**LAB 1: INTRODUCTION TO SCADA CONTROL SYSTEMS**

**Manual**

**Student Name**

**Student ID**

**Student Email**

**Purpose**: The purpose of this Lab exercise is to familiarize the student with SCADA Control Systems, Docker containers and begin configuring the virtual environment that will be used throughout the remaining labs.

**Objective:** The student will research software, protocols and tools used by SCADA Control Systems and the components within the UAH environment. The student should be able to operate the control system environments at the conclusion of this lab.   
  
**Lab Setup and Requirements:** The student will need a browser with internet access to conduct research and download any necessary lab files. This lab assumes the student has previous experience using virtual environments.

**Exercise #1 - Introduction to Docker**

Read through the provided Docker Basics pdf file.

**Exercise #2 - Introduction to the UAH SCADA and Docker Environment**

The UAH environment will utilize Virtual Box and Docker to simulate two industrial control systems:

1. One Tank Water Pump-Three Docker Containers (PLC, HMI & Sbox)
2. One Station Gas Pipeline-Three Docker Containers (PLC, HMI & Sbox)

The Water Tank and Gas Pipeline can only be run individually.   
Login for the virtual machine = (username:ccre, password:ccre) (username:root, password:toor). \*Recommend logging in as ccre and using sudo to execute privileged commands. NOTE: Any time sudo is used to execute privileged commands you will have to enter the ccre password:ccre

1. PLC (Programmable Logic Controller) container is running on Debian with OpenPLC.
2. HMI (Human Machine Interface) container is running on Alpine using ScadaBR to display the HMI.
3. Sbox (Simulation Box) is running on Debian using a complied MATLAB model and an interface program to simulate the control system traffic.

The following scripts are provided on the virtual machine (/home/ccre/scadalab/scripts) to launch the Docker containers and configure the network environment. These scripts will be utilized throughout the labs, so familiarize yourself with the content and usage of each script.

1. gaspipeline.sh-Stops & removes plc0, HMI, & sbox containers, then starts HMI, plc0 & sbox, running scripts in plc0 & sbox to start the gas pipeline sim & ladder logic
2. watertank.sh - Stops & removes plc0, HMI, & sbox containers, then starts HMI, plc0 & sbox, running scripts in plc0 & sbox to start the Water tank sim & ladder logic
3. netstart.sh-First stops plcnet & datanet, then starts plcnet & datanet (Netstart should only be run once during initial setup of the environment. In addition, it will fail if there are any existing containers attached to either network)
4. netstop.sh-Stops plcnet & datanet
5. cleanup.sh-Brute Force, removes all containers

The following references are provided to supplement or provide additional information on the tools, protocols, and software used within this environment.

* [OpenPLC](http://www.openplcproject.com)
* [HMI](http://www.wisegeek.com/what-is-human-machine-interface.htm)
* [SCADABR](https://www.youtube.com/watch?v=JjOQWwoaQuQ)
* [MATLAB](https://www.mathworks.com/products/matlab.html)
* [ModBus](http://modbus.org/)
* [Virtual Box](https://www.virtualbox.org/)
* [OpenPLC Editor](http://plcedit.org/)

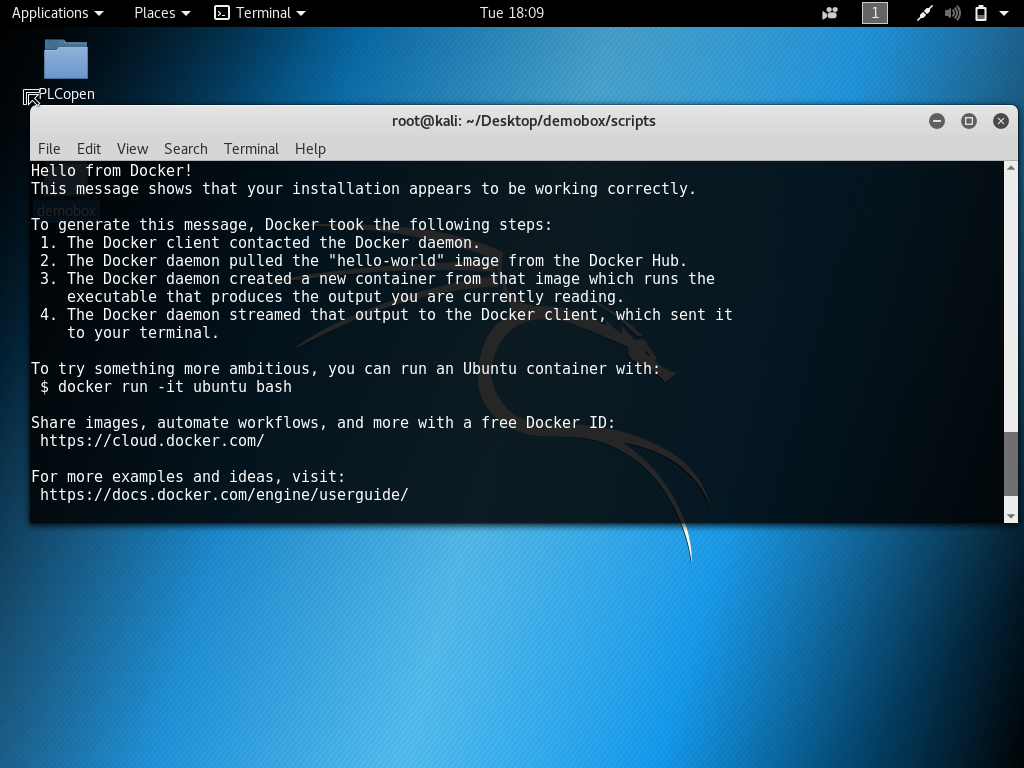
**Exercise #3 - Setup SCADA LAB Environment**

Section 1: Import the virtual machine into virtual box.

1. Download and install [Virtual Box](https://www.virtualbox.org/) on your host machine.
2. Copy lab files from the UAH Cybersecurity SCADA Labs disc provided.
3. Extract files to a destination on your host computer for use throughout the remaining labs & exercises.
4. In the Virtual Box Manager, select File>Import Appliance to import the virtual machine.
5. Click "Choose a virtual appliance file to import" icon and browse to the scadalab.ova. Click "Import".

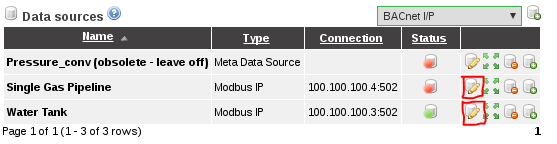
Section 2: Login to virtual machine and run Water Tank Docker containers.

1. To start the virtual machine in the VirtualBox Manager, select the scadalab VM, right-click and select Start>Normal Start. Login using the credentials provided in Exercise 2 (username:ccre, password:ccre).
2. Select "Use default config" on first launch.
3. The Docker containers have already been loaded on the virtual machine. To verify Docker is loaded, open a terminal and run *sudo docker run hello-world*. Enter [sudo] password for ccre: ccre. You should receive the following response.

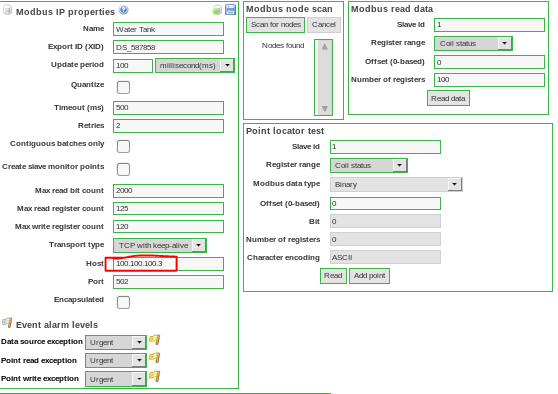


If this command says that Docker is dead or inactive you need to start it by running *sudo service docker start*

