How to Secure Your Instances with Multi-factor Authentication centos

### **Install Google Authenticator**

First, connect to your server and install the EPEL repo if it is not already installed:

# yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

Once you have installed the EPEL repo you can go ahead and install Google Authenticator

# yum install google-authenticator

### **Enable Google Authenticator**

# google-authenticator

The application will ask you whether the authentication tokens should be time-based or not. Since we are going to set up time-based tokens, enter y to continue.

Do you want authentication tokens to be time-based (y/n) y  
The application will generate a QR code on your screen which you need to scan with the smartphone application. Run the Google Authenticator application on your smartphone and scan the QR code.

The application will generate a new secret key, verification code and emergency scratch codes too. Keep these codes in a safe place.

Next, the application will ask you whether to save the key and the settings in the .google\_authenticator file for your system user. Enter y to continue.  
  
Do you want me to update your “/root/.google\_authenticator” file? (y/n) y  
The next question is whether you want to disallow multiple uses of the same authentication token. Each token will be valid for 30 seconds and will expire immediately after the use. Enter y to continue.

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  
If you don’t expect to have syncing issues between the client and the server, answer the next question with n.

By default, tokens are good for 30 seconds. 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. If you experience problems with  
poor time synchronization, you can increase the window from its default size of +-1min (window size of 3) to about +-4min (window size of 17 acceptable tokens).  
Do you want to do so? (y/n) n  
Next, you can enable rate-limiting for the authentication module. This will allow you to have 3 login attempts every 30 seconds so enter y to enable this feature.  
  
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  
By answering this question you are done with the Google Authenticator setup. The settings are saved in the ~/.google\_authenticator file and if you want to create a backup of this file you can copy it to a safe location.

### **Configure openssh (SSHD)**

# vi /etc/pam.d/sshd

Scroll down to the bottom of the file and add the following line:

#auth substack password-auth // make comment

auth required pam\_google\_authenticator.so // add this line at the end

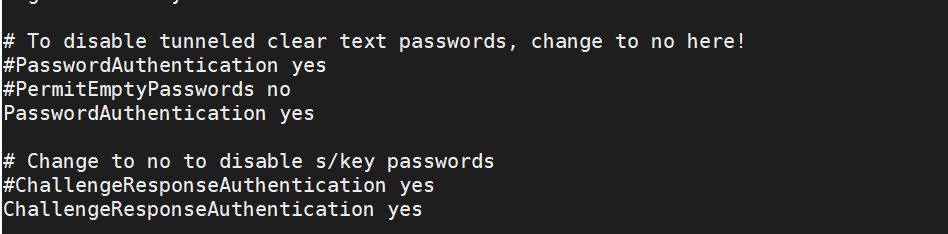
You also need to edit the SSH configuration file. Open this file:

# vi /etc/ssh/sshd\_config

Enable 2-factor Challenge Response Authentication and password authentication to yes

PasswordAuthentication yes

ChallengeResponseAuthentication yes



Save Now, restart de SSH service daemon and you’re good to go:

# service sshd restart

or

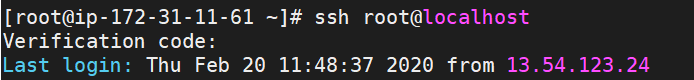
# systemctl restart sshd.service

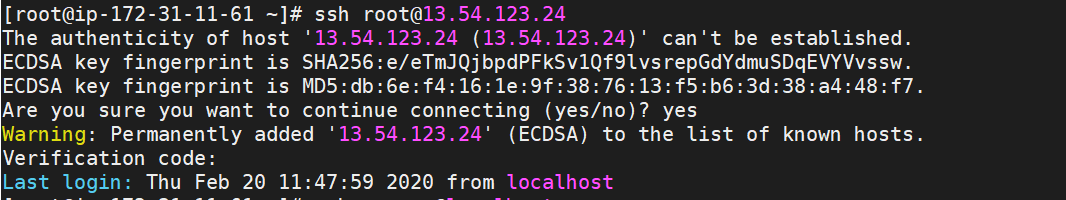
Try to login to server with ssh

# ssh root@localhost

Or

# ssh root@13.54.123.24





Link to follow

<https://www.schalley.eu/2019/02/09/adding-more-security-on-centos-7-with-google-authenticator/>

<https://aws.amazon.com/blogs/startups/securing-ssh-to-amazon-ec2-linux-hosts/>

<https://www.youtube.com/watch?v=nU3ZRvP26ZY>