15.8.2 Managing SELinux

Click one of the buttons to take you to that part of the video.

Managing SELinux 0:00-0:40

In this demonstration, we will show you SELinux or Secure Linux. SELinux allows us a more granular approach for securing Linux to the outside world, as well as our interactive sessions here.

The first thing we need to do is we need to be logged in as the root user. As you can see, I've already done that. We're going to be affecting the system as a whole so we could use sudo, but it's just much easier and much more efficient just to ensure that we are indeed logged in as the root user.

SELinux status 0:41-0:56

The first thing we want to look at is, "is SELinux already turned on?" We can run the ‘sestatus' command which shows us, "yes, it is enabled". Notice too that the current mode is enforcing. We'll get to that in just a moment.

SELinux mode 0:57-1:55

To check the mode of our current status inside of SELinux we can type getenforce and that will tell us whether or not we are enforcing the SELinux properties.

As you can see, we are. We can change that with this setenforce program and we can set that to permissive. By doing that, what we're doing is effectively turning off the protections and controls of SELinux. When we run getenforce again, you'll see it is permissive. To set it back, we can type ‘setenforce enforcing', we're turning everything back on.

Now the setenforce command can use 0 for permissive and a 1 for enforcing.

What we've done here is we've just made on the fly changes. Once we reboot those changes will be negated and whatever the configuration file shows, is what will then be enforced.

SELinux Types 1:56-2:39

So, if we change directories into at SELinux and we modify the config file. You can see inside of the config file that SELinux is enforcing and the Linux type is targeted. What that means is we are going to look at only certain processes and those are the ones that will be protected not anything else.

‘minimum' means just the modification the target policy only selected process are protected. Or we could use MLS which is multilevel security protection. We're going to look at the targeted type in this demonstration.

---

getsebool 2:39-4:45

So, quit out of here and then I'm going to go ahead and run a program called getsebool. Getsebool is Boolean, meaning is it on or is it off.

Now I'll go ahead and show all, and if we look at that, you're going to see that there are a ton of different Boolean settings for different processes.

To help, we'll just look at the samba processes. So, let's do a getsebool, we'll look at all of them, but this time we'll use grep to filter only samba processes. And you can see there's a few there. The one that them to look at here is ‘samba_enable_home_dirs'.

Samba is the is the process that allows Linux to look like a Windows server and share files with Windows. So, I don't want to enable home directories to be advertised on samba. So, let's turn that off and the way we do that is with setsebool. ‘setsebool' and now what we do is we specify the name of the Boolean. So, in this case, it ‘samba_enable_home_dirs', but we're going to set it to off, we're going to have it equal off. So that's done.

Now let's go ahead and run the command that we ran before, and you'll see that ‘samba_enable_home_dirs' is now off.

Again, this is a temporary setting. If we wanted it to be permanent, then we can type the same command ‘setsebool'. Now we need a -P and that tells us that we're going to make this permanent, and notice it is a capital P.

So, we're going to do the exact same command, ‘samba_enable_home_dirs=off'. Press ‘enter' and there you go. Now when the system reboots that will be set to off.

SELinux Permissions 4:45-7:28

The last thing that I wanted to show is, when we're using SELinux, files take on different permissions with SELinux.

Let's go ahead and change to my home directory and we'll take a look and see that really nothings there. I'm to go ahead and ‘touch file1', just to create a file.

Now we'll take a look at the additional SELinux components. So, we'll do an ‘ls -Z file1'. You can see from the beginning, that the permissions are standard, just what we've had before. This one is read/write for the user. Read, for group. And read, for everybody else. It's owned by root and it's owned by the group root.

But notice we now have additional information here. So now what we have is we have the user type, the role base, as well as the type of file this is.

We can change that information, and there might be different reasons to change it. Who can access the file, who can change the file. We can compare it with another file and set the changes to it to those.

What it does is it gives us an additional granularity of control. But just to show and give a demonstration on how to make a change, we can use the change context command, chcon.

In this case we're going to go ahead and change the type from the admin_home type to a user_home type.

Let's go ahead and make that change. We're going to say ‘chcon'. We're going to use -t. What -t does is tells us we're only changing the type, we're not changing the role, we're not changing the user type.

We'll go ahead and type user_home_t for type and we need to specify the filename. So, by doing that all that we effectively changed was this type right here.

Now when we do ‘ls -Z' for ‘file1', you see that we've changed the type from admin_home to user_home. But that's really a mistake and what we really sometimes need to do, is we just need to start over. We can restore the context with the command that just brings everything back to where it was.

So, it's ‘restorecon', specify the file name, and it just brings us back to where we were. So, when we do our settings again, it shows us that indeed it changed back to admin_home.

Summary 7:29-7:46

In this demonstration we showed you the SELinux commands including sestatus, getenforce, setenforce, and then we change the Boolean's with getsebool and setsebool. And then we used chcon for changing context and restoring context.