**Step4a) Install Google Authenticator**

First task has to be done at your mobile where you need to download the Google Authenticator App from play store or App store

[Get it from Playstore](https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2&hl=en_IN)

[Get it from App Store:](https://apps.apple.com/in/app/google-authenticator/id388497605)

Once you have downloaded keep it ready and come back to the server to perform the next task.

**Step 4b) Run Google Authenticator on EC2 and Get QR code**

Once you have the Google Authenticator installed in your mobile you are ready to perform the second phase of this configuration.

the second phase should be performed at the ec2 server.

**Note:**

This step has to be executed as the user to whom you want to set the MFA or Multi factor authentication. ec2-user or root or ubuntu  or even as your personal userid which you have created in this EC2 instance.

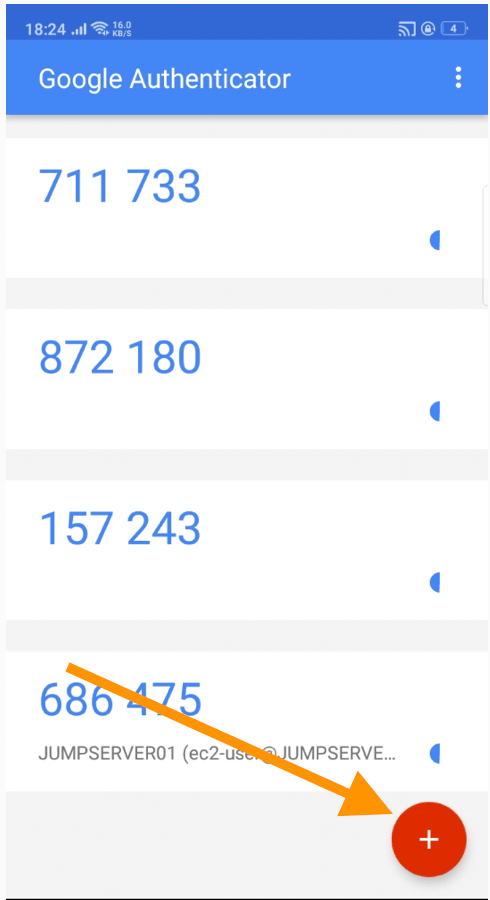
run the following command as the user of your choice in my case it is ec2-user

 google-authenticator

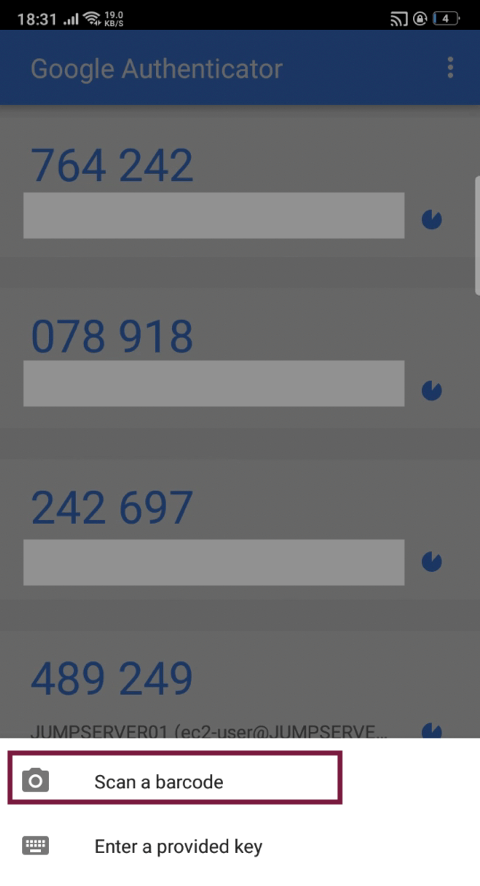
After the first question, it would show you the QR code and the Secret Key

