PLC Application Firewall and Software Whitelists

1 Overview

This lab explores security issues related to the use of Programmable Logic Controllers (PLCs) in the management of Industrial Control Systems (ICS), or similar forms of infrastructure. You should read this "Overview" and the following "Background" section before starting the lab.

1.1 Learning objectives

PLCs typically receive commands from networks containing multiple computers. Not all of these networked computers are necessarily authorized to issue all commands to the PLC. For example, some computers may be authorized to issue commands that monitor the PLC without affecting its behavior, while other computers are designated as being able to reset or reconfigure the PLC. One way to enforce this type of application policy is to use a firewall that serves as a proxy between the network computers and the PLC. These firewalls are designed to decode the commands destined for the PLC, and only permit those that meet the policy for which computer can issue which commands.

Limiting the computers that can alter a PLC's configuration does not ensure that the PLC will be loaded with a valid configuration. Malicious software on an authorized computer could load the PLC with programs or data intended to damage the infrastructure. One way to limit the ability of malicious software (or individuals) to reconfigure the PLC is to enumerate a set of validated program and configuration files. A "whitelist" of cryptographic checksums (or digests) for each valid file can then be loaded into a proxy that sits between the computers and the PLC. The proxy, (or firewall), would then only permit those files having validated digests, i.e., those whose digests appear in the whitelist.

1.2 Simulated infrastructure control system

This *plc-app* lab simulates the system illustrated in Figure 1. A PLC manages the water level of a creek-fed catfish pond, ensuring the water level stays within minimum and maximum limits. You will interact with the sys_management system to load a program and configuration data into the PLC. You will also use the sys_management computer to check the status of the PLC and to query which program and configuration data the PLC is running.

The monitor system is used to query the status of the PLC (which can also be performed at the sys_management system). The monitor also contains the "historian" subsystem which keeps a running log of the PLC status. The monitor system must be able to continually monitor the PLC, or the farmer will fail the insurance company audit of his crop damage policy.

You will not have direct access to the PLC subsystem, though you can interact with it via the sys_management and monitor computers.

A "Firewall" sits between the sys_management and montior computers and the PLC. This device can be configured to:

1. Filter commands destined for the PLC, constraining the commands that may be issued from a given IP address.

2. Prevent unauthorized programs or data from being loaded into the PLC. The firewall uses a whitelist of authorized MD5 digests to validate files destined for the PLC.

The firewall is initially in its default configuration, which imposes not limits on network traffic destined for the PLC.

1.3 Background

The student is expected to have performed the Labtainer "onewayhash" lab, or otherwise learned about the use of openssl to generate digests.

The student is expected to have some familiarity with the Linux command line, the basics of the file system.

2 Lab Environment

This lab runs in the Labtainer framework, available at http://my.nps.edu/web/c3o/labtainers. That site includes links to a pre-built virtual machine that has Labtainers installed, however Labtainers can be run on any Linux host that supports Docker containers.

From your labtainer-student directory start the lab using:

```
labtainer plc-app
```

A link to this lab manual will be displayed. The resulting virtual terminals will include:

- A display of the status of the fish pond level, titled "Physical_World".
- A bash shell on the sys_management computer.
- A bash shell on the monitor computer.
- A bash shell on the Firewall, titled "admin@firewall".
- A display of the Firewall log file titled "FIREWALL_LOG".

NOTE: When the lab starts, observe the Physical_World window. The PLC is initially disabled, and thus the pump does not run and the water rises. Throughout the lab, you will not be penalized for initial floods or other disasters. You will, however, eventually need to configure the systems to avoid those.

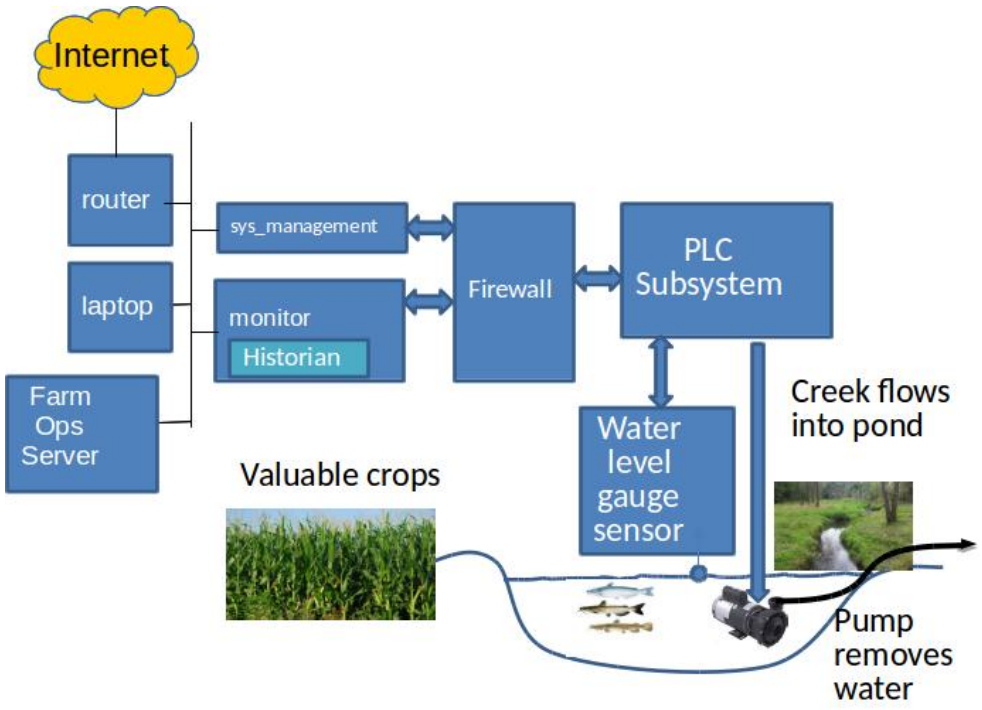

Figure 1: Network topology for the plc-app lab

3 Lab Tasks

3.1 Explore

The Physical World display is notional, it is not generated by any of the components of figure 1. It helps you understand what is happening in the physical world, independent of the subsystems. Use:

```
manage_plc status
```

from the sys_management and monitor systems to observe the state of the PLC. Observe the log messages on the Firewall log. Notice how there is periodic traffic? That is from a service on the monitor computer. If you wait long enough, you will notice that the farmer's field floods.

3.2 Load the PLC for the rainy season

The sys_management computer is used by the farmer to load software into the PLC. The

```
manage_plc load <program> <config>
```

command is used to load the PLC with a given program and configuration data. It is now the rainy season, so you should specify the `config_wet.txt` configuration. Initialize the PLC from the sys_management window using:

```
manage_plc load plc config_wet.txt
```

The ”plc” parameter is the name of the plc program file in your home directory. This operation will initialize the PLC, causing to the pump to run. The rainy season configuration file directs the PLC to keep the pond level between 15 and 25 feet, allowing the pond to absorb bursts of flow from the creek without flooding the fields.

Use the `manage_plc status` command to observe that the PLC is now operating, and controlling the pump.

3.3 Constrain PLC commands based on IP address

Go to the “monitor” system and use the `manage_plc status` command to view the status. This monitor computer is in the farm yard and is used to keep an eye on the PLC. However, a trained chicken is known to peck the keyboard. If you type or peck the `manage_plc reset` command from the monitor computer, you will notice that `manage_plc status` indicates the PLC operation has stopped.

For this task, you need to configure the firewall (on the firewall computer) to allow the sys_management computer to issue all PLC commands, but only allow the monitor computer to issue the “status” and “retrieve” commands. Use `firewall -h` on the firewall computer to learn about configuring the firewall filters to limit PLC commands from different IP addresses. Use `ifconfig` on the sys_management and monitor computers to learn their addresses. Note that before you modify filters, you will need to stop the firewall using:

```
sudo systemctl stop firewall
```

and then, after configuring the filters, restart the firewall using:

```
sudo systemctl start firewall
```

After you have configured the firewall and restarted it, do the following:

- Reload the PLC from the sys_management computer:

  ```
  manage_plc load plc config_wet.txt
  ```

- Check the status from the sys_management computer:

  ```
  manage_plc status
  ```

- Attempt to reset the PLC from the monitor computer:

  ```
  manage_plc reset
  ```

- Confirm the PLC is still running, i.e., the reset command was blocked:

  ```
  manage_plc status
  ```

Each of the above operations must yield the desired result. If not, return to the firewall and correct its configuration. Then repeat each and every one of the above steps to demonstrate proper application filter settings.

3.4 Configure the PLC for the dry season

Now that it is the dry season, Farmer Jones wants the pond to hold more water. From the sys_management terminal, use:

```
manage_plc load plc config_dry.txt
```

to configure the PLC for the dry season. Then just watch what happens over the course of about a minute. After you've watched the Physical world status window and observed a disaster, poke around a bit.

On the monitor system, view the `historian.log` file in the home directory. Do you see any suspicious parameter settings?

From the sys_management terminal, use:

```
manage_plc reset
manage_plc load plc config_dry.txt
```

to reset and reload the PLC. Watch the firewall log, do you notice any suspicious traffic? And again review the historian.log. Then use the `manage_plc retrieve` command to retrieve the program and configuration file that put the PLC into this state. Are they the files you loaded? Note, now is a good time to consider revisiting the use of the `openssl dgst -md5` command.

3.5 Living with malware

It turns out the farmer had installed a bootleg copy of pharmvilla on his laptop computer, and that introduced malware onto the sys_mangement computer. That malware loads corrupt PLC programs and configuration data. In this lab, we assume you are unable to remove the malware or prevent it from running. (If you view that statement as a challenge, please complete the lab as intended before chasing the malware.) Your task is to configure the firewall so that it only permits files having validated MD5 digests to be loaded onto the firewall.

On the firewall computer, use the `firewall -h` command to learn how to use a whitelist of MD5 digests.

Before defining a whitelist on the firewall, you must first stop it:

```
sudo systemctl stop firewall
```

and then, after configuring the whitelist, restart the firewall using:

```
sudo systemctl start firewall
```

Then use

```
manage_plc reset
manage_plc load plc config_dry.txt
```

to restore the PLC to its desired configuration and patiently wait until the physical word informs you that you have completed the lab – or until you notice something not work. If something goes wrong, review your MD5 checksums and logs and try again.

4 Submission

After finishing the lab, go to the terminal on your Linux system that was used to start the lab and type:

```
stoplab
```

When you stop the lab, the system will display a path to the zipped lab results on your Linux system. Provide that file to your instructor, e.g., via the Sakai site.

This lab was developed for the Labtainer framework by the Naval Postgraduate School, Center for Cybersecurity and Cyber Operations under National Science Foundation Award No. 1438893. This work is in the public domain, and cannot be copyrighted.