Labo Linux Essentials

Trainer:
Name: Sander Descamps
Email: sander.descamps@axxes.com
Email: sander_descamps@hotmail.com

Trainee:
Name: _____
Email: _____

Labo

Deploy new Linux vm (SRVLAB01)

1. Get the ISO for Ubuntu 18.04.3 (desktop or server).
2. Open VMWare Workstation/vCenter and create a new vm
   - Name: SRVLAB01 (or follow your own naming convention)
   - 1 cpu
   - 2 GB RAM
   - server: 5GB HDD | desktop: 20GB HDD (thin provision)
   - Connect to a vlan with internet connectivity
   - Attach the Ubuntu ISO

3. Boot from the Ubuntu ISO
4. Set language
5. Set keyboard layout
6. Leave network on DHCP
7. Leave the proxy configuration blank
8. Keep default mirror server
9. Storage: Use An Entire Disk
   (note: If you are comfortable disk partitioning you can manually configure 2 partitions. One partition of 400MB for /boot. A second partition with LVM for the root partition.)
10. Select disk and accept default settings
11. Set hostname and configure a (admin)user
    note: Keep in mind the keyboard layout affect your password,
12. Enable “Install OpenSSH server” (spacebar)
13. Skip Features server snaps
14. Wait for the installation to fully finish (1-5min)
15. Reboot and remove the ISO
16. It is a good idea to make a snapshot after the reboot. If you make a mistake, than you can easily turn back. (For windows trainees: create a Veeam Backup)

Basic Commands

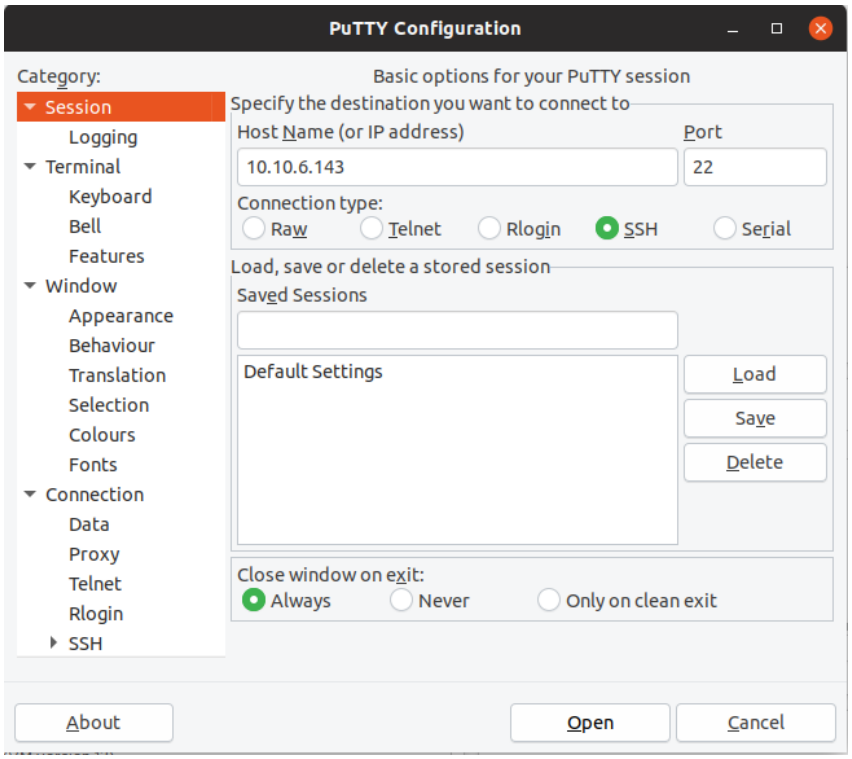
1. Connect to the machine via the vmware console and login.
2. Check the current kernel version
3. Check the os release file. What is the full version of the os you're running.
4. Navigate to /tmp/
5. Create a new file. Call it 'newfile.txt'
6. List all files and make sure the file is present
7. Pipe the output of the list command into the file
8. Print newfile.txt on the console
9. Print newfile.txt but only display the line which contains 'newfile.txt'
10. Make a copy of newfile.txt and call it 'copyofnewfile.txt'
11. Update the file newfile.txt with the current list of the /tmp directory
12. Rename newfile.txt to updatednewfile.txt
13. View the difference between updatednewfile.txt and copyofnewfile.txt, is the output very clear?
14. Check the man page of the command you used and look for something more clear.
    man <command>
15. Copy both files to your home directory
16. Navigate to your home directory
17. Display the command history and save it into a file called comhist.txt (1 command)

