# 14.1.3 Executing and Sourcing a Script

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

Executing and Sourcing a Script 0:00-1:09

In this demonstration, we will show the difference between executing and sourcing a script. In addition, we'll show that the necessary pieces for a scripter in place, as well.

First thing we need to do is create our script so I'll go ahead and list files in the directory just to show that there are no scripts. Let's go ahead and create our script. We'll use the 'vi' editor and will just call this my 'script.sh'. We have to start our script as all others. We enter the directive to show this is a script, and we're just going to enter one command in here. It's just going to be 'echo $$'. What this does is this displays the current process ID that is being used at that time. We'll go ahead and write, then quit our script. We now have the script, 'myscript.SH'.

Now we can't run it currently, because the permissions on the file don't allow execution.

---

Change permissions 1:04-1:58

To show that we can do a long listing and you'll see for 'myscript', the permissions here do not have the execute permission. Let's go ahead and add that. We'll do that by providing the change mod command, so 'chmod', all plus 'x'. That means were going to add the execute permission on all parameters for 'myscript.sh'.

Now when we do an ' ls -s', you'll see that the file has changed to a green color which, in CentOS is typically meaning that it is an executable. Again, here you can see that indeed, the permissions are now correct.

---

Parent Process vs. Shell Process 1:59-2:38

Let me run the 'ps' command just to show processes. Notice that my bash process ID is 11630. I'll go ahead and execute my script. It's in my current directory so I must use the dot slash (./). That means look in this directory, and I'll execute 'myscript.sh', press enter, and notice that the process ID is different. It's 11894. Well that's different than my current bash process ID. If we run it again, just to show you, the process ID does change.

---

Source a Script 2:39-3:34

Now, if I source the script, meaning I'm not executing it in its own shall, I'm executing it in my current shall. I do that just by entering a dot (.), meaning use this directory, enter my script, and notice that indeed the process ID is the same 11630. What that means is every time I run a script, by definition, I'm going to open a new shell. That's where that directive of the '#!/bin/bash' comes from. It means open a new shell. That's what happens when I execute a script. When I source a script, meaning I'm just extending what I currently am using, to run an application or execute something, it uses my current environment.

---

Summary 3:35-3:41

In this demonstration, we demonstrated creating a script, providing its permissions for execute, and then executing the script, and sourcing script.

---