CentOS7 FreeRADIUS Google Dual Factor Authenticator

# Introduction

Google Authenticator : authentication factor is a single piece of information used to to prove you have the rights to perform an action, like logging into a system. An authentication channel is the way an authentication system delivers a factor to the user or requires the user to reply. Passwords and security tokens are examples of authentication factors; computers and phones are examples of channels.

FreeRADIUS is a popular open source radius server. Radius is a standardized authentication system that can be used to authenticate many different devices including VPNs, Routers, Switches, Computers, and much more. For more information on FreeRADIUS see http://freeradius.org/.

## step1: SELINUX Setting

Before installations, I recommend setting it in permissive mode:-

vi /etc/selinix/config

Update SELINUX = enforcing to permissive

## step2: Prerequisites

Update your CentOS 7 and install Deployment Tool. You can run this commands to update your CentOS and for Deployment Tool installation.

yum -y update

yum groupinstall "Development Tools" -y

# step3: Installing FreeRADIUS

yum install freeradius freeradius-utils

# step4: Change user and group under which raddb would run from radiusd to root.

sudo vi /etc/raddb/radiusd.conf

Locate user and group

user = radiusd

group = radiusd

Update

#user = radiusd

#group = radiusd

user = root

group = root

# step5: Locate pam remove # placed before it. This is to enable pam in default site.

sudo vi /etc/raddb/sites-enabled/default

Locate pam

# Pluggable Authentication Modules.

# pam

*Update*

# Pluggable Authentication Modules.

pam

# step6: Enable pam module

ln -s /etc/raddb/mods-available/pam /etc/raddb/mods-enabled/pam

# step7: Configure RADIUS Client/Agent, edit clients.conf and add following line.

sudo vi /etc/raddb/clients.conf

remove # before ipv4addr = \*.

# step8 : Configure ‘users’

vi /etc/raddb/users

Locate the following

#DEFAULT Group == "disabled", Auth-Type := Reject

# Reply-Message = "Your account has been disabled."

#

*Update as follows*

DEFAULT Group == "disabled", Auth-Type := Reject

Reply-Message = "Your account has been disabled."

DEFAULT Auth-Type := PAM

# step9: *Execute following commands to open firewall port for RAIDUS service*.

sudo firewall-cmd --get-services | grep radius

sudo firewall-cmd --permanent --zone=public --add-service=radius

sudo firewall-cmd --reload

sudo firewall-cmd --list-service --zone=public

# step10: Execute radiusd in debug mode to verify authentication

sudo radiusd -X

# step11: start and enable freeradius to start at boot up.

systemctl start radiusd.service

systemctl enable radiusd.service

systemctl status radiusd.service

# step12: Installing Google's PAM

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

sudo yum install google-authenticator

# step13: Edit /etc/pam.d/radiusd

(comment all lines)

#@include common-auth

#@include common-account

#@include common-password

#@include common-session

auth requisite pam\_google\_authenticator.so forward\_pass

auth required pam\_unix.so use\_first\_pass

account required pam\_unix.so audit

account required pam\_permit.so

# step14: Add a user to the system

useradd <username>

passwd <username>

example:

adduser tom

passwd tom

# step15: change User

cd /home/tom/

su tom

then run

google-authenticator

**google authenticator create codes as below**

Your new secret key is: XQH7L6A7W6OK3JYS

Your verification code is 158428

Your emergency scratch codes are:

60469537

95985887

80580778

52386153

52951956

Do you want me to update your "/home/tom/.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, tokens are good for 30 seconds and 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 1:30min to about 4min. 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

# step16: change permissions to .google-athenticator

chmod 0600 /home/tom/.google-authenticator

# step17: Use radtest from radiusd-util package using the local unix account, tom.

radtest username userpassword with scratchcode hostname.com port sharedsecret

(scratchcode google-authenticator generated codes)

exanple:

radtest tom tom60469537 localhost 18120 testing123

Note:- password+google-authenticator generated code