PLURALSIGHT | HELP CENTER

SSH on Linux (syntax and passphrases)

Tags: ACG

SSH is the predominant remote management tool for administering Linux systems.

Whether you are working between linux servers or from your Mac (which is basically pretty Linux) you will be using ssh to get a terminal/command line on the remote server. A few people get stuck on this if they are learning to do it for the 1st time.

Let me start by referring you to the AWS document on troubleshooting SSH connections There are some recommendation there you should probably look thought.

For beginners the best place to start is the AWS EC2 Instance Connect (opens in new tab) utility which can be used from the AWS Console. This bypasses the need for other tools, but on the down side you don't get to learn how to do it the hard way unless you want to.

For those interested in other methods that you might need to use in your workplace here is some additional reading:

- http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html (opens in new tab)
- https://aws.amazon.com/premiumsupport/knowledge-center/linux-credentials-error/ (opens in new tab)
- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/TroubleshootingInstancesConnecting.html (opens in new tab)

To gain the most information about the problem add a -v to the command to get the error in verbose mode.

Here are the common issues that other students have had:

1. Key does not have the right permissions attached. If it is too loose SSH will refuse to use it for security reasons. The recommendation is chmod 400.
   `sudo chmod 400 /path/my-key-pair.pem`
2. Using the right IP address for the target. For the labs keep it simple and use the IP address not DNS names (various reasons) once you have it working experiment with the DNS names. In more complex environment you should use the DNS names, however while you are learning, keep it simple.
3. Security Groups not correct to allow network access to or from the bastion host.
4. Wrong type of key. PuTTy (opens in new tab) and PPK keys are only used when you are connecting from a Windows server. If you are not using Windows and PuTTy, use the .pem key and the ssh format.
   `ssh ec2-user@##.##.##.## -i my-key-pair.pem`

   or

   `sudo ssh ec2-user@##.##.##.## -i my-key-pair.pem`

or

`sudo ssh ec2-user@##.##.##.## -i/path/my-key-pair.pem`

5. Inadequate personal right.
   When SSH asks for the passphrase it is doing so to ensure that you have the right to use that key. You can prove that you do by providing the passphrase. However for our labs that should not be necessary. If you run the ssh command with root permissions, it is assumed by the operating system that you have the necessary rights. To do this you can pre-pend `sudo` in front of the ssh this tells the operating system to give you root permissions for this one command. The other options is to run `sudo su` previously to attempting ssh. This tells the operating system to give you root permissions until you leave or cancel the command. As you watch the lectures you will see the lecturers do this sometimes at the beginning of the lab so that they have all the access needed to complete the lab.

6. Sometime people try to use the wrong user account. For the basic AWS AMI the user is **ec2-user**. For other Linux distributions or custom AMIs the default username may be different (root, ubuntu, bob, etc) but you will not have to worry about that until you are much more proficient.

7. From time to time someone accidentally types **-1** instead of **-i**. The -1 forces the SSH session to use the old SSH v.1 version. Which is no longer supported. Take a few minutes to read of the MAN page to understand what you have accidentally called.
   https://linux.die.net/man/1/ssh (opens in new tab)

8. From time to time based in random chance you might get a new instance with the same IP address as you have previously had. You will get a warning similar to this.

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@ WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!@

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!

Someone could be eavesdropping on you right now (man-in-the-middle attack)!

It is also possible that a host key has just been changed.

The fingerprint for the ECDSA key sent by the remote host is

SHA256:31qsIOAomqwC2LN2f7I2UJYK/MXYb0WuV8pxxxxxxx.

Please contact your system administrator.

Add correct host key in /Users/moveon/.ssh/known_hosts to get rid of this message.

Offending ECDSA key in /Users/moveon/.ssh/known_hosts:106

ECDSA host key for ##.##.##.## has changed and you have requested strict checking.

Host key verification failed
```

This is basically saying that your SSH client already has a fingerprint for that IP address, and this does not match. You can clear this by editing the file **/Users/moveon/.ssh/known_host**. Remove the line for the IP address shown in the warning, and you will be fine.

Here is a couple of good short video that take a broader look at SSH and AWS Session Manager which may will help you better understand the what the elements do and a bit more about the background.

https://acloud.guru/series/acg-fundamentals/view/intro-ssh (opens in new tab)

You might also want to look at using AWS Session Manager (opens in new tab). It offer both centralized access and a more secure connection to internal system without a Bastion host/jump box.

https://acloud.guru/series/acg-fundamentals/view/3b09d9fc-e559-a044-4b7c-cc7f9b7bb5c3 (opens in new tab)

If you need help, please email Pluralsight Support (opens email form) for 24/7 assistance.