**LAB 2: SCADA Control System Networking**

**Manual**

**Student Name**

**Student ID**

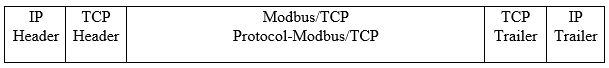
**Student Email**

**Purpose:** The purpose of this Lab exercise is to familiarize students with Modbus/TCP protocol and analyze network traffic in the SCADA control system environment.   
  
**Objective:** The student will utilize Radzio to observe Modbus traffic and Wireshark to analyze network traffic. Wireshark filters will then be applied to view specific traffic and packets. The 4Y students will use Python to open a network socket and send fake Modbus/TCP queries and responses to clients and servers.   
  
**Lab Setup and Requirements:** To begin this lab, you will need to have the virtual machine and the Water Tank Docker containers running. Radzio! Modbus Master Simulator and Wireshark are used to observe, capture and analyze the network traffic.

**Exercise #1 - Introduction to Modbus**

Modbus is a serial communications protocol published for use with its programmable logic controllers (PLCs). Modbus has become a de facto standard communication protocol, and it is now a commonly available means of connecting industrial electronic devices. Modbus is used in multiple master-slave applications to monitor and program devices, communicate between intelligent devices and sensors and instruments and monitor field devices using PCs and HMIs.   
  
Some notable facts about Modbus include:

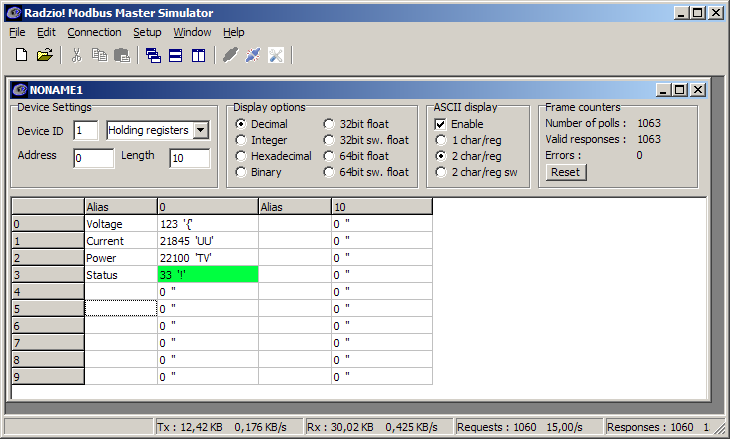
* Modbus has two implentations, IP/TCP and serial
* Modbus/TCP Relationship is that of master and slave
* Modbus transactions are a three-step process similar to the 3-way-hand shake
  1. Request
  2. Response from PDU
  3. Exception Response PDU
* Modbus functions include: Public, User-defined and Reserved
* Modbus Data types are Discrete input, Discrete output (coils), input registers, and holding registers



**Exercise #2 - Use Water Tank and verify traffic with Radzio! Modbus Master Simulator**

In this exercise, you will use observe the Modbus traffic from the PLC container.

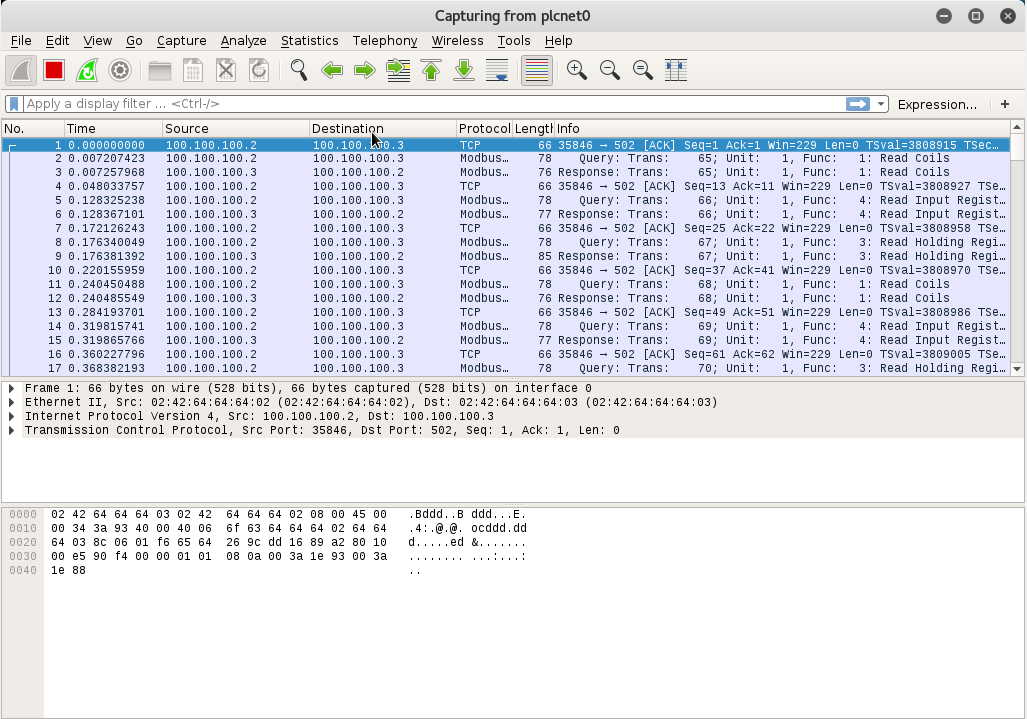
1. Login to the virtual machine using the credentials provided in Lab 1 and start the Water Tank containers (*./watertank.sh*).
2. Before using Wireshark to analyze the network traffic, we will verify the Modbus traffic using a helpful tool called Radzio! Modbus Master Simulator.
3. Open a terminal and navigate to the Radzio directory (/home/ccre/scadalab/lab2/Radzio). Run *wine RMMS.exe* to launch the tool.
4. Select File>New to open a new table.
5. Select Connection>Settings. Select Modbus TCP protocol. IP = PLC VM1 IP address (100.100.100.3). TCP Port = 502. Select Ok.
6. Select Connection>Connect. The Polls and Status should be increasing if the Modbus traffic is successfully being transmitted from the PLC container.



**Exercise #3 - Introduction to Wireshark & Traffic Captur**

Wireshark is a widely-used network protocol analyzer. With Wireshark, you can view your network at a microscopic level and use it to capture packets for analysis. It's also used by hackers to gather data in transit.   
  
Now that you have verified the Modbus traffic with Radzio you are going to use Wireshark to capture and analyze the network traffic.

1. On virtual machine, launch Wireshark. (Applications -> Internet -> Wireshark)
2. Go to the Capture>Options list and select the interface you want to begin capturing traffic on (plcnet0).
3. Select Capture to start capturing the network traffic.
4. Take note of the columns and information captured by Wireshark.
5. Stop capture after one minute.

  
For more information or to download Wireshark visit the Wireshark site [here](https://www.wireshark.org/).  
For additional instructions on how to capture packets in Wireshark, check out the following tutorial [here.](http://www.youtube.com/watch?v=6X5TwvGXHP0)

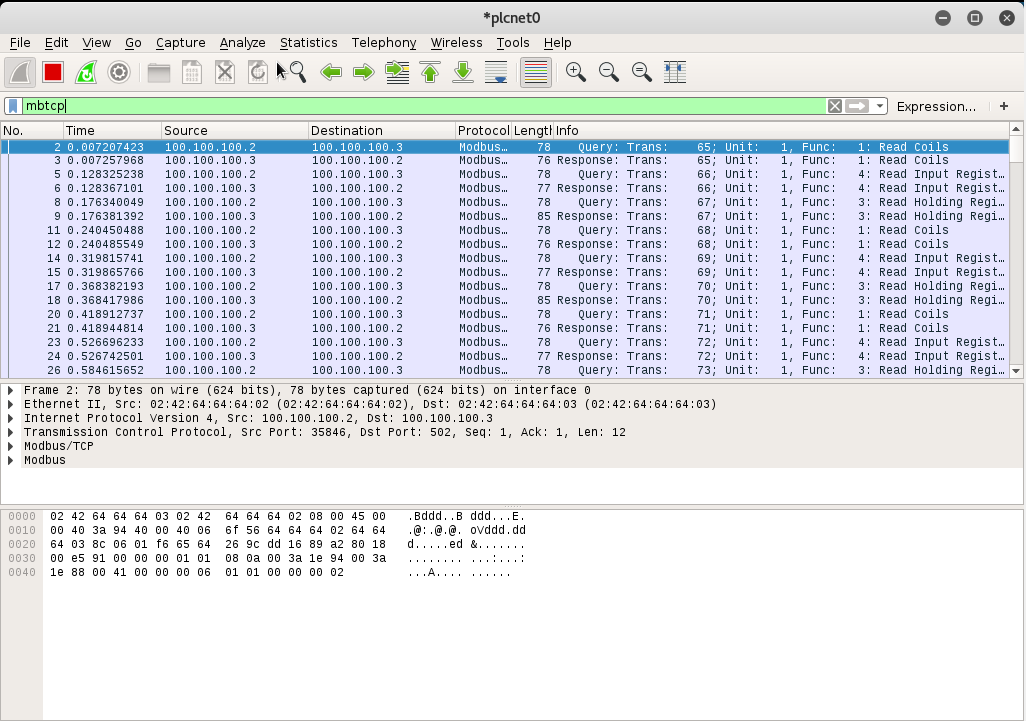
**Exercise #4 - Wireshark filters**

In this exercise, the student will become familiar with applying Wireshark filters to network traffic. Various filters can be used in the Wireshark packet capturing software.   
  
