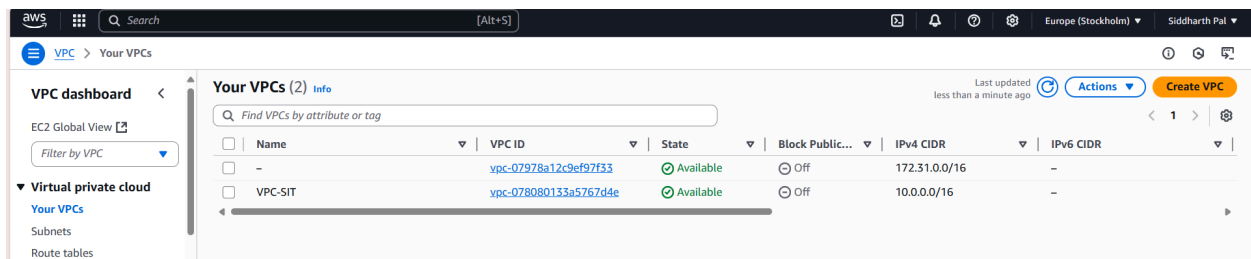


AWC Class Notes – VPC, Subnets, EC2 Setup (18 June 2025) Siddharth Pal || 22BCSI14

Step 1: Create a VPC

- Name: **VPC-SIT**
- CIDR Block: **10.0.0.0/16**



Step 2: Create Subnets

- Web Subnet
 - Name: **web-subnet**
 - CIDR: **10.0.1.0/24**
- DB Subnet
 - Name: **db-subnet**
 - CIDR: **10.0.2.0/24**

The screenshot shows the AWS Management Console interface for the 'Subnets' page. The left sidebar contains navigation links for VPC dashboard, EC2 Global View, and various VPC resources. The main content area displays a table of subnets. The 'db-subnet' is selected, and its details are shown below the table.

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-0039e9f11fe01c9ba	Available	vpc-07978a12c9ef97f33	Off	172.31.32.0/20
-	subnet-0920373a71fff3006	Available	vpc-07978a12c9ef97f33	Off	172.31.16.0/20
-	subnet-0bba0c36ace27ede5	Available	vpc-07978a12c9ef97f33	Off	172.31.0.0/20
web-subnet	subnet-0533299f16420f909	Available	vpc-078080133a5767d4e VPC...	Off	10.0.1.0/24
db-subnet	subnet-08922fe4927922c96	Available	vpc-078080133a5767d4e VPC...	Off	10.0.2.0/24

subnet-08922fe4927922c96 / db-subnet

Details

Subnet ID subnet-08922fe4927922c96	Subnet ARN arn:aws:ec2:eu-north-1:961308088645:subnet/subnet-08922fe4927922c96	State Available	Block Public Access Off
IPv4 CIDR 10.0.2.0/24	Available IPv4 addresses 250	IPv6 CIDR -	IPv6 CIDR association ID -
Availability Zone eu-north-1c	Availability Zone ID eu-north-1c	Network border group eu-north-1	VPC vpc-078080133a5767d4e VPC...

✓ Step 3: Create Route Tables

- **Web Route Table**
 - Name: **web-rt**
 - Associate with: **web-subnet**
- **DB Route Table**
 - Name: **db-rt**
 - Associate with: **db-subnet**

Route tables (1/4) Info

Find route tables by attribute or tag

	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	
<input type="checkbox"/>	-	rtb-0ff0e66aa272f143a	-	-	Yes	vpc-07978a12c9ef97f33	9
<input checked="" type="checkbox"/>	web-rt	rtb-0e121e64c47555af7	subnet-0533299f16420f...	-	No	vpc-078080133a5767d4e VPC...	9
<input type="checkbox"/>	-	rtb-0803f236eb81725d8	-	-	Yes	vpc-078080133a5767d4e VPC...	9
<input type="checkbox"/>	db-rt	rtb-0746de80cb6743177	subnet-08922fe4927922...	-	No	vpc-078080133a5767d4e VPC...	9

rtb-0e121e64c47555af7 / web-rt

Details Routes Subnet associations Edge associations Route propagation Tags

Details

Route table ID rtb-0e121e64c47555af7	Main <input type="checkbox"/> No	Explicit subnet associations subnet-0533299f16420f09 / web-subnet	Edge associations -
VPC vpc-078080133a5767d4e VPC-SIT	Owner ID 961308088645		

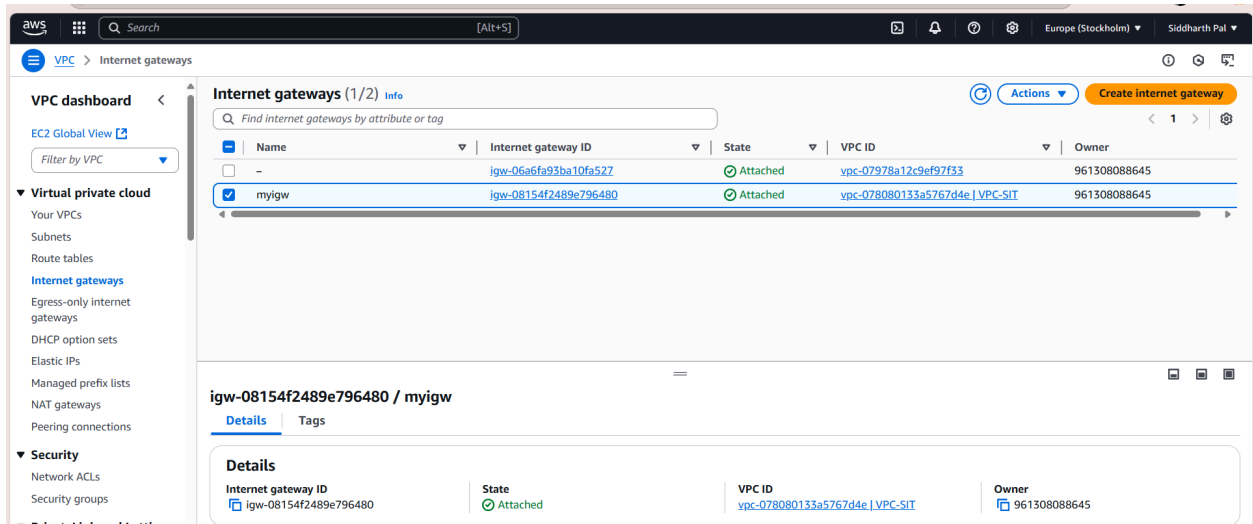
✓ Step 4: Create and Attach Internet Gateway

- Name: **myigw**
- Attach to **VPC-SIT**

✓ Step 5: Edit Routes in **web-rt**

- Add route:
 - Destination: **0.0.0.0/0**
 - Target: Internet Gateway (**myigw**)

🔒 This makes **web-subnet** a **Public Subnet**



The screenshot shows the AWS Management Console interface for the 'Internet gateways' section. On the left, there is a navigation menu with options like 'VPC dashboard', 'EC2 Global View', and 'Virtual private cloud'. The main content area shows a table of internet gateways. One gateway, named 'myigw', is selected and highlighted. Below the table, the details for 'myigw' are displayed, including its ID, state (Attached), VPC ID, and owner.

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-06a6fa93ba10fa527	Attached	vpc-07978a12c9ef97f33	961308088645
myigw	igw-08154f2489e796480	Attached	vpc-078080133a5767d4e VPC-SIT	961308088645

igw-08154f2489e796480 / myigw

Details

Internet gateway ID	State	VPC ID	Owner
igw-08154f2489e796480	Attached	vpc-078080133a5767d4e VPC-SIT	961308088645

Step 6: Deploy EC2 Instances

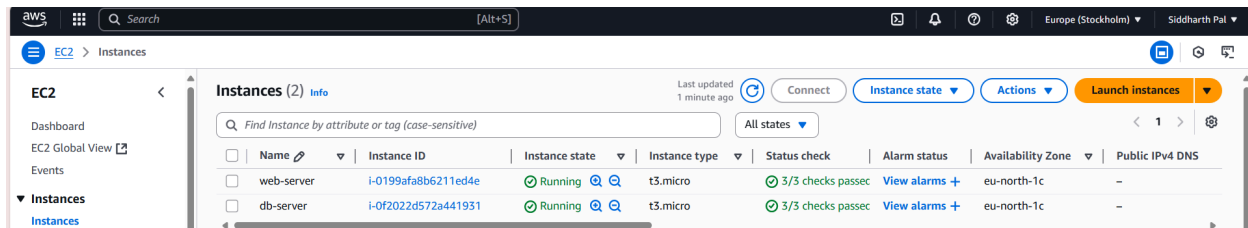
Web Server Instance (Public)

- **Key Pair:** aws-vpn
- **VPC:** VPC-SIT
- **Subnet:** web-subnet
- **Auto-assign Public IP:** ☒ Enabled
- **Security Group:** Allow All Traffic
- **Launch**

DB Server Instance (Private)

- **Key Pair:** aws-vpn
- **VPC:** VPC-SIT

- Subnet: **db-subnet**
- Auto-assign Public IP: **✗ Disabled**
- Security Group: Allow **All Traffic**
- Launch



The screenshot shows the AWS Management Console interface for EC2 Instances. The left sidebar contains navigation links for EC2, Dashboard, EC2 Global View, Events, and Instances. The main content area displays a table of instances. Two instances are listed: 'web-server' and 'db-server'. Both are in a 'Running' state. The 'web-server' instance has ID 'i-0199afa8b6211ed4e' and the 'db-server' instance has ID 'i-0f2022d572a441931'. Both are t3.micro instances in the eu-north-1c availability zone. The status check for both shows '3/3 checks passed'.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
web-server	i-0199afa8b6211ed4e	Running	t3.micro	3/3 checks passed	View alarms +	eu-north-1c	-
db-server	i-0f2022d572a441931	Running	t3.micro	3/3 checks passed	View alarms +	eu-north-1c	-

Connecting to Web Server (Jump Host)

1. Copy the **Public IP** of the **Web Server** (from **web-subnet**).

Open **MobaXterm**, and connect via SSH using:

```
pgsql
CopyEdit
ssh -i aws-vpn.pem ec2-user@<PUBLIC-IP>
```

- 2.

After login:

```
bash
CopyEdit
sudo su

cd
```

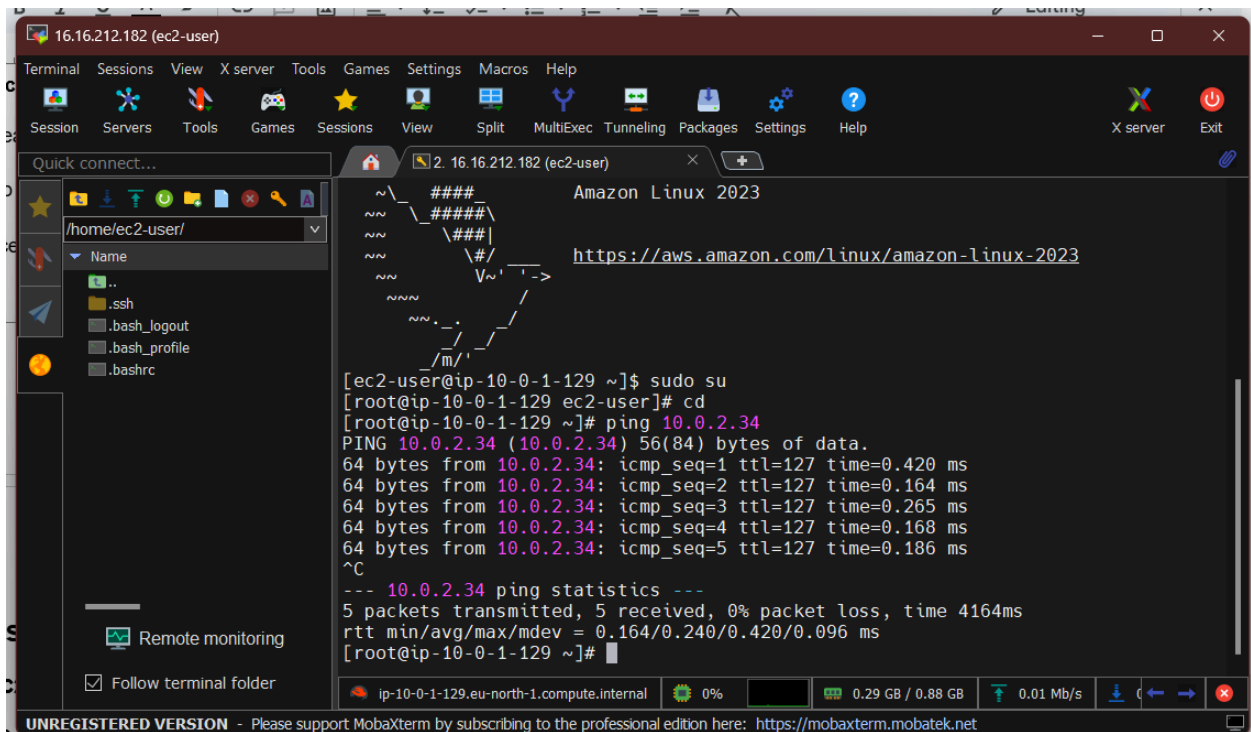
- 3.

To test private connection:

```
bash
```

CopyEdit

ping <PRIVATE-IP-OF-DB-SERVER>



The screenshot shows a MobaXterm terminal window titled "16.16.212.182 (ec2-user)". The terminal displays the following commands and output:

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-0-1-129 ~]$ sudo su
[root@ip-10-0-1-129 ec2-user]# cd
[root@ip-10-0-1-129 ~]# ping 10.0.2.34
PING 10.0.2.34 (10.0.2.34) 56(84) bytes of data.
64 bytes from 10.0.2.34: icmp_seq=1 ttl=127 time=0.420 ms
64 bytes from 10.0.2.34: icmp_seq=2 ttl=127 time=0.164 ms
64 bytes from 10.0.2.34: icmp_seq=3 ttl=127 time=0.265 ms
64 bytes from 10.0.2.34: icmp_seq=4 ttl=127 time=0.168 ms
64 bytes from 10.0.2.34: icmp_seq=5 ttl=127 time=0.186 ms
^C
--- 10.0.2.34 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4164ms
rtt min/avg/max/mdev = 0.164/0.240/0.420/0.096 ms
[root@ip-10-0-1-129 ~]#
```

The terminal window also shows a file explorer on the left with the path "/home/ec2-user/" and a status bar at the bottom indicating "UNREGISTERED VERSION" and a link to the professional edition.

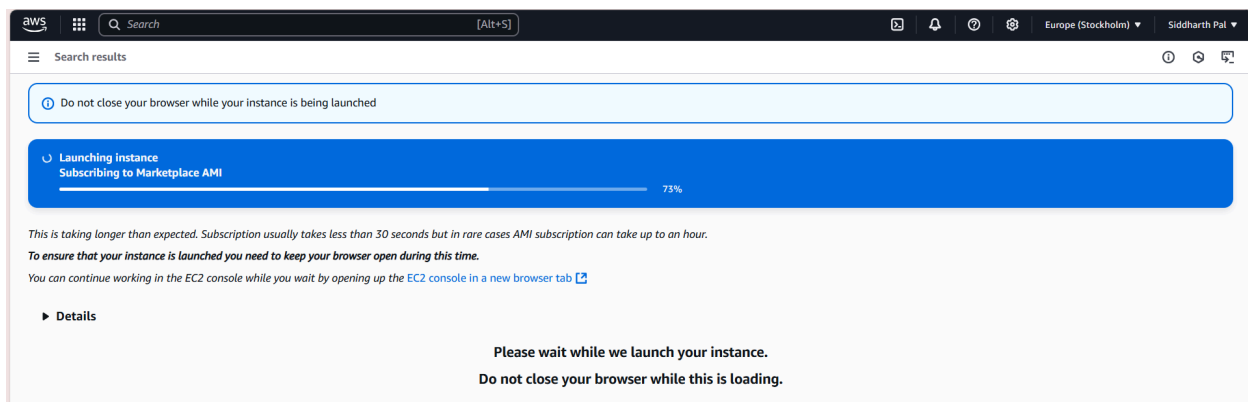
▼ Setting Up OpenVPN Server

✓ Step 1: Launch OpenVPN EC2 Instance

1. Go to **EC2 > Launch Instance**
2. Click "**Browse more AMIs**"
3. Search for: **OpenVPN**
4. Select the one with **\$0.025/hr** (from AWS Marketplace)
5. Click **Continue / Subscribe & Launch**

✓ Step 2: Instance Configuration

- **Instance Type:** `t2.micro`
- **Key Pair:** Use existing (`aws-vpn`)
- **VPC:** `VPC-SIT`
- **Subnet:** `web-subnet`
- **Auto-assign Public IP:** ☒ Enabled
- **Security Group:** Leave default (No changes)
- **Storage:** Change to `gp3`
- **Launch the instance.**



Do not close your browser while your instance is being launched