Connect via SSH and update

1. Check the ip address and take note (ip a)

   _____

2. Verify that the ssh server is running
   ps -ef | grep sshd
3. Verify if the machine is listening on the proper ssh port
   netstat -tulpn (use sudo)
4. Use putty (or something else) to make an SSH connection to the server



5. Enter username and password
6. Change the password of the user you have created.
   passwd
7. Update package cache
   apt update
8. List packages that need to be updated
   apt list --upgradable
9. Update all packages on the server
   apt upgrade

Deploy SRVLAB02

1. Open your vCenter/VMware workstation and import the ovf template (select all files in the folder)
2. Follow the wizard
3. The vm (SRVLAB02) is configured to use DHCP. Make sure SRVLAB01 and SRVLAB02 are in the same vlan.
4. Start the machine

In case of HyperV

1. Download the ISO for CentOS server (minimal)
2. Create a new vm and attach the ISO (HyperV: vm need to be generation 1)
3. Start de vm and follow the wizard
    1. auto start network
    2. storage keep default
4. Set a root password and create an extra user
5. Wait for the installation to finish and reboot
6. Logon
7. Change the sudo settings with the command 'visudo'. Search for the sudo-settings for the wheel group. Un-comment the line with 'NOPASSWD' and comment the line without 'NOPASSWD'. Save and quit
8. Add the extra user you created to the wheel group
    usermod -aG wheel <username>

Connect to SRVLAB02

1. Check the vCenter for the IP of SRVLAB02 or login and check the I manually.

> Alternative:
>
> From SRVLAB01 run
> `ip=10.10.6; for i in {1..254} ;do (ping $ip.$i -c 1 -w 5  >/dev/null && echo "$ip.$i" &) ;done`
>
> If you recognize an unknown IP-address, that is probably SRVLAB02. You can verify the mac address via: `ip neigh`

2. Connect from SRVLAB01 to SRVLAB02 via ssh.
    User: axxes
    Password: linuxistop

Connect to SRVLAB02

1. You are connected to the vm via ssh
2. Navigate to /etc/sysconfig/network-scripts/ and list the folder
3. Look for the file with format 'ifcfg-<interface>' and open it. ('lo' is local loopback, that not to right one)
4. The interface is configured with a dynamic ip. Change it to a static IP.
    BOOTPROTO=static
    IPADDR=192.168.1.130
    NAME=<name interface>
    NETMASK="255.255.255.0"
    GATEWAY="192.168.1.1"
    DNS1=8.8.8.8
    DNS2=1.1.1.1
    ONBOOT=yes

5. Now we need to restart our interface to activate the new configuration. With 'ifdown <interface>' we bring down the interface. With 'ifup <interface>' we bring the interface up. But be careful. Bringing down the interface will disconnect the ssh. Therefore we past both

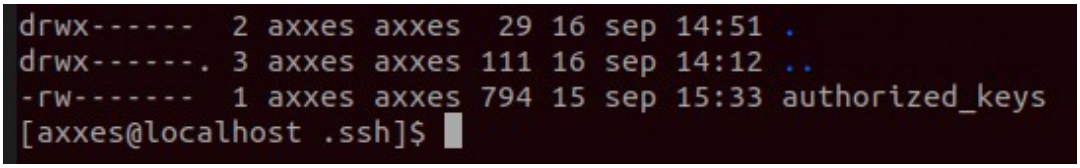
commands together with '&&'

```
sudo su
ifdown ens192 && ifup ens192
```

6. Because you change the ip-address, the ssh session will hang. Reconnect to the new ip address.
7. Check the new interface configuration. Make sure IP, network mask and broadcast address are correct (There are two ways to display the networksettings, try both). With 'ip route' check if the default route is correct.

Configure an ssh key-pair

1. Open an ssh session to both SRVLAB01 and SRVLAB02.
2. In SRVLAB01 create a ssh key-pair, (keep default path)
   ssh-keygen -t rsa
3. Navigate to ~/.ssh (/home/user/.ssh) and check the content of every file. Do you know the purpose of every file?
4. Switch to SRVLAB02 and navigate to the home directory of the axxes user
5. Create in the home directory a new folder '.ssh'
6. List the home directory, do you see the new folder? Why? (Hint: add option -a)
   List the home directory and check the folder permissions.
7. Set the permissions to 700 on the .ssh folder (user → read, write and execute | group → none | other → none)
   chmod 700 .ssh
   chmod u+rwx,g-rwx,o-rwx .ssh
8. Create a file 'authorized_keys' in the .ssh folder, set the permissions that only the user can read and write. Groups and others are not allowed to view, write or execute the file.



9. Copy the public key into the 'authorized_keys'-file. It should look like this:
   ssh-rsa AAAAB3c2EAAAADAQ……79BcZ/vtRePboLI6HuIrFPVwwlMsh user@srvlab01
10. Open /etc/ssh/sshd_config and add/uncomment the following lines
    RSAAuthentication yes
    PubkeyAuthentication yes
    AuthorizedKeysFile %h/.ssh/authorized_keys
11. Go back to SRVLAB01 and create a file ~/.ssh/config. Open the fie and add
    Host *
    IdentityFile ~/.ssh/id_rsa
12. Connect now from SRVLAB01 to SRVLAB02, Did you get a prompt for a password?

Install git and clone repo

1. Connect to SRVLAB02
2. Check if git is installed. If not, install it.
3. Clone the git repository
   git clone https://github.com/sanderdescamps/Axxes-linux-training.git

Install webserver

1. Connect to SRVLAB02
2. Add the nginx repository
   rpm -Uvh http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm
3. Install nginx
4. Navigate to /usr/share/nginx/html/
5. Open the index.html file with nano or vi
6. Modify the content. (If you need inspiration, in the git repo you find a very basic example)
7. Start the nginx service
   systemctl start nginx
8. Check the service status
   systemctl status nginx
9. Open a browser on your desktop and go to http://<ip_of_SRVLAB02>
10. Test the site also from SRVLAB01 (terminal) with the 'curl' command
11. Reboot the server, check if the web-page comes back online. Check also the status of the service.
12. Nginx is not configured to start at boot. To fix this, check the man-page of systemctl and find the right command.

Troubleshooting

1. Connect to SRVLAB02
2. Install bind-utils, net-tools, tcpdump, traceroute, nmap, telnet
   yum install bind-utils net-tools tcpdump traceroute nmap telnet
3. Ask Sander to come by and do some modifications to the (network) configuration of SRVLAB02. The goal is to find what is broken and fix it again. Bellow a set of tools that you could use to troubleshoot the issue.
   netstat, ip a, ip route, ip neigh, ss, ping, dig, nslookup, tcpdump, traceroute…

Additional exercises

The exercises bellow are extras. It doesn't matter in which order you do them. Also if there are things you like to try, feel free to experiment.

1. Extend the 5GB disk of SRVLAB02 to 6GB. Add the extra space to the root partition.
   1. Extend disk in Vmware
   2. Refresh the iSCSI disks in the os
   3. Extend the physical volume
   4. Extend the logical volume
   5. Extend the file-system
2. Give someone else access to your server
   1. create a new user (and configure an ssh keypair)
   2. Check the firewall's between the servers and make sure ssh (port 22) is allowed.
3. On SRVLAB02, Create a script that adds the date and time to the webpage. Put the script in the crontab and run it every minute.
4. On SRVLAB01, create a rsync that makes a backup of the source-file(s) of the website running on SRVLAB02. Put the backup in /opt/backup
5. For Java trainees: Java can run on Linux, you could test one of your applications and compare the difference in resources with Windows.

6. For the Windows trainees: If you had the Veeam course, Try to backup one of the vm's and do a file restore.
7. Advanced: Rescue from a broken boot sector:
    1. Take a snapshot first
    2. Remove all initramfs and vmlinuz files from the /boot folder.
    3. Reboot the machine
    4. You will see that the server is not able to boot anymore. With help of a live cd you can still recover the server. With the help of the internet, it is possible. Good luck.