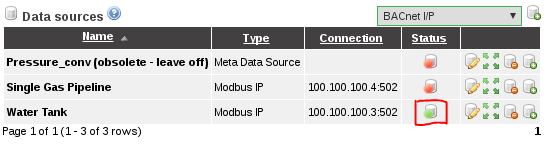
1. Now navigate to the scripts folder (/home/ccre/scadalab/scripts) and run the script to configure the network (*./netstart.sh*). Then start the Water Tank containers, run *./watertank.sh*.
2. To verify the containers loaded and started correctly, run *sudo docker ps* in a terminal to view the status of the containers.
3. To launch the water tank HMI, pull up the internet browser on the virtual machine and navigate to 100.100.100.2:8080/ScadaBR then login to ScadaBR (username:admin, password:admin).
4. Verify the IP address of the Modbus data source is pointing to your PLC container (100.100.100.3-Water Tank & 100.100.100.4-Gas Pipeline). Click on Data Sources on the top menu and then on the edit button for the Water Tank.



1. Once the edit screen is opened, verify the IP address matches the PLC container IP (100.100.100.3 for the Water Tank & 100.100.100.4 for the Gas Pipeline) and then click on the blue floppy disk to save.



1. Go back to the Data Sources screen and the enable water tank data sources to allow ScadaBR to pull data from OpenPLC. (NOTE: PLC container must be running for this to work. If PLC container is down, ScadaBR will throw a lot of errors saying that it can't communicate with the data source.)



1. Click on Graphical Views, select the HMI you want to use from the drop down menu.



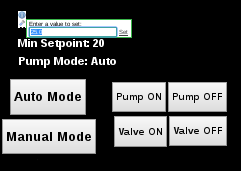
1. Verify the levels are changing on the Water Tank HMI.
2. Disable the Water Tank Data Source and enable the Gas Pipeline Data Source. Repeat steps 4-10 for the One Station Gas Pipeline utilizing the gaspipeline.sh script. (Note: the netstart.sh script only needs to be run initially on setup).
3. Disable all data sources and run cleanup script (./cleanup.sh)

CHECKPOINT: Refer to Troubleshooting Guide for FAQs regarding configuration and setup.

**Exercise #4 - Interact with HMI and Configure for Unsafe Operations**

Section 1: Interact with the Water Tank HMI

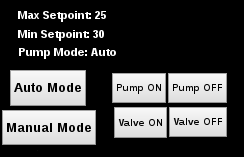
1. To start the Water Tank containers, run *./watertank.sh*.
2. Enable the Water Tank Data Source.
3. Pull up the Water Tank HMI in ScadaBR. Turn the Pump On by turning on Manual Mode and clicking the "Pump On" button. Wait ~30 seconds and observe the water tank activity. Now turn the Pump Off by clicking the "Pump Off" and observe the activity.
4. Turn the HMI on Auto and observe the fluctuations in water levels.
5. Change setpoints to Minimum=20 and Maximum=25 and observe difference when range is narrowed.



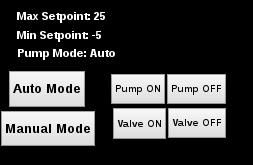
Section 2: Configure Water Pump for Unsafe Operation

The following modifications could be made by an attacker which would cause undesirable results.

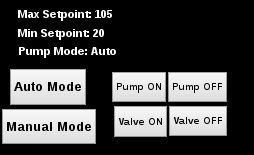
1. Attack 1: Set the Minimum setpoint more than the Maximum setpoint (Expected result: This turns the pump on/off frequently which might damage the pump. The level stabilizes at maximum setpoint)



1. Attack 2: Set the Minimum setpoint to -5. (Expected Result: Tank completely empties. The pump never starts if the model is started from zero)



1. Attack 3: Maximum setpoint set to more than 100 (Expected Result: HMI becomes unreliable and the tank overflows)



**Copyright:** University of Alabama in Huntsville