Cisco IOS Privilege Levels and Parser Views

For this lab, please create a GNS3 file and name it appropriately.

Drag and drop 2 routers Cisco 3725 in the simulation. No connection is required and everything will be tested locally. The routers will be R1 and R2. On R1 we shall test privilege levels and on R2 parser views.

1. Cisco IOS Privileges

By default > is the privilege level 0 and # (enabled) is privilege level 15. When a user receives a privilege level x, (s)he can access all the commands defined on level x and below. The lab is straight-forward.

Open the console of R1. We need to have a running interface and we shall cheat it with a local interface loopback 1. In the configuration mode execute:

```
R1-conf#      interface loopback 1
R1-conf-if#   ip address 1.1.1.1 255.0.0.0
```

Let us activate the new model of authentication (in the configuration mode):

```
R1-conf#      aaa new-model
```

Let us define two users in the local database of users (the same would be on TACACS+):

```
R1-conf#      username moucha privilege 15 secret cisco
              username viktor privilege 5 secret black
```

Thus the user moucha with the password cisco has a privilege level 15 while viktor, with a password black has privilege level 5.

Then we need to specify that in order to access the console of the router, authentication and authorisation should be performed based on the local database of usernames.

```
R1-conf#      aaa authentication login default local
              aaa authorization console
```

Now we need to say that all the configuration command and the executed commands on the router must be first authorised (before execution):

```
R1-conf#      aaa authorization config-commands
              aaa authorization exec default local
```

Now we say that by default, the required level for the commands to be executed is 10.

```
R1-conf#      aaa authorization commands 10 default local
```

We now exit multiple times until the login prompt is presented. Let us login as viktor. After the login we see:

```
R1#           //and thus believe that we are privileged users but:
R1#           show privilege
              Current privilege level is 5
```

Let us run something:

R1# show run
R1# configure terminal

We should see: % Invalid input detected at '^' marker. - the commands do not exist for viktor even though by using ? You see that the commands are there in the list. Try the same commands for moucha. They are working.

The problem is now viktor cannot do anything useful. Let us assign him ping but not traceroute.

We need to login as moucha and then in the configuration mode type:

```
R1-conf#    privilege exec level 2 ping
            privilege exec level 10 traceroute
```

Thus for ping you need level 2 while for traceroute you need level 10.

Login as viktor and try ping 1.1.1.1 and traceroute 1.1.1.1

That is it.

The problem of privilege levels is that a command will be given a privilege level and a user will be given a privilege level and the user can execute the command if and only if the level of the command is lower or equal to the level of the user.

What if we have two users and we want them to be able to use the same commands (like for example to set an ip address) but not on the same interfaces. Well, privilege levels are no longer sufficient and for this we have parser views.

2. Parser Views

We now use router 2. What we want is that moucha should be able to set up an ip address on loopback 1 and viktor on loopback 2, ONLY. Thus the same command ip address can be executed by moucha on loop 1 but not on loop 2 and by viktor on loop 2 but not on loop 1.

In the configuration mode:

```
R2-conf#    aaa new-model
            enable secret cisco          //we set up a password for the enabled mode
            exit
```

Let us set up the interfaces:

```
R2-conf#    interface loopback 1
            ip address 1.1.1.1 255.0.0.0
            exit
            interface loopback 2
            ip address 2.2.2.2 255.0.0.0
            exit
```

Keep in mind that if the interface is not up, you cannot define it into a parser view.

```
R2#         enable view              //the password is cisco - this is the root view
```

You should see: %PARSER-6-VIEW_SWITCH: successfully set to view ‘root’.

Let us define the view for moucha:

```
R2-conf#        parser view MOUCHA-VIEW
R2-conf-view#   secret MOUCHA
                commands exec include ping
                commands exec include all show
                commands exec include configure
                commands exec include configure terminal
                commands configure include interface Loopback1
                commands configure include interface
```

So we gave moucha access to ping, all show commands, config and interface loop 1 and the password of that view is MOUCHA.

Let us define the view for viktor:

```
R2-conf#        parser view VIKTOR-VIEW
R2-conf-view#   secret BLACK
                commands exec include ping
                commands exec include all show
                commands exec include configure
                commands exec include configure terminal
                commands configure include interface Loopback2
                commands configure include interface
```

So we gave moucha access to ping, all show commands, config and interface loop 1 and the password of that view is BLACK.

Let us end and disable.

Now we enable the views:

```
R2>             enable view MOUCHA-VIEW          //password is MOUCHA

R2#             ?                                //and see which commands are available
R2#             configure terminal
R2-conf#        interface loopback 1
                exit
                interface loopback 2
                % Invalid input detected at '^' marker.

R2>             enable view VIKTOR-VIEW
                //try the same steps
                //loop 2 works but loop 1 does not
```

PS: On the interfaces you cannot do anything. To set the address you need to add to the list of parsed commands the “ip address”.

Try also show run. See what it is displayed.

We need to tie the parser views to usernames:

```
R2-conf#        username moucha view MOUCHA-VIEW secret cisco
                username viktor view VIKTOR-VIEW secret black
                username backup privilege 15 secret backup
```

```
R2-conf#        aaa authentication login default local
                aaa authorization console
                aaa authorization config-commands
                aaa authorization exec default local
```

We created the username backup because without this, moucha and viktor are now restricted to only the commands allowed in the parser views and cannot do anything else!!! Without the backup user we are essentially left out of the router config, except the interfaces Loopback 1 and 2.