Security workshop

Tacacs lab

Contents

1 getting the tacacs+ server configured

```
$ sudo apt-get install tacacs+
$ sudo groupadd -r cisco
$ sudo vi /etc/tacacs+/tac_plus.conf
```

1.1 change the following settings

- we want to set the shared key for routers who want to use our service to TacacsPassword
- We also want to limit access for users based on groups. For this example we will use settings in tac_plus.conf

1.2 change this line

```
key = TacacsPassword
```

In the real world we'd choose a much stronger shared key e.g.

```
$ pwgen -s 64  1
BRSWUWgJLkuxyqfmwfrlRC8JW54bpm3a2rMEe1IWwwpupwGBreGCXGTdbqkMGo2F
```

1.3 ... then at the end of the file .... add:

```
#
# "level 2" users who cannot "debug" or "config"
#
group = l2_tacacs_users {
    default service = permit
    login = file /etc/passwd
    enable = file /etc/passwd
    service = exec {
        priv-lvl = 15
    }
    cmd = configure {
        deny "."
    }
    cmd = debug {
        deny "."
    }
}
#
# "level 2" users with full privileges
```

```
#
group = netops {
    default service = permit
    login = file /etc/passwd
    enable = file /etc/passwd
    service = exec {
        priv-lvl = 15
    }
}
#
# group member with entry in password fileapt-
#
user = sysadm {
    member = netops
}
#
# group member not in password file
# use tac_pwd command to encode password
#
user = rancid {
    member = netops
    login = des GAxtUcNh5DBFQ
}
```

1.3.1 check tacacs_plus config

```
$ sudo service tacacs_plus check
```

You should see a response like:

```
 * Checking TACACS+ authentication daemon configuration files successful tacacs+
```

1.3.2 restart tacacs_plus to pick up the new settings

```
$ sudo service tacacs_plus restart
```

2 getting a cisco device to talk to your tacacs

Enter configuration mode:

```
tacacs-server host 10.10.9.1
tacacs-server key TacacsPassword
```

Check that you can reach the tacacs server and authenticate correctly:

```
test aaa group tacacs+ sysadm <password> port 49 legacy
```

You should see a response like:

```
Attempting authentication test to server-group tacacs+ using tacacs+
User was successfully authenticated.
```

2.1 Now you can finish configuring the router to use tacacs for login control:

```
aaa new-model

aaa authentication login default group tacacs+ enable
aaa authentication login NSRCCONSOLE local-case
aaa authentication enable default group tacacs+ enable
aaa authorization exec default group tacacs+ none
aaa accounting delay-start
aaa accounting exec default start-stop group tacacs+
aaa accounting commands 15 default start-stop group tacacs+

! This lets us login via the console even if tacacs isn’t working
username NSRCCONSOLE password 0 tpyPo9dT
line con 0
 exec-timeout 15 0
 login authentication NSRCCONSOLE
```

2.2 Now you can verify accounting

```
Router#show aaa sessions
Router#show aaa users all
```