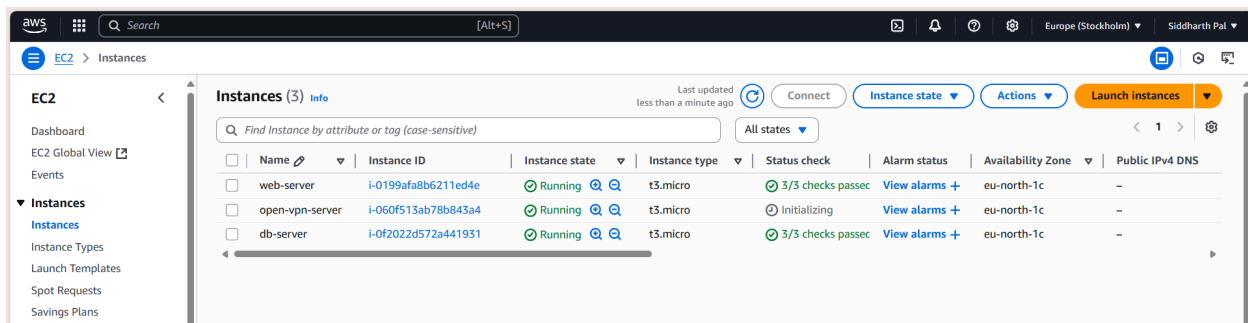
Launching Instance
Subscribing to Marketplace AMI

73%

This is taking longer than expected. Subscription usually takes less than 30 seconds but in rare cases AMI subscription can take up to an hour.
To ensure that your instance is launched you need to keep your browser open during this time.
You can continue working in the EC2 console while you wait by opening up the EC2 console in a new browser tab

Details

Please wait while we launch your instance.
Do not close your browser while this is loading.



EC2 > Instances

Instances (3) Info

Last updated less than a minute ago

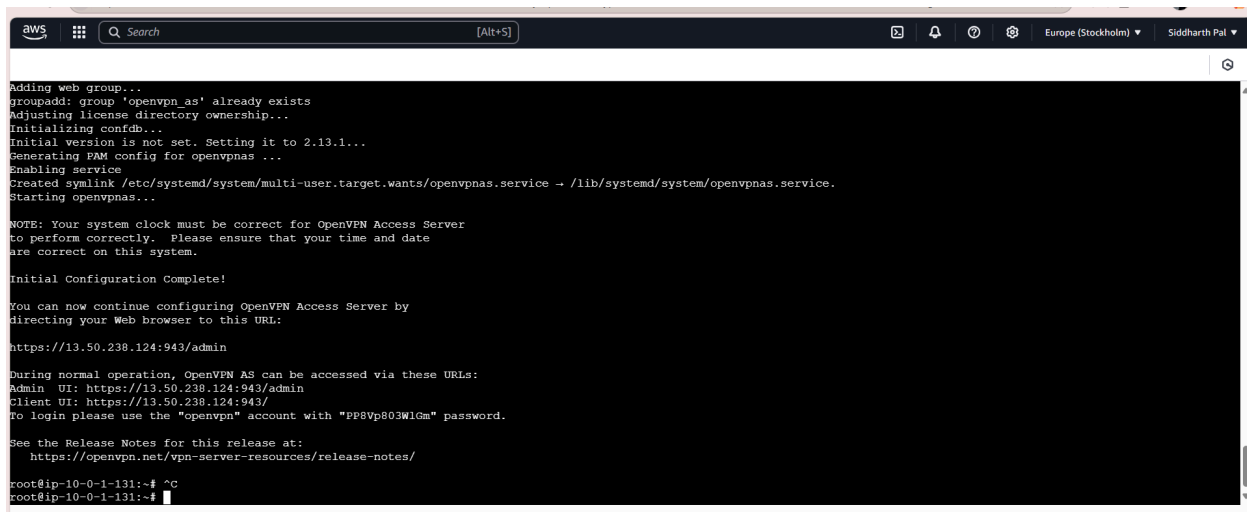
Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input type="checkbox"/>	web-server	i-0199afa8b6211ed4e	Running	t3.micro	3/3 checks passed	View alarms +	eu-north-1c	-
<input type="checkbox"/>	open-vpn-server	i-060f513ab78b843a4	Running	t3.micro	Initializing	View alarms +	eu-north-1c	-
<input type="checkbox"/>	db-server	i-0f2022d572a441931	Running	t3.micro	3/3 checks passed	View alarms +	eu-north-1c	-

Step 3: Access the OpenVPN Admin UI

1. Click **Connect > EC2 Instance Connect**
2. Accept the **Terms & Conditions** prompt.
3. Copy the **Admin URL**, **Username**, and **Password** from the instance output/logs.
4. Paste the **Admin URL** in your browser.
5. Login using the provided **Username** and **Password**



```
Adding web group...
groupadd: group 'openvpn_as' already exists
Adjusting license directory ownership...
Initializing confdb...
Initial version is not set. Setting it to 2.13.1...
Generating PAM config for openvpnas ...
Enabling service
Created symlink /etc/systemd/system/multi-user.target.wants/openvpnas.service → /lib/systemd/system/openvpnas.service.
Starting openvpnas...

NOTE: Your system clock must be correct for OpenVPN Access Server
to perform correctly. Please ensure that your time and date
are correct on this system.

Initial Configuration Complete!

You can now continue configuring OpenVPN Access Server by
directing your Web browser to this URL:


https://13.50.238.124:943/admin


During normal operation, OpenVPN AS can be accessed via these URLs:
Admin UI: https://13.50.238.124:943/admin
Client UI: https://13.50.238.124:943/
To login please use the "openvpn" account with "PP8Vp803WlGm" password.

See the Release Notes for this release at:
https://openvpn.net/vpn-server-resources/release-notes/

root@ip-10-0-1-131:~# ^C
root@ip-10-0-1-131:~#
```

Step 4: Create VPN User

- In the OpenVPN Admin Panel:
 - Go to **User Management > User Permissions**
 - Create a new user with desired **Username & Password**
 -  Save these credentials safely.

 **OPENVPN**
Access Server
v2.131

STATUS

CONFIGURATION

USER MANAGEMENT

User Permissions

User Profiles

Group Permissions


AUTHENTICATION

TOOLS

DOCUMENTATION

SUPPORT

Logout



POWERED BY  **OPENVPN**
© 2009-2024 OpenVPN Inc.
All Rights Reserved

User Permissions

Search By Username/Group (use '%' as wildcard)

No Default Group


Search/Refresh

Username	Group	More Settings	Admin	Allow Auto-login	Deny Access	Delete
openvpn	No Default Group		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
New Username	No Default Group		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Save Settings

Step 5: Configure OpenVPN Client on Your PC

1. Download & Install the **OpenVPN Client** on your PC.
2. Open it and paste the **Client UI URL** you got earlier.
3. Click **Next**, accept terms.
4. Enter your **Username & Password**
5. Click **Import** → then **Turn On VPN**
6. When prompted, enter the password again.

 **VPN is now connected**

The image shows two overlapping windows. The foreground window is the 'OpenVPN Connect' application, displaying a 'Profiles' screen. It shows a 'CONNECTED' status for the 'OpenVPN Profile' at 'openvpn@13.50.238.124'. Below this, there are 'CONNECTION STATS' including a speed of '3KB/s', a graph, and 'BYTES IN' of '1.19 KB/S' and 'BYTES OUT' of '368 B/S'. The 'DURATION' is '00:00:08' and 'PACKET RECEIVED' is '4 sec ago'. The user is identified as 'YOU' with the username 'openvpn'. The background window is a text editor showing the following text:

```
During normal operation, OpenVPN AS can be accessed via these URLs:  
Admin UI: https://13.50.238.124:943/admin  
Client UI: https://13.50.238.124:943/  
To login please use the "openvpn" account with "PP8Vp803WlGm"  
password.
```

Final Test: Access DB Server from Local PC

Now that you're on VPN, you can **SSH into the DB Server (private subnet)** directly from your PC terminal:

```
ssh -i aws-vpn.pem ec2-user@<PRIVATE-IP-OF-DB-SERVER>
```

```
ec2-user@ip-10-0-2-34: ~$ Permission denied (publickey,gssapi-keyex,gssapi-with-mic)
PS C:\Users\sidd2> cd downloads
PS C:\Users\sidd2\downloads> ssh -i "aws-vpn.pem" ec2-user@10.0.2.34
```

```
#_
~\_ ##### Amazon Linux 2023
~~ \#####\
    \|###|
    \|#/ --- https://aws.amazon.com/linux/amazon-linux-2023
      V~' '->
        ^^^^
          .-.
            / _/_/
             m/'
```

```
[ec2-user@ip-10-0-2-34 ~]$ |
```

 Success! You've bypassed public IPs securely using VPN + Jump Host setup.

Ayush Singh || 22BCSE63