**Step 4c) Scan the Shown QR code in your Google Authenticator App**

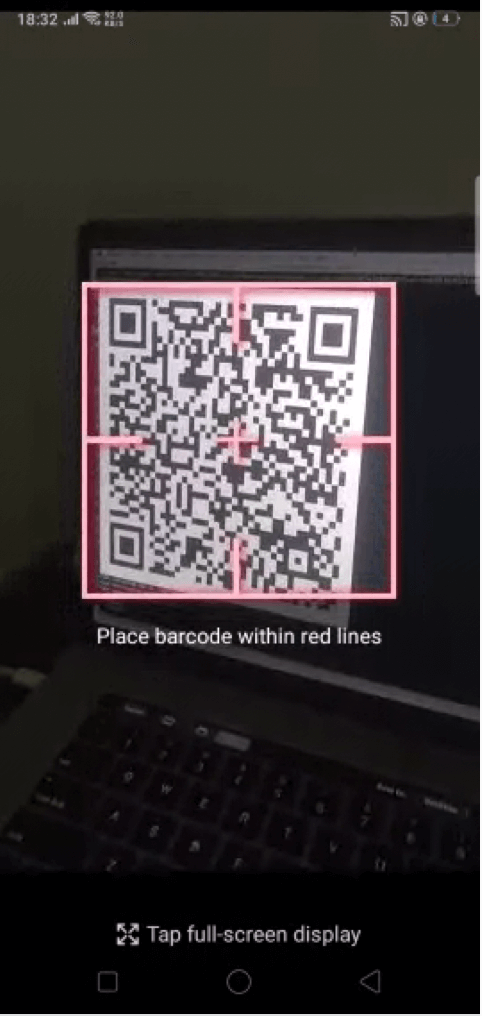
Now take your mobile and go to Google Authenticator App and click on the plus sign on the bottom right corner

[](https://www.middlewareinventory.com/wp-content/uploads/2020/02/AWSMFASetup-1.png)

Click on the Scan a barcode option

[](https://www.middlewareinventory.com/wp-content/uploads/2020/02/Screenshot-2020-02-08-at-6.31.33-PM.png)

Scan the barcode

[](https://www.middlewareinventory.com/wp-content/uploads/2020/02/Screenshot-2020-02-08-at-6.32.04-PM.png)

Once it has been added come back to your terminal

**Step 4d) Complete the Google Authenticator Setup in EC2**

Once you have crossed the QR code section, You would be prompted 4 more questions

Here are my answers. (Refer the following snippet)

[ec2-user@ip-172-31-83-181 ~]# google-authenticator

Do you want authentication tokens to be time-based (y/n) y

\*\*\*\*\*\*\* THERE WOULD BE A QR CODE DISPLAYED HERE \*\*\*\*

Your new secret key is: 2IAROUZWA6ZRSRRR89ZLYNZUC2A

Your verification code is 601376

Your emergency scratch codes are:

85535499

25397636

98473698

70322035

60012461

Do you want me to update your "/root/.google\_authenticator" file? (y/n) y

Do you want to disallow multiple uses of the same authentication

token? This restricts you to one login about every 30s, but it increases

your chances to notice or even prevent man-in-the-middle attacks (y/n) y

By default, a new token is generated every 30 seconds by the mobile app.

In order to compensate for possible time-skew between the client and the server,

we allow an extra token before and after the current time. This allows for a

time skew of up to 30 seconds between authentication server and client. If you

experience problems with poor time synchronization, you can increase the window

from its default size of 3 permitted codes (one previous code, the current

code, the next code) to 17 permitted codes (the 8 previous codes, the current

code, and the 8 next codes). This will permit for a time skew of up to 4 minutes

between client and server.

Do you want to do so? (y/n) n

If the computer that you are logging into isn't hardened against brute-force

login attempts, you can enable rate-limiting for the authentication module.

By default, this limits attackers to no more than 3 login attempts every 30s.

Do you want to enable rate-limiting? (y/n) y

That’s it. You have successfully set up Google Authenticator with AWS EC2 instance

Step5:  Restart SSH services  on the EC2 server

Now as the last step in our endeavour, we need to restart the sshd service as root

sudo service sshd restart

Step6: SSH to validate the AWS MFA setup.

Now use the same way you used to login to the EC2 instance.

If everything was done perfectly.  you would be prompted for a verification code

➜ ~ ssh -i ~/Downloads/mykeypair.pem ec2-user@3.95.13.122

Verification code: