



Security

Initial conditions/prerequisites:

- ssh (secure shell) must be installed on each system (type ssh for a test, if list appears ssh is installed)
- obtain user names and IP addresses on you system as well as your neighbor's.

Goals:

- Connecting successfully to a different system using the ssh command.
- Able to login to a system using the ssh command without providing a password.
- Be able to shutoff Linux services.

This lab is designed to give an experience in using ssh to log into a system given an IP address to the system is provided. The ssh command starts a connection between two or more systems.

Procedure	Explanation / Background
<p>If you have not yet done so, create a username and password for yourself on your system typing (must be a super user):</p> <pre>→ useradd "yourname" <enter></pre> <pre>→ passwd "yourname" <enter></pre>	<ul style="list-style-type: none">• Login as superuser for you to create a username and password.• Do not include " " when you type your user name.• <enter> means press the enter key.
<p>Now you will create a username and password for yourself on your neighbor's system typing (must be a super user):</p> <pre>→ useradd "yourname" <enter></pre> <pre>→ passwd "yourname" <enter></pre>	<ul style="list-style-type: none">• As your neighbor to log in as a superuser for you to create an account on his/her system.• A different useraccount name is suggested to clarify on what system you are logged in.
<p>To obtain the IP address of a system you must first be logged in as a superuser. Once you are a superuser type the following command to display the IP address of your system:</p> <pre>→ ifconfig <enter></pre>	<ul style="list-style-type: none">• An inet address will appear, this address is your IP address (its best to write this address, you will use it)

Procedure	Explanation / Background
<p>The same procedure of obtaining an IP address is to be followed on your neighbor's system. Again be sure your neighbor is logged in as a superuser to be able to display the IP address of his/her system.</p>	<ul style="list-style-type: none"> • Copy your neighbor's IP addresses as well, it will also be used frequently.
<p>First you will use ssh to login to your newly created account on your neighbor's system. Before using ssh you may “ping” your neighbor's IP address to realize a connection between the two systems. Make sure you are using <i>your neighbor's IP address</i>.</p> <p>→ <code>ping 128.3.5.211 <enter></code></p>	<ul style="list-style-type: none"> • When pinging use the Ctrl-C to return to command line. • If you receive an error saying there is no route to host, then probably the host (your neighbor's system) is not running. • The useraccount is the account you made on your neighbor's system. Enter the password you created when prompted to do so.
<p>Now that you are logged in to a different system, you try and log back to your own. Type:</p> <p>→ <code>ssh useraccount@yourIPAddress <enter></code></p>	<ul style="list-style-type: none"> • You should notice a prompt asking for a password to log in to your system.
<p>From the command line on your system create a file that will be passed to your user account on your neighbor's system containing your name within.</p> <p>→ <code>touch testfile <enter></code></p> <p>→ <code>vi testfile <enter></code></p>	<ul style="list-style-type: none"> • touch will create an empty file in your present working directory. • You are to enter your name within the testfile, write the file and exit.
<p>The file created (testfile) will now be passed to your account on your neighbor's system using the scp command.</p> <p>→ <code>scp testfile useraccount@neighborIPAddress:./ <enter></code></p>	<ul style="list-style-type: none"> • When asked to provide a password do so. The file is copied to the ./ directory within your account on your neighbor's system.
<p>Follow the previous procedure of logging into your neighbor's system using the ssh command.</p> <p>→ <code>ssh useraccount@neighborIPAddress <enter></code></p>	<ul style="list-style-type: none"> • You may check to see if the file you copied from the useraccount on your system was copied to your neighbor's system by using the ls command. • The testfile you created should be present, open it to view its contents.

Now you are finished practicing using the ssh command and tired of providing a password everytime you log in or send a file, an exercise that will allow you to not provide a password when you log in to your neighbor's system as a user would do you much good.

Procedure	Explanation / Background
<p>Before you can log in without providing a password there are a number of things that must be done. First you will create the file called ssh_config on your system:</p> <p>→ touch ssh_config <enter></p>	<ul style="list-style-type: none"> The file you created ssh_config is made to replace another ssh_config file.
<p>Become a superuser before making any changes. To the file created add the following two using the vi editor:</p> <pre>Host * ForwardX11 yes</pre>	<ul style="list-style-type: none"> Make sure to write the file and return to command line prompt.
<p>Copy the file ssh_config you have just created to the following directory as follows:</p> <p>→ cp ssh_config /etc/ssh/ <enter></p>	<ul style="list-style-type: none"> The copy that you created of the file replaces the one that already exists. If you are asked to overwrite old file type yes (you must be a superuser to do this) .
<p>On your system you will now use the ssh-keygen command to create a public key and a private key, keep the default name id_rsa (do not provide a passphrase in the process):</p> <p>→ ssh-keygen <enter></p>	<ul style="list-style-type: none"> The following command is a key generator, creating a public and a private key defining your system.
<p>The public key created is to be copied to your useraccount on your neighbor's system using the scp command:</p> <p>→ scp ~/.ssh/id_rsa.pub useraccount@neighborIPAddress:~/.ssh <enter></p>	<ul style="list-style-type: none"> A space exists after id_rsa.pub. The scp command is written as follows: scp (source of file) (destination of file)
<p>You will be prompted to provide a password, do so and use the ssh command used previously to login to the useraccount on the other system:</p> <p>→ ssh useraccount@neighborIPAddress <enter></p>	<ul style="list-style-type: none"> Check to see if the id_rsa.pub file was copied successfully to your useraccount on your neighbor's system.
<p>The following step allows the public key you copied from your system to be part of an “allow” list or the authorized_keys file:</p> <p>→ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys <enter></p>	<ul style="list-style-type: none"> You may open the authorized_keys file to check to see if the key generated on your system is included.
<p>Update the permissions of the authorized_keys file using the chmod command:</p> <p>→ chmod 600 ~/.ssh/authorized_keys <enter></p> <p>→ exit <enter></p>	

Procedure	Explanation / Background
<p>You should now be logged in to your account on your own system, if you are not close the terminal and start a new one. From the command line type the following (do not provide any passphrase in the process):</p> <p>→ <code>eval `ssh-agent` <enter></code></p> <p>→ <code>ssh-add ~/.ssh/id_rsa <enter></code></p>	<ul style="list-style-type: none"> The following commands will allow passwordless access to your neighbor's system and are registering your key with the ssh-agent.
<p>Log in to your account on your neighbor's system:</p> <p>→ <code>ssh useraccount@yourIPAddress <enter></code></p>	<ul style="list-style-type: none"> You should not be asked to provide a password and should be logged in to the system automatically !

Shutting off Linux Services Section: Here you will be able to do a number of operations on processes running on your system. (If you are familiar with the windows task manager, this lab should be contrasted with operating the windows task manager).

Procedure	Explanation / Background
<p>You are logged in as root on your system. To view all the processes running you type:</p> <p>→ <code>ps -A<enter></code></p>	<ul style="list-style-type: none"> The ps command reports the status of each process on your system. If you do a man page on ps you will find that -A is an option that displays all the processes running on your system.
<p>Start the mozilla browser by typing:</p> <p>→ <code>mozilla <enter></code></p>	<ul style="list-style-type: none">
<p>Use the ps command to display all the processes running on your system, and locate the process PID containing the mozilla-bin application.</p>	<ul style="list-style-type: none"> You can use the previously mentioned ps -A command. The PID number is located to the left of the process name.
<p>Once the PID is found you will stop the mozilla application from running using the command:</p> <p>→ <code>kill -STOP (PID) <enter></code></p>	<ul style="list-style-type: none"> Make sure to not include the () when you type in the PID. Notice on your screen that the mozilla application no longer is running.
<p>If you wish for your mozilla application to continue once it has been stopped type:</p> <p>→ <code>kill -CONT (PID) <enter></code></p>	<ul style="list-style-type: none"> Again make sure to not include () when you type the PID. CONT commands the process to continue once it has been stopped by the STOP command. Refer to the mozilla browser you started earlier, it should now be running.

Procedure	Explanation / Background
<p>Now you are finished with the mozilla browser and you wish to close it, type:</p> <pre>→ kill (PID) <enter></pre>	<ul style="list-style-type: none">The browser should close as you press the enter key. This command can be helpful if you are trying to close a window using the graphical without success. When using the kill command make sure you know what process you are “killing” . You may kill a process that will freeze your system, such as the Xfree86 process.

Stuff to do when you're done with the lab

Here are some things to try if you finish early.

1. Use the man pages to on *ssh* and test other options with *ssh* not used in this lab.
2. Man page *service* aswell and use other options not used within this lab.

Lab Worksheet

Name: _____

Date: _____

Find and write down the IP address of your system. What command did you use?

What command is used to copy a file from one system to another across a network?

What option part of the kill command is used to stop a process? To continue a process ?

List a new option you learned while searching the man page of the ps command and its function:

The ssh-keygen command has the function of doing what ?

Is it possible to use ssh without being a superuser ? Explain
