

Lab 7: Examining and installing Pi-hole

Introduction

The software you want to use may not always be available from a repository for your Linux Distro. You will have to install software (for business or personal) from other sources at some point. Additionally, not all software is open-source or free. Many developers and enterprises will not take the time to maintain packages in any repository, instead they will distribute or sell their software via various other methods.

Installing software from internet sources outside of an open repository can be risky. Firstly, you must trust the developer or proprietor of the software. Secondly you must make sure the software you download is genuine and has not been tampered with. With proprietary software, you must also study the EULAs (End User Licence Agreements). EULAs are inevitably imposed on you and your systems' users upon encountering software or services. Additionally, companies change the EULA and Privacy policies from time to time and often the new terms are imposed on existing and new users of the software or service.

At this point in this course you do have enough knowledge to install any software from any sources on your Linux system. If you are installing a software from outside of repositories for your specific distro, you are responsible for updating and maintaining the software. It is very important to know how to update or remove the specific application since your package management software is not going to be able to do that for you.

One way to find out what the software does to your system during installation is to examine the installation scripts. By examining the installation script, you can not only make sure that the installation will work as intended, but also figure out what changes it makes to your system in case you want to remove or uninstall the software manually. Again, your package management software may not be involved in this at all, so you are tasked with taking care of this yourself.

As a practice we are going to be installing pi-hole on a Linux VM, while examining the installation process and some aspects of this application itself. The reason we are going through the trouble of installing software from outside of repositories is because (1) many commercial products work like this, and (2) many non-commercial software is distributed this way as well!

Create a test VM (or a clone of your existing VM)

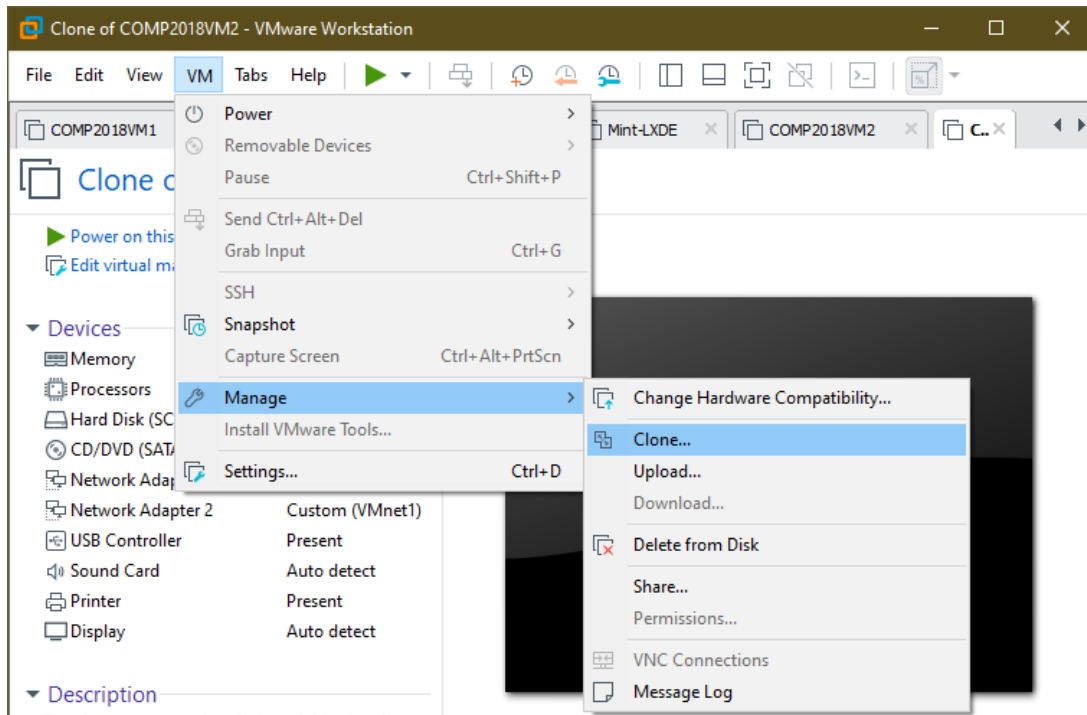
Since we are using VM for our main OS installation we have a big advantage. We can afford to do a lot of trial and error without much costs by cloning, backing up, and restoring VMs as much as needed. So, go ahead and clone your VM and work with the clone. In case anything goes wrong, you can just delete the clone and start again.

If you are creating a new VM from scratch for this or any future projects, it is very handy to make sure that VM has two interfaces: a connection to Internet (via NAT) and one to local network with static IP (same as our current class lab VM).

Note: I do not recommend changing the default VMware NAT LAN settings as this will affect every other VM on your machine. Keep the default NAT VMnet for accessing Internet in your VMs. If you

are experimenting with the Virtual LAN settings, create new Virtual LAN, since other VMs are not attached to the LAN and are not affected by changes as you experiment.

To create a clone VMware Workstation in Windows, you can click on VM > Manage > “Clone...” via the menus at the top:



If you are using VMware Fusion in MacOS you can copy all the VM files to another folder and launch the VM from the new location since it seems VMware Fusion may not have an easy to access clone function (Please research this since I do not have access to a MacOS machine to check).

