



MERIK
SOLUTIONS

Establishment of SSH Tunnel to Access Private RDS

Using Termius on MAC

SSH Tunnel Setup for Accessing Private RDS Instance through Mac

Hello there! It's great to see you working on setting up an SSH tunnel to access a private RDS instance of a Postgres database using Termius on your Mac.

Problem

Before we begin, let me explain what we are trying to achieve. In this scenario, we have a **private RDS instance of a Postgres database running on an EC2 instance that is not publicly accessible**. To access this instance, we need to use a **bastion public EC2 instance**, which will act as an intermediate server between our Mac and the private RDS instance. We will be using **Termius**, an SSH client, to establish a secure connection between our Mac and the bastion instance, and then create an SSH tunnel to the private RDS instance.

Solution

Without further ado, **let's get started!**

Step 1: Launch the Bastion EC2 instance

The first step is to launch a bastion public [EC2](#) instance in the same VPC as the private [RDS](#) instance. Make sure the bastion instance has a public IP address, which will be used to connect to it from your Mac.

Step 2: Install Termius on your Mac

Next, download and install [Termius](#) on your Mac. Termius is a user-friendly SSH client that allows you to manage your SSH connections and tunnels.

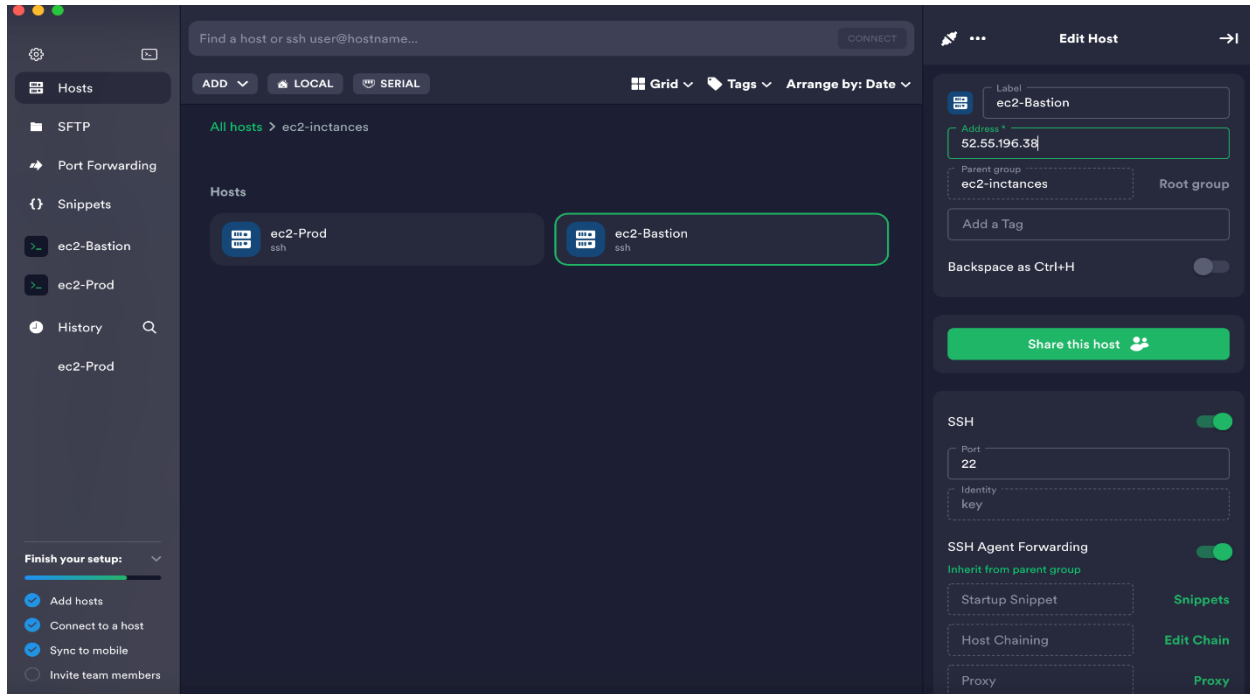
Step 3: Configure the SSH connection

Configure the SSH connection to the Bastion instance.

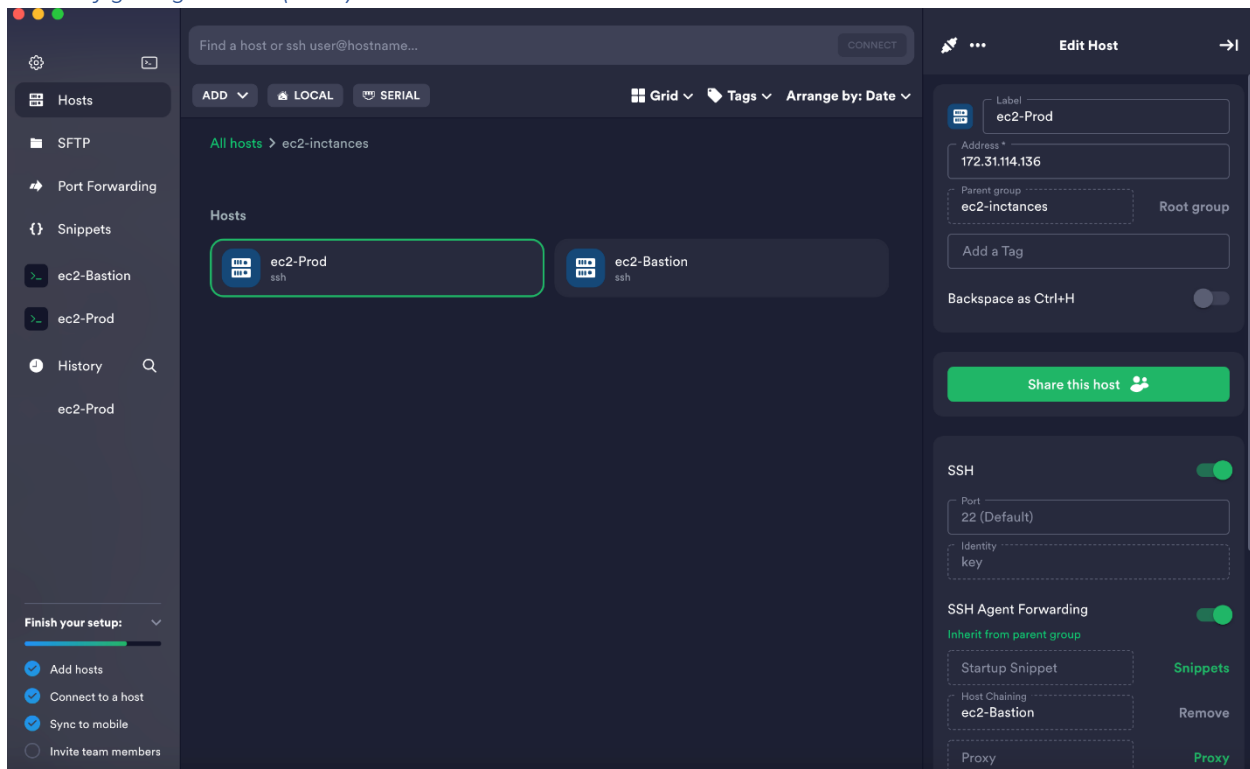
- Open Termius and create hosts for Bastion and Private instances. Enter the Bastion instance's public IP address, your username, and the path to your private key file. Click "Save" to save the connection.
- Do the same for the Private (Prod) EC2 instance except by entering the Private IP address for the Private instance.
- Keep both Public and Private instances in the same group and select that group in the "Parent Group" option just below the Address option in the right sidebar of the Edit Host Section. ***e.g Parent Group is "ec2-instances"*** in our case, which can be easily created by grouping your instances.

- While configuring the private instance, Chain the EC2-Bastion (public Instance) in Host Chaining Option under the SSH Section to establish a tunnel between public and private EC2 instances. e.g., In our case **Chain Host is “ec2-Bastion”**

3.1 Configuring Public (Bastion) EC2 Instance



3.2 Configuring Private (Prod) EC2 Instance



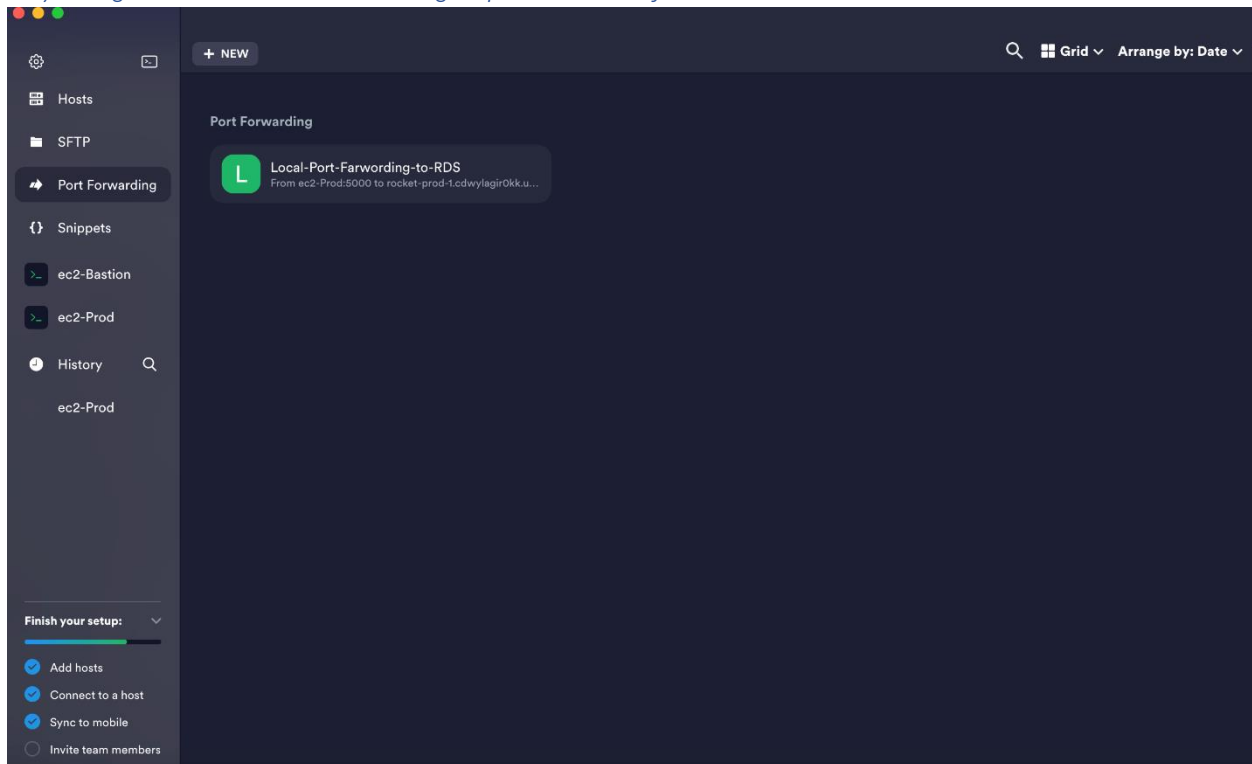
Step 4: SSH Tunneling and Port Forwarding

Finalize Tunnel by port forwarding. Click on the **“Port Forwarding”** option and configure it. Once the SSH connection to the Bastion instance is established, we can now create an SSH tunnel to the private RDS instance. In Termius, click the "Tunnels/Port Forwarding" tab and create a new tunnel. Enter the following details:

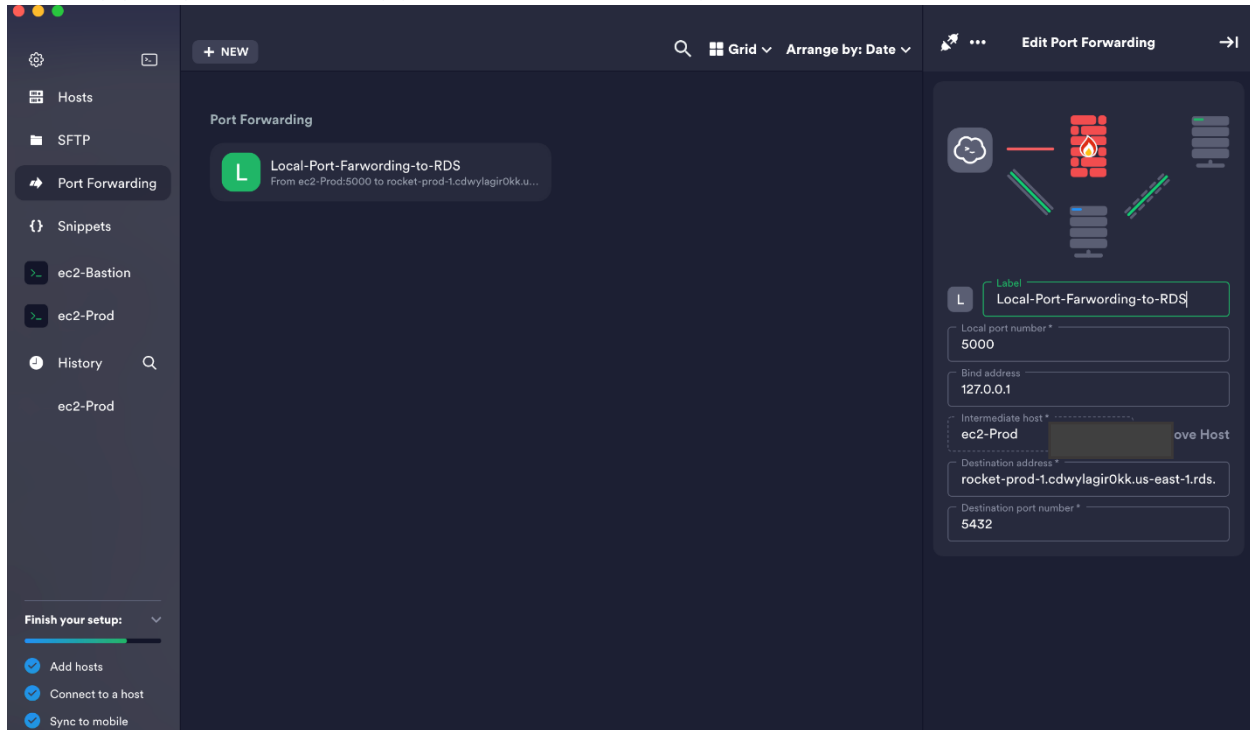
- Local Port: Choose a local port on your Mac that will be used to connect to the private RDS instance. For example, 5000.
- Bind Address: Add bind address 127.0.0.1
- Intermediate Host: Select your configured private EC2 instance named ec2-Prod in our case, in the **“Intermediate Host”** option.
- Destination Port: Enter the private IP address or Endpoint of the RDS instance. For example, Type <Your-Private-RDS-Endpoint> in the **“Destination Address”** option and Type 5432 in **“Destination Port Number”**.
- Source: Choose "Local" to indicate that the tunnel will be initiated from the local to the Bastion instance.

Click "Save" to save the tunnel configuration.

4.1) Navigate to the “Port Forwarding” Option in the Left-Side Menu



4.2) Configuring Port Forwarding (Local Port to Private Port)



Step 5: Connect to the private RDS instance

We can now connect to the private RDS instance from our Mac.

Open your Postgres client (e.g., psql or pgAdmin) and enter the following details:

- Host: 127.0.0.1
- Port: The local port you chose in Step 4 (e.g., 5432 or Local Port Forwarded to the EC2 Instance).
- Database Name: The name of the database you want to connect to.
- Username: Your Postgres username.
- Password: Your Postgres password.

5.1) Registering a new server

Register - Server

General

Connection

SSL

SSH Tunnel

Advanced

Name

TestConnectionToPrivateRDSThroughSSH

Server group

Servers

Background

X

Foreground

X

Connect now?

Comments

i




?

X Close

↺ Reset

Save

5.2) Establishing Connection

 **Register - Server**  

General

Connection

SSL

SSH Tunnel

Advanced

Host name/addresses

127.0.0.1

Port

5000

Maintenance database

postgres

Username

UserName

Kerberos authentication?

☐

Password



.....


Save password?


☐


Role

Service

 Close

 Reset

 Save

That's it! You should now be able to connect to the private RDS instance securely using the SSH tunnel.

Conclusion

In conclusion, setting up an SSH tunnel on top of a bastion public EC2 instance to access a private RDS instance of a Postgres database through a Private (Prod) Instance using Termius on your Mac is a straightforward process. By following the steps outlined above, you can establish a secure connection to your private database and protect it from unauthorized access. If you encounter any issues, feel free to reach out for assistance. Happy Learning!