Friday, Mar. 10
Winter 2017

CME 451 – Lab 5

Router Operation and Network Analysis [1]

Pre-Lab: While the material provided in this lab is intended to be self-contained, the interested reader is urged to also browse through the online tutorial of Python programming by Python Software Foundation, available here: `https://www.python.org/doc/`.

Verification: Exercises marked with an **asterisk** (∗) are required to be verified and signed off by a lab instructor during your assigned lab time. When you complete a step that requires verification, simply demonstrate the step to a lab instructor. Turn in the completed verification sheet to your TA at the end of the lab.

Code Submission: Exercises marked with a **dagger** (†) can be completed after your assigned lab time. Submit your code and relevant outputs electronically in one zip file (named <NSID>-<Lab#>) through the BlackBoard no later than a week after your lab. Enhance the readability of your code by adding explanatory comments wherever appropriate. If you are unsure about what is expected, ask your TA.

1 Introduction

In this lab, you will learn the basic Cisco IOx router operation. You will learn to configure the router and then use the router to access data from the Internet. In addition, you will learn the basics of machine learning algorithms and use machine learning method to perform network analysis.

2 Router Operation

2.1 Introduction

The router we will learn to use is the Cisco IR829 Integrated Services Router as shown in Figure 1. IR829 Industrial Integrated Services Routers provide highly secure, reliable, and easy-to-manage 3G/4G LTE WAN cellular and Wireless LAN connectivity for mobile environments. The router supports Cisco IOS Software, is very compact, and designed for harsh environments. This router can be used in a variety of Internet-of-Things (IoT) tasks, such as management of fleet vehicles and mass transit applications.

Figure 1: Cisco IR829 Router

The front panel of single modem IR829 is shown in Figure 2. The functionality of these eight sets of slots are:

[1] Lab handouts are available online on the Blackboard. Please report typos and send comments to francis.bui@usask.ca.

1. Cellular 0 AUX
2. Modularity slot
3. Gigabit WAN (SFP)
4. Gigabit Ethernet LAN/PoE (RJ45)
5. Serial Ports
6. USE 2.0 type-A port
7. Power Input
8. WLAN ANT 0 (2.4GHz)

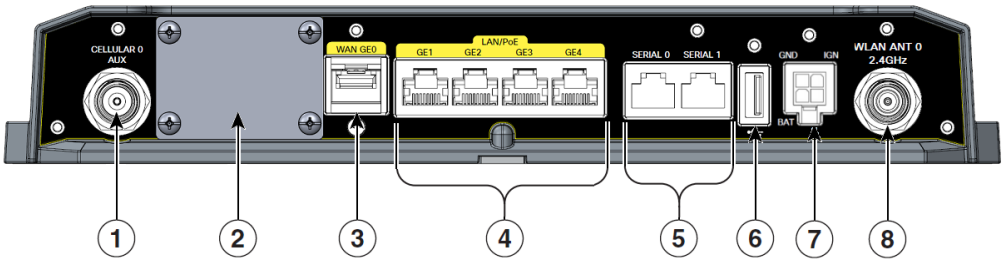

Figure 2: Cisco IR829 front panel

The back panel of IR829 is shown in Figure 3. The seven components are:

1. WLAN ANT 0 (5GHz)
2. WLAN ANT 1 (2.4GHz)
3. Cover over SIM cards, reset button, and console port, as shown in the right of Figure 3.
   (1) Reset Button
   (2) Console Port
   (3) Sim Card Slot
4. GPS SMA
5. Denote SIM card order
6. WLAN ANT 1 (5GHz)
7. Cellular 0 MAIN

The IR829 router comes with multiple cables and antennas for you to use. You can easily install them into the corresponding slots.

In order to connect a PC to an Ethernet switch port, you can use an Ethernet cable to connect the RJ-45 port on PC and one of GE1-4 LAN ports on the router.

