

Cobbler Installation Part 1

Introduction:

The procedure in this document is a very condensed form of the tutorial found in:

<https://www.ibm.com/developerworks/library/l-cobbler/>

Please see the above document for detailed explanations of the various components of Cobbler. In particular, this document currently contains no screen captures to show the output from the commands. You can see much of the output by cross referencing with the above document. I recommend keeping both documents open in your web browser.

Please ask questions in class about anything you do not understand.

Please keep in mind that we will spend 2 or 3 class sessions covering this topic. It is an important topic and it is not a basic topic. We will move at the pace that ensures that everyone is learning both this topic as well as continuing to fill in the missing pieces about Linux.

This topic would not normally be covered in a beginning Linux class.

In this document, the commands you should enter during the installation are in green boxes. An example of a command box is:

```
yum install emacs  
cp file1 to file2
```

```
mkdir -p /tmp/a/b/c/  
rm filename
```

You can use the mouse to select, copy and paste the commands from this document to the command line or an editor open in command line mode. The cut and paste commands for a Linux terminal window are a little different than what you might be used to.

```
Copy - Control_Shift_C  
Paste - Control_Shift_V  
Cut - Control_Shift_X
```

Automate and manage installation with Cobbler

Setting up a network environment can involve many steps until everything is ready to start the installation. The process is not straightforward, and manual registration of each client machine that must be provisioned can be annoying. Cobbler addresses these shortcomings by creating a central point of management for all aspects of machine provisioning. It can reconfigure services, create repositories, extract operating system media, act or integrate with a configuration management system, control power management, and more.

Starting the installation:

You should be logged into the terminal as **student** not as root. In general it is not good practice to log into a GUI console as root. It is considered a security risk to browse the Internet as root. As student, you can issue commands as root by using the sudo command. Try:

```
sudo whoami
```

You can get a root shell by using the command:

```
sudo su -
```

You should do this now and use that window to issue all the commands for this installation.

IMPORTANT

You need to know the IP address and hostname for the machine you are working on.

It will be something like 172.16.9.190 and htc190.najah.edu

Please check that it is correct using the following commands:

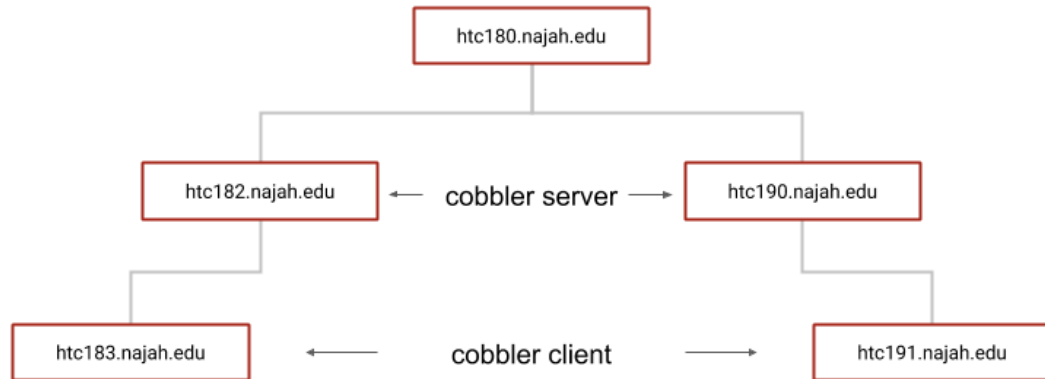
```
ip -4 address | grep inet | grep -v '127.0.0.' | awk '{print $2}' | sed s'/\././';hostname
```

You should see something like:

```
172.16.9.182  
htc182.najah.edu
```

The diagram below shows the basic layout of our test room. It is not a network diagram. Each group of two computers in the room make up a Cobbler Server and a Cobbler Client. You installed the basic operating system on the Cobbler Server using the USB stick. You will now be configuring the Cobbler Server so that you can install the operating system on the Cobbler Client over the network, completely unassisted.

Components to your mini cluster



This is a logical component diagram
This is not a network diagram

Part A:

Installing Cobbler and the needed support packages

In the first part of the class you installed a number of additional CentOS repo files. All the repo files on your current machine are configured to get files from the main server in the classroom. Cobbler is in that repository. So let's install some packages:

```
yum install cobbler fence-agents cobbler-web dnsmasq syslinux pykickstart xinetd  
yum install perl
```

Brief description of the packages:

- cobbler - Main cobbler software package
- fence-agents - Support packages for cobbler
- cobbler-web - GUI web interface for Cobbler (We will try to use it later in the class)
- dnsmasq - Software to handle DNS and DHCP for Cobbler
- syslinux - Initial low level PXE boot images.
- pykickstart - Python support package for kickstart files
- perl - Scripting language (I believe we have already installed this)

Part B:

Starting and Enabling Cobbler Daemons

Cobbler uses a few daemons or background process.

A daemon is a computer program that runs as a background process

CentOS uses a command called systemctl to manage daemons. Some of the things we do with daemons are start, stop, restart, enable, disable and check their status.

The 3 main Cobbler daemons are:

- cobblerd - The main cobbler program
- httpd - A web server
- xinetd - Manages network connections

Enable the daemons:(set it to start on reboot):

```
systemctl enable cobblerd  
systemctl enable httpd  
systemctl enable xinetd
```

Start the daemons:

```
systemctl start cobblerd  
systemctl start httpd  
systemctl start xinetd
```

Part C:

Disable SELINUX

Linux uses a security mechanism called selinux. Unfortunately, it interferes with way Cobbler works, so we are going to turn it off:

```
setenforce 0
```

Now edit the file /etc/selinux/config and change

SELINUX=enforcing
to
SELINUX=disabled

NOTE: Do not add spaces to the line.

```
vi /etc/selinux/config
```

Part D:

A first check of the installation

We are now going to check that the installation is proceeding correctly. Enter:

```
cobbler check
```

You will get some warnings, these will be fixed later. You should not get any errors.

Part E:

Opening network ports

We need to configure the firewall. The following commands open the “ports” to remote hosts. Perhaps you recognize port 80 as the httpd or web server port. These ports must be open so that hosts can contact your Cobbler Server in order boot and download the operating system.

```
firewall-cmd --add-port=22/tcp --permanent  
firewall-cmd --add-port=80/tcp --permanent  
firewall-cmd --add-port=443/tcp --permanent  
firewall-cmd --add-service=dhcp --permanent  
firewall-cmd --add-port=69/tcp --permanent  
firewall-cmd --add-port=69/udp --permanent  
firewall-cmd --add-port=4011/udp --permanent  
firewall-cmd --reload
```

Part F:

Configuring /etc/cobbler/settings

First create the encrypted root password (annajah2018) using the shell command:

```
openssl passwd -1
```

Copy the output of encrypted password and you will insert it in /etc/cobbler/settings for default_password_crypted. For example:

```
htc190> openssl passwd -1  
Password:  
Verifying - Password:  
$1$0cRXUryN$qTacWmoiV.1eeM8QaLUuY.
```

Be sure to copy (Control-Shift-C) all the highlighted text. Then paste this text (Control-Shift-V) into the editor as described below.

Edit /etc/cobbler/settings and make the following changes:

```
vi /etc/cobbler/settings
```

You must set these parameters.

Replace <server's IP address> with the IP address of your Cobbler server.

The same IP address is used for both next_server and server .

- default_password_crypted: "Output from openssl -1"
- manage_dhcp: 1
- manage_dns: 1
- manage_tftpd: 1
- restart_dhcp: 1
- restart_dns: 1
- pxe_just_once: 1
- next_server: <server's IP address>
- server: <server's IP address>

Part G:

Configuring /etc/cobbler/modules.conf

Edit /etc/cobbler/modules.conf and make the following changes:

```
vi /etc/cobbler/modules.conf
```

```
[dns]
module = manage_dnsmasq

[dhcp]
module = manage_dnsmasq

[tftpd]
module = manage_in_tftpd
```

Part H:

Creating /etc/cobbler/dnsmasq.template

We now going to create the template file dnsmasq. This file is read by cobbler to create the /etc/dnsmasq.conf. We start by renaming original /etc/cobbler/dnsmasq.template file.:

```
mv /etc/cobbler/dnsmasq.template /etc/cobbler/dnsmasq.template.save
```

Now we want to edit (create) /etc/cobbler/dnsmasq.template using the values below.

NOTE: When you open `/etc/cobbler/dnsmasq.template` it is now an empty file. You are creating the file using the input below.

You should cut and paste these lines exactly as they are into /etc/cobbler/dnsmasq.template:

```
vi /etc/cobbler/dnsmasq.template
```

Cut and paste these lines to the file:

```
#-----  
# Cobbler generated configuration file for dnsmasq  
#  
  
# resolve.conf .. ?  
#no-poll  
#enable-dbus  
read-ethers  
addn-hosts = /var/lib/cobbler/cobbler_hosts  
  
dhcp-range=172.16.9.194,172.16.9.200,255.255.240.0,1h  
dhcp-ignore=tag:!known  
dhcp-option=66,$next_server  
dhcp-lease-max=1000  
dhcp-authoritative  
dhcp-boot=pxelinux.0  
dhcp-boot=net:normalarch,pxelinux.0  
dhcp-boot=net:ia64,$elilo  
  
$insert_cobbler_system_definitions  
#-----
```

Part I:

Restarting daemons and syncing the software

Cobbler is almost ready for use. Restart the services and synchronize the changes to the filesystem for them to take effect. Remember also to restart the xinetd service to make TFTP available.

Run the following commands to restart the services and sync cobbler. This should complete without any errors. There will be output but no errors.

```
systemctl restart cobblerd  
sleep 10  
cobbler sync  
systemctl restart xinetd
```

Part J:

Importing a Linux Distribution into Cobbler

To add the CentOS 7 installation tree, first download the ISO media and then run the following commands to mount the media and extract its content. This imports CentOS release into Cobbler which it will a little later to perform the installation. The last command will print information about the CentOS 7 installation in Cobbler. This command should complete without any errors.

```
mkdir -p /var/lib/cobbler/isos  
cd /var/lib/cobbler/isos  
curl -o CentOS7-x86_64-Everything.iso  
http://htc180.najah.edu/software/isos/CentOS-7-x86_64-Everything-1708.iso  
mkdir -p /mnt/iso  
mount -o loop /var/lib/cobbler/isos/CentOS7-x86_64-Everything.iso /mnt/iso  
cobbler import --arch=x86_64 --path=/mnt/iso --name=CentOS7  
  
cobbler distro report
```