "AvailableIpAddressCount": 251,
    "AvailabilityZone": "ap-southeast-1a",
    "DefaultRouteTable": false,
    "MapPublicIpOnLaunch": false
  }
}
```



ap-southeast-1.console.aws.amazon.com/vpconsole/home?region=ap-southeast-1#subnets:

VPC Subnets

Subnets (1/4) Info

Find subnets by attribute or tag

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-00b1afc1e5e29345	Available	vpc-00aa32b792ae81dee	Off	172.31.32.0
-	subnet-09cd94db8c5f77451	Available	vpc-00aa32b792ae81dee	Off	172.31.0.0/
-	subnet-0cb3624f781ebf468	Available	vpc-0dd4a7e7c2aa2d85e	Off	10.0.1.0/24
-	subnet-03776fe5f5968f808	Available	vpc-00aa32b792ae81dee	Off	172.31.16.0

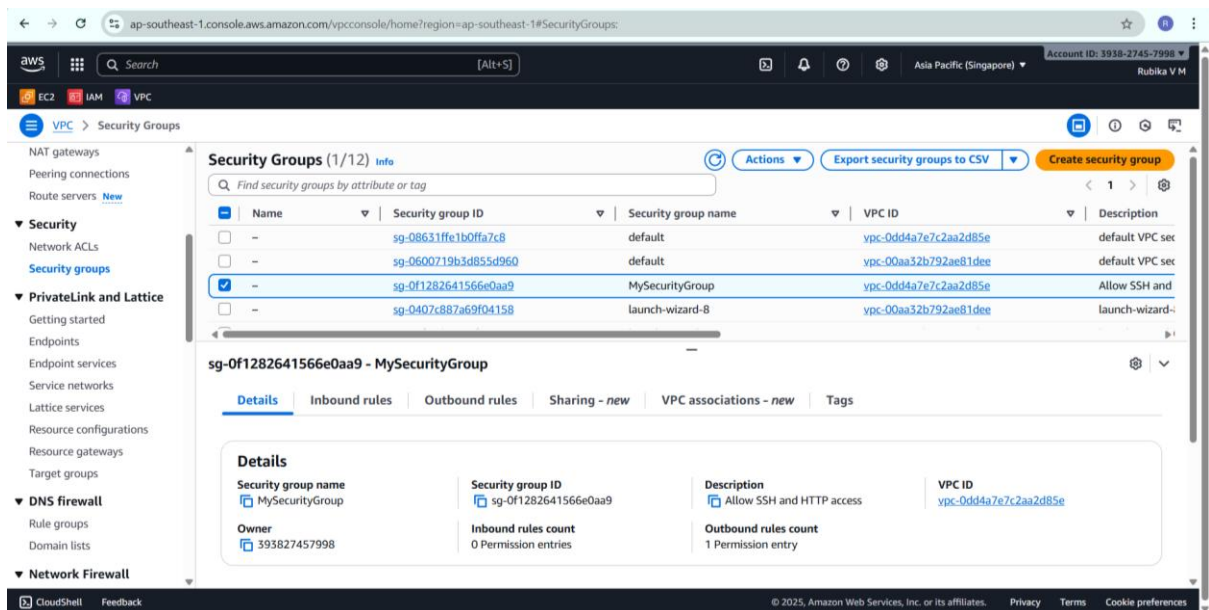
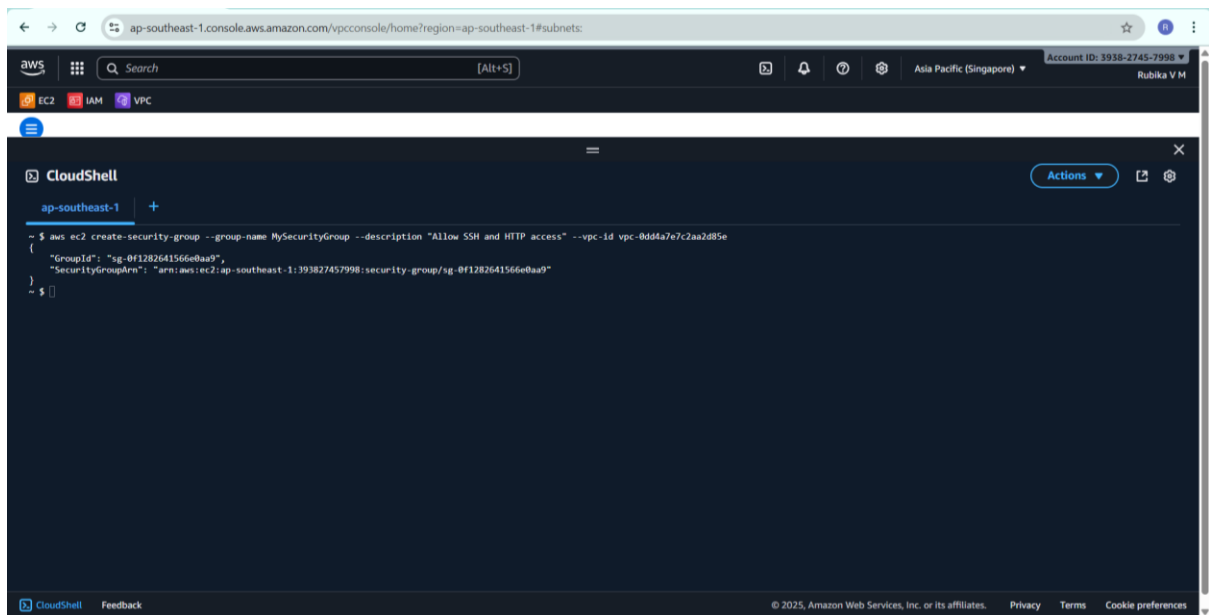
subnet-0cb3624f781ebf468

Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

Details

Subnet ID subnet-0cb3624f781ebf468	Subnet ARN arn:aws:ec2:ap-southeast-1:393827457998:subnet/subnet-0cb3624f781ebf468	State Available	Block Public Access Off
IPv4 CIDR 10.0.1.0/24	Available IPv4 addresses 251	IPv6 CIDR -	IPv6 CIDR association ID -
Availability Zone apse1-ar2 (ap-southeast-1a)		VPC vpc-0dd4a7e7c2aa2d85e	Route table rtb-0b0c213b701ecc958

Creation of security group through CLI



Changing port number:

The screenshot shows the AWS Management Console for the 'ap-southeast-1' region. The 'Instances' page displays a table with one instance: 'port_instance' (ID: i-09627ebf5879b38ab), which is in the 'Running' state. Below the table, the 'Details' tab for this instance is selected, showing the 'Instance summary' section. A tooltip indicates that the 'Public IPv4 address' has been copied. The summary shows the instance ID, the public IP address (52.221.191.87), and the private IP address (172.31.22.161). The instance state is 'Running'.

The screenshot shows the AWS CloudShell terminal interface. The terminal output displays the Amazon Linux 2023 logo and the URL 'https://aws.amazon.com/linux/amazon-linux-2023'. The user runs the command 'sudo su' to become root. Then, the user runs 'nano /etc/ssh/sshd_config' to edit the SSH configuration file. The terminal shows the configuration file being edited, and the user runs 'systemctl restart sshd' to restart the SSH service. The terminal output shows the instance ID 'i-09627ebf5879b38ab (port_instance)' and the public IP address '52.221.191.87'.

```

❯ ec2-user@ip-172-31-22-161:~
vmrub@Rubiii MINGW64 ~
$ cd downloads
vmrub@Rubiii MINGW64 ~/downloads
$ ssh -i awsperm.pem -p 24 ec2-user@52.221.191.87
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html

#
#####
#####
#####
#|
#|
V~'~>
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Tue Oct 21 16:35:32 2025 from 157.51.19.131
[ec2-user@ip-172-31-22-161 ~]$

```