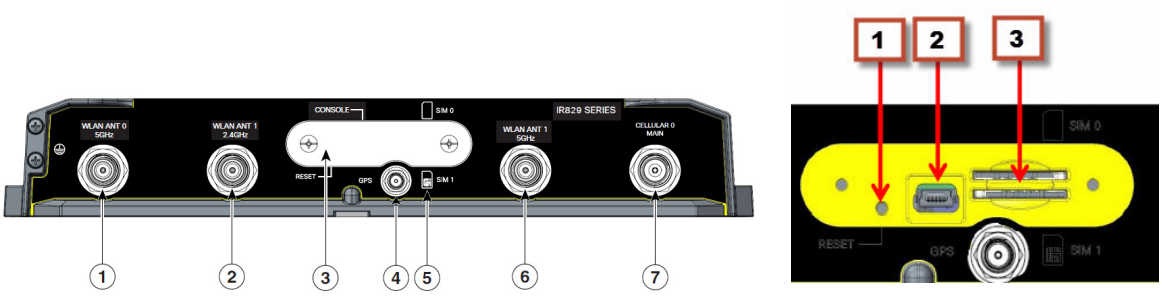

Figure 3: Cisco IR829 back panel

2.2 Basic Configuration

Before we can use the router, we always need to configure its interfaces. To do so, we need to connect a PC to the console port of the router. The router comes with a mini-USB to USB cable for you to perform configuration. First use a screwdriver to remove the cover as shown in Figure 3. Then connect the mini-USB side of the cable to the console port of the router and connect the opposite end of the cable to the USB port on your PC. Wait for your PC to discover the new device and install the corresponding driver. If your PC cannot install the driver automatically, you can manually download the driver at `https://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx` and install it.

After the installation complete, right click `Computer` on desktop and choose `Manage`. Click `Device Manager` on the left side, and then extend the `Ports(COM&LPT)` in the right side. You will see some COM ports as shown in Figure 4. Write down the COM port number of the standard COM Port in the red rectangular as we will use this port to login to the router.

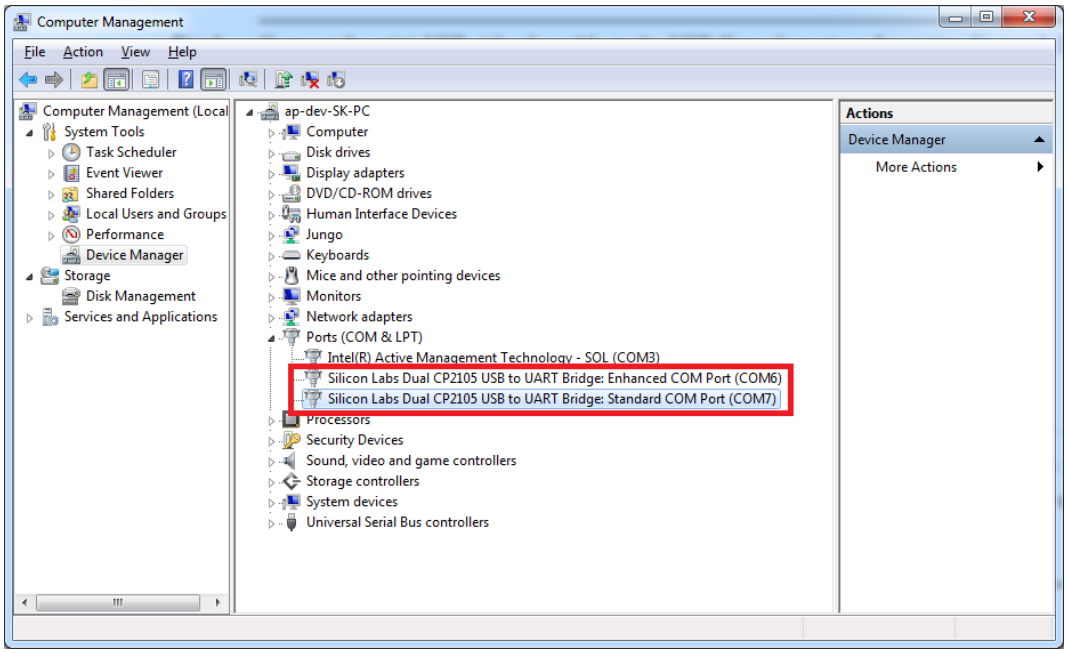

Figure 4: COM port for router

After verifying the cables are properly connected, you can connect the power cable to the router. The router will take some time to boot up and initialize.

On PC side, we can log into the router through console cable via `Putty`. After connecting the console cable and power cable, open `Putty` and configure the connection as shown in Figure 5. First select the connection type as `Serial`. The type the COM port number in serial line, in this case `COM7`. Leave the speed as 9600 and other settings to default. Then click `Open` to start the connection. A terminal will prompt out and the router booting