Apply the following filters to the traffic captured in Wireshark:

1. ip.addr == 100.100.100.1 this filter is used to focus on packets that are the source or destination for your virtual machine.
2. ip.addr == 100.100.100.1 && ip.addr == 100.100.100.3 this filter will show only traffic between your virtual machine and the PLC container.
3. http or dns this filter will show traffic over TCP port 80(HTTP) and TCP port 53(DNS)
4. tcp.port==502 this filter will show traffic over TCP port 502 (Modbus)
5. tcp.flags.reset this filter will show traffic with TCP reset flag
6. http.request this filter will show all HTTP get request
7. tcp contains traffic this filter will show all TCP traffic
8. !(http or dns) this filter will remove all HTTP and DNS traffic
9. Tcp.analysis.retransmission this filter will shows all retransmissions, helpful for finding what is taxing you network.

Another way to filter for Modbus traffic is *mbtcp*. Apply that filter to see the traffic between the HMI (100.100.100.2) and the PLC (100.100.100.3) containers.

\*NOTE: HTTP traffic and Modbus traffic is **not** generated if browser is not opened to 100.100.100.2:8080/ScadaBR and Water Tank Data Source enabled.

  
  
For a complete list of Wireshark filters, check out this list [here](file:///C:\Users\djl0012\Documents\Lab2-SCADA%20System%20Networking\Lab%20Walkthrough_References\pptx\wiresharkdisplayfilters.pdf) .

**Exercise #5 - Python script to open network socket and inject traffic**

In this exercise, the student will write and/or use Python to open a network socket and send Modbus traffic from the host.

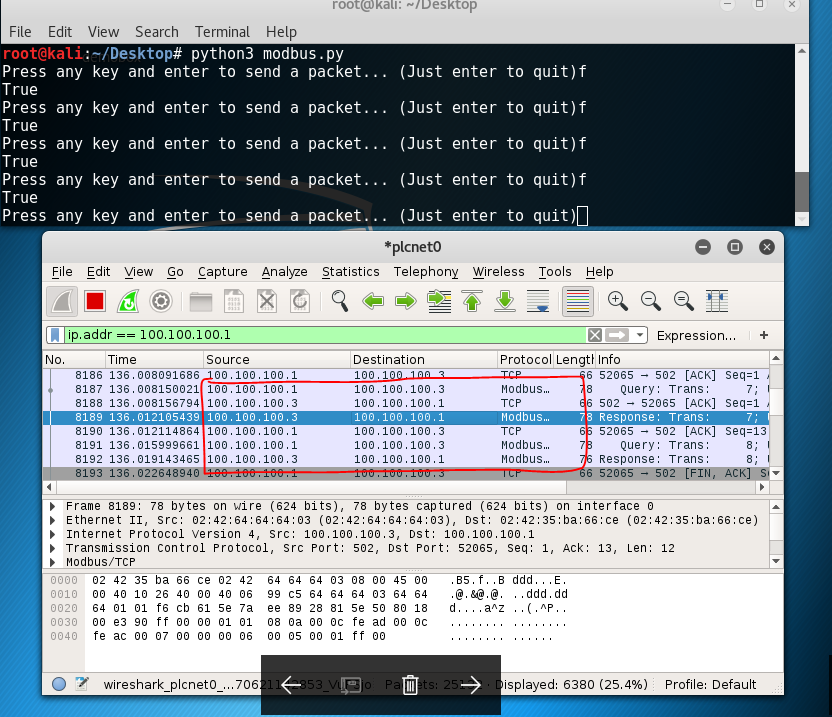
1. Open a terminal from the Lab 2 directory (/home/ccre/scadalab/lab2).
2. Run *nano modbus.py* to open the script to update the IP address.
3. Add the PLC IP (100.100.100.3) to the script and save and close. Save File.
4. Run *python3 modbus.py*
5. Press any key and enter to send a new packet. To quit, hit enter.

The python script for modbus.py.
First line. from pymodbus3.client.sync import ModbusTcpClient
The second line is blank.
Third line. inp = input(u"Press any key and enter to send a packet... (Just enter to quit)")
Fourth line. client = ModbusTcpClient('100.100.100.3')
Fifth line. while(inp):
Sixth line. cline.write_coil(1, True)
Seventh line. result = client.read_coils(1, 1)
Eighth line. print(result.bits[0])
Ninth line. client.close()
Tenth line. inp = input(u"Press any key and enter to send a packet... (Just enter to quit)")

**Exercise #6 - Observe injected traffic in Wireshark**

Now we can use Wireshark to capture the traffic injected by the Python script.

1. On virtual machine, launch Wireshark.
2. Go to the Capture>Options list and select the interface you want to begin capturing traffic on (plcnet0).
3. Use filter ip.addr == 100.100.100.1 to see the traffic from the virtual machine. Now find where the Modbus traffic injected from the host.



**Copyright:** University of Alabama in Huntsville