If you are using VirtualBox, select the VM and click on the “Snapshots” button at the top, then select “Full Clone” radio button in the wizard.

Obtaining and installing Pi-hole

The software and all information are available from www.pi-hole.net. Here is a summary of what packages it provides and uses in order to function (<https://docs.pi-hole.net/main/origins/>):

1. DNS server and DHCP server function using a package called dnsmasq
2. Curl for retrieving data from URLs'
3. Webserver server by using lighttpd package
4. Php scripting language for the UI and some functions
5. AdminLTE Dashboard, web UI administrative dashboard that uses JS/Jquery
6. SQL DB server using sqlite3

As you may have noticed, Pi-hole uses a collection of other software to function. A lot of Linux/Unix software are designed this way (in fact it is encouraged to use existing utilities and tools to create new applications). You can, in fact, configure and install every package individually in order to achieve the

exact same results (or even better, customizing it to your exact needs). However, someone else has already done a lot of the work for us, so we can simply use this instead.

Examine the “basic-install.sh” script

In the main documentations (<https://docs.pi-hole.net/main/basic-install/>) there are 3 methods of installation, all of which involves running the “basic-install.sh” bash script. Let’s examine this script file first.

1. Create a folder called pi-hole in your home directory
 - a. `mkdir pi-hole && cd pi-hole`
2. obtain the “basic-install.sh” script so we can examine it:
 - a. `wget -O basic-install.sh https://install.pi-hole.net`
3. Now let’s look inside the script file to see what it does:
 - a. `nano basic-install.sh`
 - b. Alternatively, you can copy the script to your host computer and use your favorite text editor to examine it! It may help the readability of the script.

>>>>Answer the following questions:<<<<<

Q1: By examining the script what is the very first command that the script issues in bash?

Q2: There are “Undocumented Flags” for running the script from bash (flags are options when running a command from bash, for example ls command has --help, -l, and -a as flags that can be set by you when running the utility from bash). What are the 4 undocumented flags that you can invoke in bash when using this script?

Q3: since you found these “Undocumented flags”, how do you run the script without installing lighttpd webserver if you have another webserver already installed? For answer complete this command: `sudo bash <...>`

In case you had not noticed yet, the documentation at <https://docs.pi-hole.net/main/basic-install/> does not specify what distribution the installation is intended for (it does not specify different installer or different commands for Red-Hat, Debian, etc... only one install script for all).

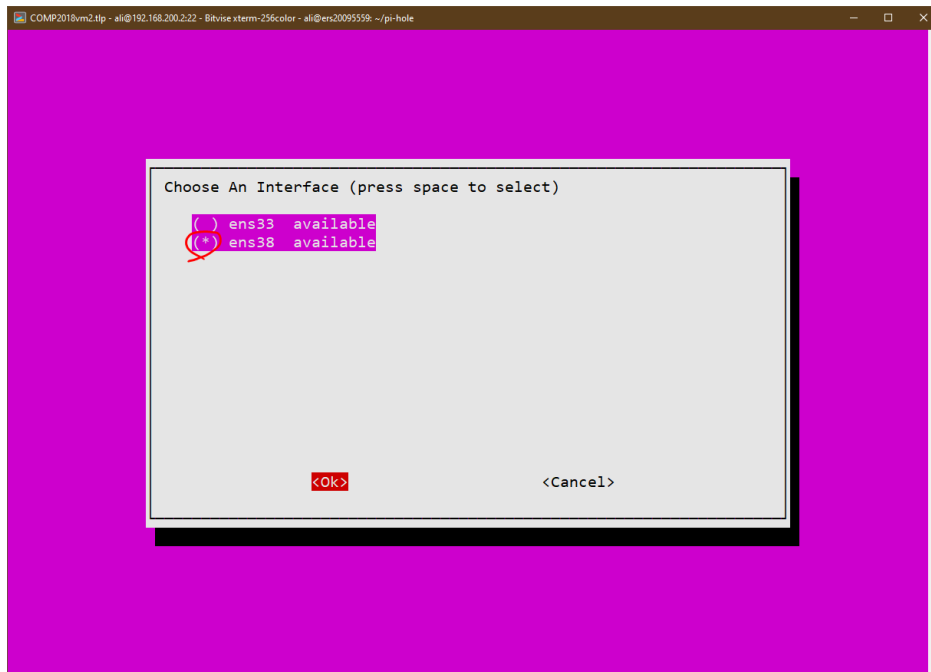
Q4: how does the script detect if it is running in a Debian distribution? Hint: find function called “distro_check()”.

Q5: Find the function “install_manpage()”. What pages and in what chapters, are added to the man pages in your OS? Hint: there are 3 new entries added to 2 different chapters of the man pages.

Execute the installation script

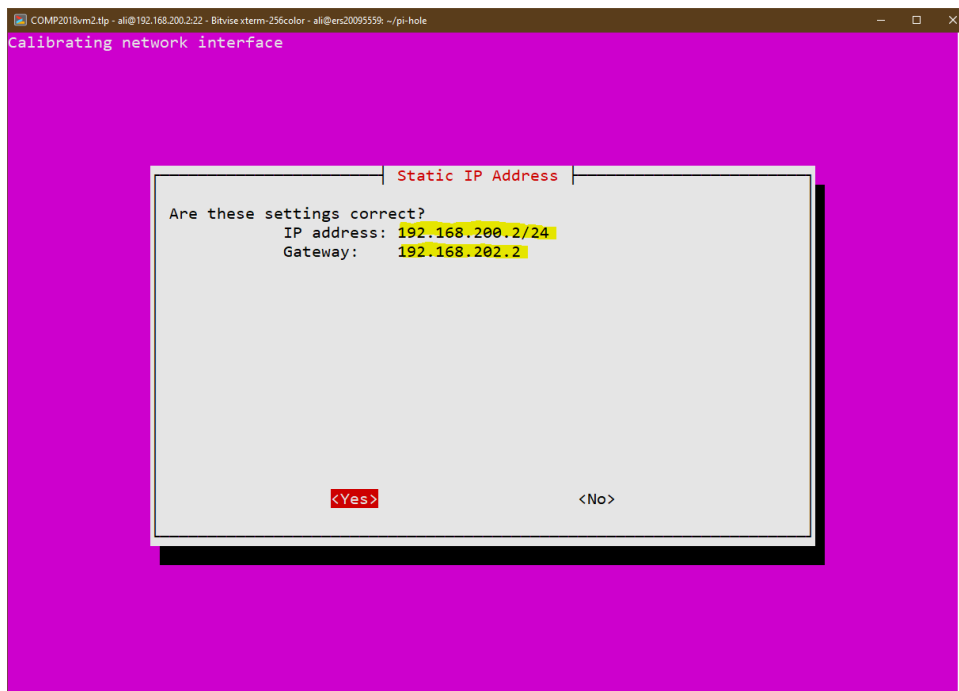
Make sure that the basic-install.sh has execute permissions. Go ahead and use `sudo bash basic-install.sh` to run the installation script.

During the install it will ask you to select the network interface for running the server on. Make sure to choose the interface that is “host-only” with static IP and not your NAT interface!



Select the desired settings for your DNS provider and lock lists.

NOTE: make sure that your static IP address is detected properly. Read the IP address and enter the correct IP address if the script is not detecting the correct IP address (i.e. the static IP that you have set for your second network interface). This is crucial that you are reading and applying the correct settings.



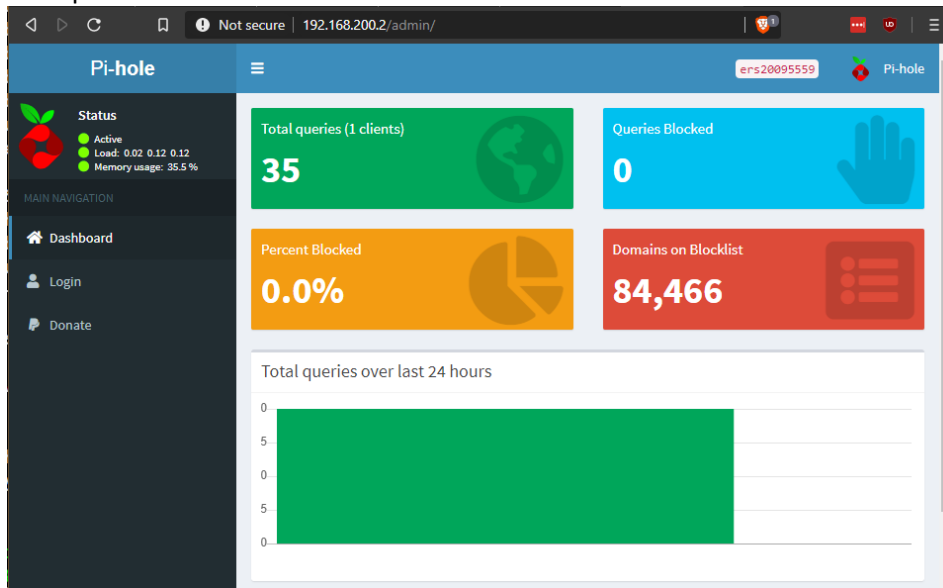
On the last screen, the installation should display your randomly generated “admin” password for the WebUI. Make sure to copy this or write it down. You can change this via terminal if you forget or miss this step (lookup the man pages for pihole to find out how, also see Q6).

Once the setup is done you can open a web browser in your host and navigate to the IP address of your VM.

>>>>Screenshot1.png<<<<

Take a screenshot of the web interface displaying your hostname at the top to indicate that you have a working WebUI.

Example:



If you can see the above web page, congratulations! You have a working Pi-hole. You can log in and play around, as well as change your host OS DNS server manually to see its effects.

Examine pihole command

>>>>Answer the following questions:<<<<

Q6: Read the man pages for pihole. What command can you type if you want to set or change the webUI password?

Q7: according to man pages for pihole, what command uninstalls pihole from your system?

Assignment Submission

1. Answer Questions Q1 to Q7 in the Text Submission box in Blackboard
2. Attach the Screenshot1.png to the assignment