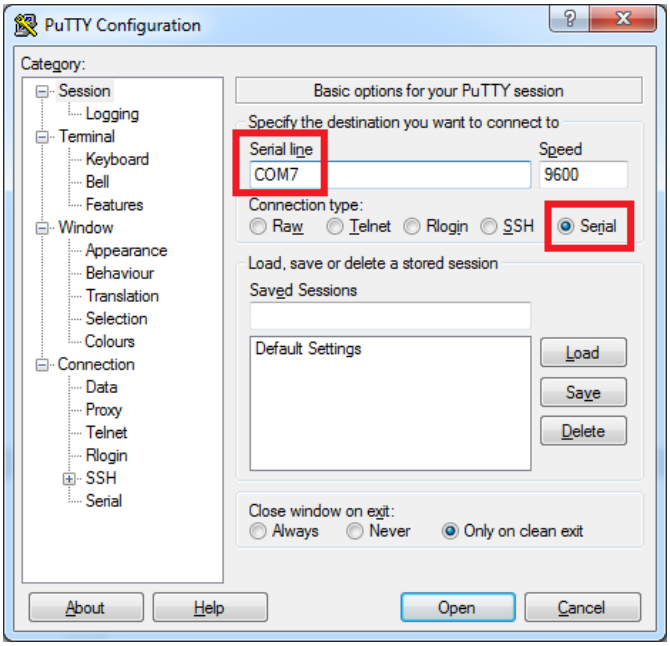

Figure 5: Putty connection configuration

information will be shown in the terminal as shown in Figure 6. If there is nothing shown in the terminal after a long time, press the `Enter` key using your keyboard. If all the configuration are correct, you should see information in the terminal. When `Press RETURN to get started!` appears in the terminal, the booting is complete and you can push the `Enter` key to start configuration.

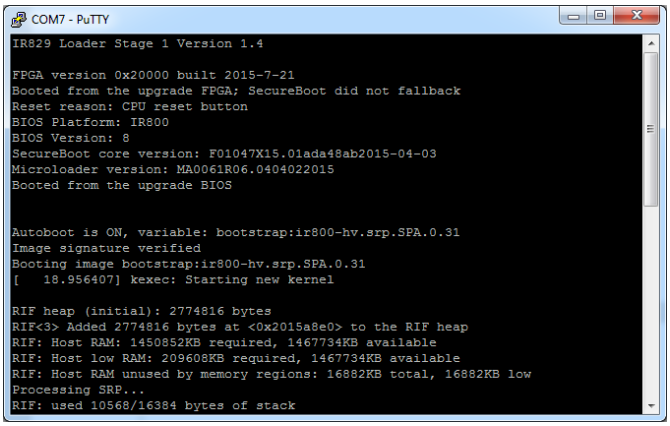

Figure 6: Terminal and boot information

Now, the terminal should show the router name, for example

```
IR800>
```

In order to configure the router, you need to enter the privileged EXEC mode. Type `en` or `enable` in the terminal to enter the privilege EXEC mode. As for now, this is the first time you configure the router and you do not import any running-config yet, you do not need a password to enter the privilege EXEC mode. (You can set the enable password in your running-config file. After you import the running-config file, each time you enter the privilege EXEC mode, you have to enter the password you set.) In privileged EXEC mode, the `>` will become `#`:

```
IR800#
```

In privileged EXEC mode, type `setup` to launch the setup command facility to start the basic setup. The setup command facility guides you through the configuration process by prompting you for the specific information that is needed to configure the system.

```
IR800# setup
--- System Configuration Dialog ---
Would you like to enter the initial configuration dialog? [yes/no]:
```

Type `yes` to start the initial configuration.

```
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.

Default settings are in square brackets '[]'.

Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system

Would you like to enter basic management setup? [yes/no]:
```

Type `yes` to continue.

```
Configuring global parameters:
Enter host name [Router]: CME451IOT
```

Type the host name. We use `CME451IOT` here.

```
The enable secret is a password used to protect access to
privileged EXEC and configuration modes. This password, after
entered, becomes encrypted in the configuration.
Enter enable secret: cme451iot
```

Type the secret password used later to enter privileged EXEC mode.

```
The enable password is used when you do not specify an
enable secret password, with some older software versions, and
some boot images.
Enter enable password: cme451ioten
```

Type the enable password. This will be used to enter privileged EXEC mode if you do not specify the secret password.

```
The virtual terminal password is used to protect
access to the router over a network interface.
Enter virtual terminal password: cme451iotvt
```

Type the virtual terminal password.

```
Configure SNMP Network Management? [no]:
```

Choose whether to configure SNMP network management. For this exercise, we choose no here.

Then a summary of available interfaces is displayed.

```
Enter interface name used to connect to the
management network from the above interface summary: GigabitEthernet0
```

Choose one of the available interfaces for connecting the router to the management network, for example GigabitEthernet0.

```
Configuring interface GigabitEthernet0:
Configure IP on this interface? [yes]: yes
    IP address for this interface: 172.1.2.3
    Subnet mask for this interface [255.255.0.0] : 255.255.0.0
```

Configure the IP address and the subnet mask for this interface.

After that, the configuration command script is created which summarize the commands you have entered. You can check each line to verify your configuration. Then the following will prompt:

```
[0] Go to the IOS command prompt without saving this config.
[1] Return back to the setup without saving this config.
[2] Save this configuration to nvram and exit.
```

Choose [2] to save your configuration. Then the system will take some time to build the configuration. After building the configuration, the new host name will appear:

```
CME451IOT>
```

In privileged EXEC mode, you can use `show configuration` to verify the initial configuration. This command will bring the script to show again.

If you make a mistake while using the setup command facility, you can exit (by press `Ctrl-C`) and run the setup command facility again.

In addition to the setup command facility, you can also configure each interface separately. For example, to configure the LAN interface (GE1-4) to access the network, type the following commands to configure an IP address to GigabitEthernet 5 interface:

```
CME451IOT# configure terminal
CME451IOT (config)# interface gigabitEthernet 5
CME451IOT (config-if)# ip address 10.10.10.10 255.255.255.0
CME451IOT (config-if)# no shutdown
CME451IOT (config-if)# exit
CME451IOT (config)# exit
```

In privileged EXEC mode, you can use `configure terminal` to enter configuration mode. Then you need to specify the interface which you want to configure, in this case it is `gigabitEthernet 5`. Then you can assign an IP address and subnet mask to the chosen interface. Finally, enter `no shutdown` to enable the interface. The following two `exit` are used to return back to the privileged EXEC mode.

Then type the following commands to configure an IP address to VLAN 1 interface. The procedure is the same as those configuring GigabitEthernet 5.

```
CME451IOT# configure terminal
CME451IOT (config)# interface vlan1
CME451IOT (config-if)# ip address 192.168.0.1 255.255.255.0
CME451IOT (config-if)# no shutdown
CME451IOT (config-if)# exit
CME451IOT (config)# exit
```

Exercise 2.1 (*Router Configuration*) ∗
Follow the content of Section 2 to connect the console of the router to the PC and perform the basic configuration.

3 Machine Learning for Network Analysis

Machine learning is to give computers the ability to learn without being explicitly programmed. It explores the study and construction of algorithms that can learn from and make predictions on data. Machine learning has achieved great success in computer vision, optical character recognition (OCR), and large data sets analysis.

The `scikit-learn` module provides you abundant tools to perform machine learning and data analysis in Python. With `scikit-learn` you can implement machine learning algorithm simply and efficiently. You only need to set the key parameters without implementing each detail computation of the algorithm.

If you have already installed `Anaconda`, you can install `scikit-learn` by

```
conda install scikit-learn
```

When using machine learning algorithms, you always have to do the following tasks: collecting enough amount of training data, choosing the learning algorithm, performing the training, and finally testing the trained model.

`scikit-learn` comes with few standard datasets you can use:

```
>>> from sklearn import datasets
>>> digits = datasets.load_digits()
```

These dataset are stored in 2D format with dimension (num_of_sample, num_of_feature). You can print the data in digits using `print(digits.data)` and you can check the ground truth of each data sample by `digits.target`.

`scikit-learn` provides many functions to implement machine learning algorithms, for example, the support vector machine (SVM).

```
>>> from sklearn import svm
>>> clf = svm.SVC(gamma=0.001, C=100.)
```

You can train the `SVM` classifier using the digits dataset by

```
>>> clf.fit(digits.data[:-1], digits.target[:-1])
```

Here you should specify the training data and the ground truth to allow the classifier to adjust its internal parameters. In this example, we use all the data except the last one to train the SVM classifier.

After training, we can test our trained model using the last data in the dataset (which is not used for training) by

```
>>> clf.predict(digits.data[-1:])
```

Exercise 3.1 (*Linear Regression*) †
The previous example show a classification task which works on discrete data. For continuous data samples, we should perform regression operation to analyze them. Regression can find the relationship among the independent training variables and generate a curve which represents the trend of the training set. The curve can then be used for prediction.

Linear regression, as a simple and popular regression method, is able to return you the linear relationship among continuous data samples. In this exercise, you will need to use `scikit-learn` to perform linear regression on a set of randomly generated data.

Specifically, you need to first generate 200 random data, perform linear regression, and then plot the generated data and generated curve.
Hint: You can refer to the following code to generate the random data points.

```
>>> import numpy as np
>>> from sklearn.utils import check_random_state
>>> n = 100
>>> x = np.arange(n)
>>> rs = check_random_state(0)
>>> y = rs.randint(-50, 50, size=(n,)) + 50. * np.log(1 + np.arange(n))
```

In order to plot the result, you might need `matplotlib` module.

CME 451 Lab 5 - Instructor Verification Sheet

Turn this page to your grading TA.

Name:

Student No.:

Instructor Verification: Exercise 2.1 Configuration Script	
Verified:	Date & Time: