SSH AGENT LAB

**Overview**

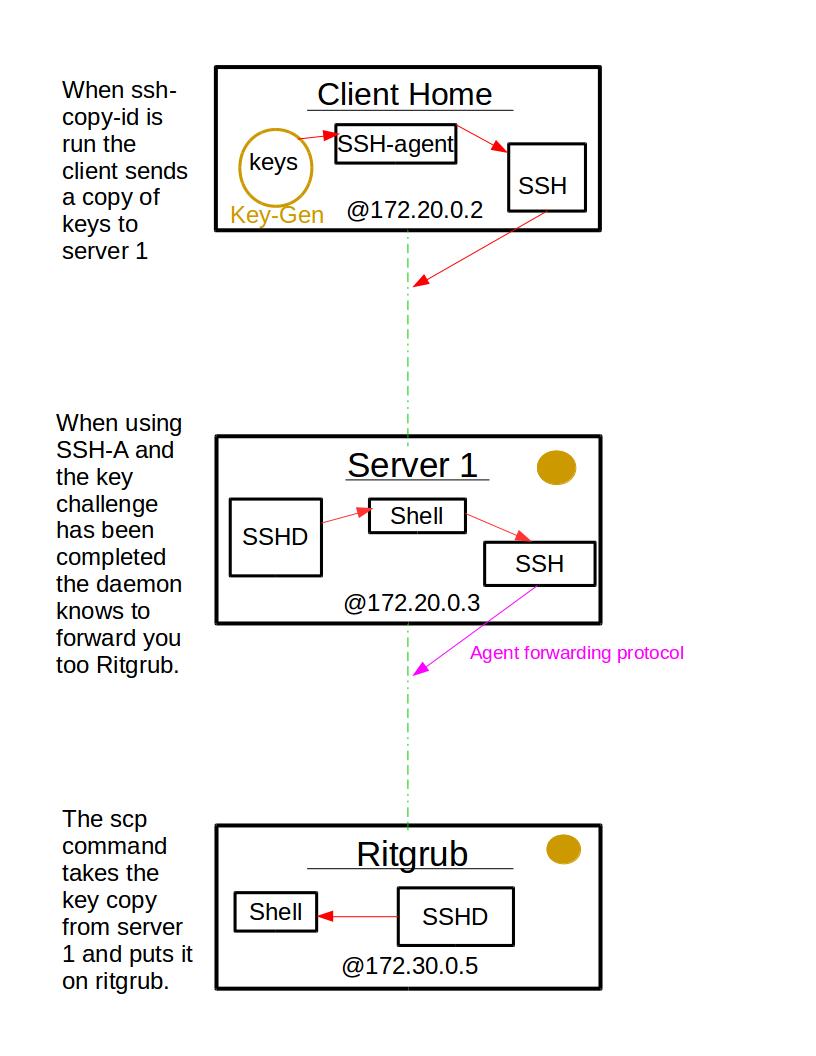
this lab was developed for the Labtainer framework by the Naval Postgraduate School, Center for Cybersecurity and Cyber Operations under National Science Foundation Award No. 1438893. This work is in the public domain, and cannot be copyrighted. It covers the use of ssh-agent principles.

**Performing the lab**

the lab is started from the labtainer working directory on your Docker-enabled host, e.g., a Linux VM. From there, issue the command:

labtainer pythonlab

the resulting virtual terminal should be a terminal connected to a blank client terminal and a terminal and connected to “ritgrub” and a link to these instructions.



**Tasks**

1. **Generate authentication keys (public/private RSA key pair) on the client computer input**:

ssh-keygen -t rsa

This will create a private key and its corresponding public key and place them in your .ssh directory.

When prompted for a password put “password”

2. S**etup SSH on server for user to use their authentication keys**

The server's IP address is 172.20.0.3. The user is "ubuntu", and the user's password is also "ubuntu".

On the client computer:

* use this command:

ssh-copy-id -i ~/.ssh/id\_rsa.pub 172.20.0.3

* When prompted to "...continue connecting (yes/no):, type "yes".
* Provide the user password when prompted.

The above step will copy the public key to the ~/.ssh/authorized\_keys file on the remote sever.

**3. SSH to this server**

the server’s IP is 172.20.0.3 . use the password “password” when prompted.

Exit when you are done.

**4. start and set up the SSH-Agent**

start the agent with the command: eval `ssh-agent`

Note `ssh-agent` must be between 2 back ticks.

Add the ssh keys to the server via: ssh-add. When prompted use the password: ubuntu .

Then activate the daemon on 172.20.0.3 so it knows you will be using an Agent. This is accomplished with the command: ssh -A 172.20.0.3 .

if all goes well you were able to reach this server without inputting your password.

SSH to ritgrub @ 172.30.0.5

use the password “ubuntu” when prompted.

Return to the client terminal and ssh to 172.20.0.3 .

Once there issue the command the scp -r .ssh 172.30.0.5 . then type the password : password. this will send your key information over to the ritgrub server.

Exit the server.

Then ssh -A to 172.20.0.3 and ssh to ritgrub.

**5. creating the ssh config file**

When using ssh within the .ssh directory (on the client) there is a hidden file (that must be created) called config. This config file allows for bonus customizations to the ssh settings of the current environment(s).

**6. editing and using the conifg file**

To begin, you must specify which Host you will be using these configurations on (on the client). You can specify this by typing Host fred. On an indented line below you must again specify the HostName <172.20.0.3>. Then on another indented line put ForwardAgent yes. That last command will enable ssh-agent forwarding without the need of typing ssh -A to awaken a daemon. Now test that this is so.

**Stop the Labtainer**

when the lab is completed, or you’d like to stop working for a while, run:

stoplab telnetlab

from the host labtainer working directory. You can always restart the labtainer and continue your work where you left off. When the Labtainer is stopped, a zip file is created and saved to a location displayed beneath the stoplab. When you are completely finished send that file to your instructor.