Setting up the PiPinger!

--Craig A. Knight
Computer & Network Support Instructor
Davis Hart Career Center

A few months ago, I was having issues with a network that I manage for a local company. In their network, they have multiple sites that are connected using a Virtual Private Network (VPN) connection. The problem was that one of the VPN tunnels appeared to be continually dropping connection for some reason. In order to troubleshoot this problem, I decided that it would be handy to have a dedicated, low-power device that I could just plug into the network and monitor whether this VPN connection was working or not. Enter the Raspberry Pi! Since I have been using the Raspberry Pi computer in an Intro to Technology class that I teach, I thought that this might be a cool way to solve my VPN monitoring issue.

This how-to will outline the procedure that I used to setup a simple Raspberry Pi Model B/B+ computer as a network pinging device that a system administrator could easily connect on a network to keep track of devices being "up" (responding to PINGs) on their network. If a device is not responding for some reason, then the PiPinger will send a text message to a person to alert them to that fact! The "PiPinger" (as I like to call it) actually helped me to track down the problem that I was having. I hope that it might help you with your network management as well.

So let's get started!

Procedure

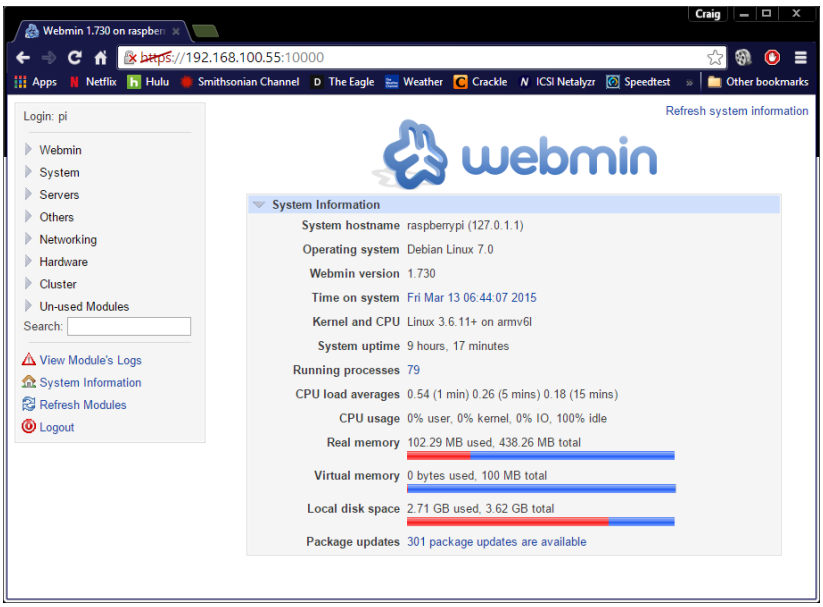
1. Setup a Raspberry Pi Model B/B+ with the Raspbian OS. This can be done quite easily by preparing a "NOOBS" SD card. The procedure for doing this can be found at: http://www.raspberrypi.org/help/noobs-setup/

2. After installing the Raspbian OS, you will need to open a terminal window and fully update your system by typing the following command:

   sudo apt-get update

3. After fully updating your RPi, you will want to install a web administration interface called "Webmin" on your computer. Webmin will allow you to remotely connect to your RPi using a web browser and perform several different things on the system. The procedure for installing Webmin on your system can be found at: http://blog.technotesdesk.com/install-webmin-on-raspberry-pi-via-apt-get/

4. By default, Webmin will wait for a connection on the network using TCP port number 10000. You can test to see if Webmin is working on your computer by simply opening a web browser and typing in the IP address 127.0.0.1:10000. You should see a webpage like the one shown below.

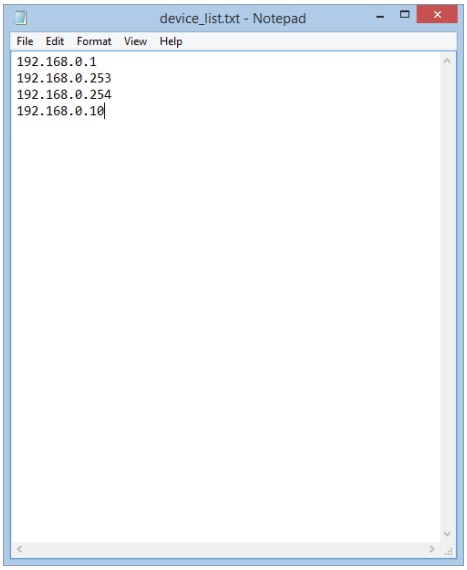


5. You will next need to statically assign IP addressing information to the RPi. This is necessary so that we can remotely connect to the RPi and perform administrative tasks. If you are not familiar with how to do this, you can refer to the tutorial that can be found at: http://www.modmypi.com/blog/tutorial-how-to-give-your-raspberry-pi-a-static-ip-address

6. You will next want to download a copy of the pipinger.py program. This program is available as a "zipped" file at the hartcareercenter.org website, in the Computer Maintenance / Networking career path section. After downloading the file, you will need to unzip it into an appropriate location on your RPi.

7. Using a text editor (like "leafpad"), you will need to edit the pipinger.py program and enter information that will be appropriate for your setup. The primary thing that will need to be entered is the email information where you will want to have alert messages sent. I have already setup the program to use the gmail service, so if you are using a gmail account (which I recommend), you will pretty much just need to enter your username and password information.

8. The next step of the process will be to edit the device_list.txt file that is used by the pipinger.py program. This simple text file will contain the IP addresses of the devices on your network that you would like to send PING requests to. The format of this file will look like the following:

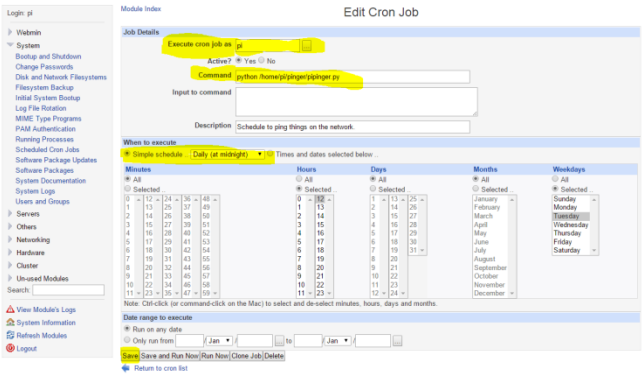


9. We can now test out the pipinger.py program and verify that it works. Using the IDLE development program, open and run the pipinger.py program. If there are no errors, then a new file will be created in the folder where the pippinger.py and device_list.txt files reside that is called pinger.log. This is a simple text file that will have a log showing information about each of the devices that were PINGed in your device list. You can open it up in any simple text editor to view its contents.

10. If the pipinger.py program is working correctly, we can now set it up to run automatically at a time that you schedule. This can be done using the CRON utility that is available within the Raspbian OS. Even though we could set this up via command line, we will configure this using the Webmin utility that was installed earlier (mainly because it is easier…and I like GUIs!).

    If you are working directly on the RPi, you can open up the Webmin management interface by opening a web browser and typing in https://127.0.0.1:10000 in the address bar. You can use your primary sudo account to login to the interface. If you are connecting to your RPi from another computer simply type https:// xxx.xxx.xxx.xxx:10000 (where the x's are your address). When the Webmin administration page loads you will select the "system" option in the left-hand pane, and then select the "Scheduled Cron Jobs" option that appears under "System". You should get a window that looks like what is shown below.

You can now select the "Create a new scheduled cron job" option along the bottom of the page which will bring up a window like the one shown below. You will then need to fill in or configure the items that are highlighted in the picture below.



Here is what you will need to type in:

- For the “Execute cron job as” box type in your main user (pi by default)
- For the “Command” box type in: “python” followed by the path to wherever you saved the pipinger.py program.
- In the “When to execute” section, select the days and times for when you want the pipinger.py program to be run.
- You can then select “Save” at the bottom of the screen and you should be all setup to have pipinger.py running on a schedule.

NOTE: By the way, Webmin can be used for managing a whole bunch of stuff on your RPi from other computers. You might want to play around with what you can do with it!

CONGRATULATIONS! If everything is setup correctly, your PIPINGER will now ping different computers on your network and let you know if one of them does not respond, by sending you an email. How cool is that!!!! Of course, I am really not a very good programmer, so feel free to tweak the pipinger.py code and modify or add any features that you like!

Happy Hacking, everyone!!!!!