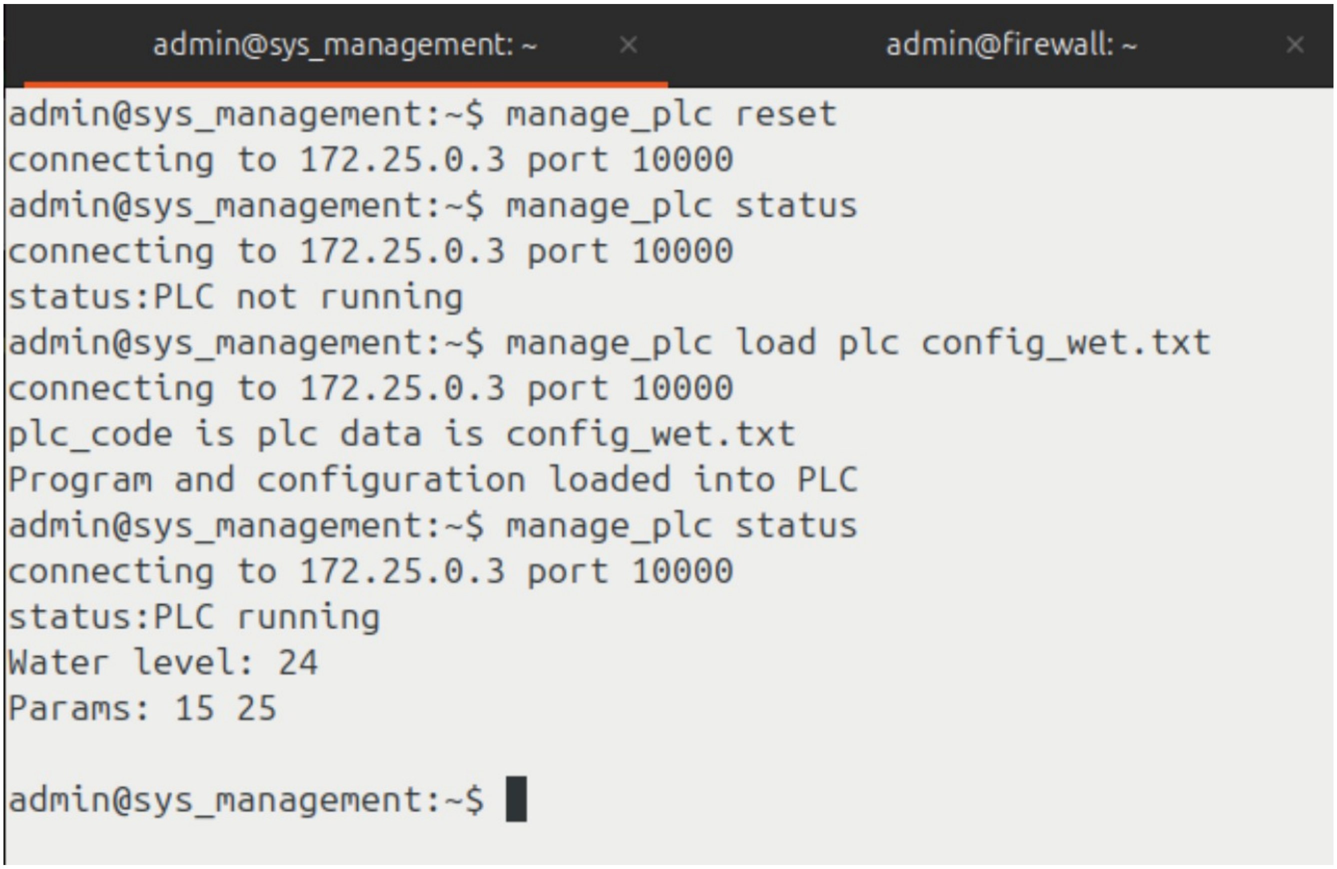
Alex Shah
EN.695.741.81.SP25 Information Assurance Analysis
Mod 13 Lab PLC
April 27, 2025

Setup

I downloaded the labtainer VM and ran it using Virtualbox. I started the lab by running "labtainer plc-app" which launched several terminal windows, and then I followed the lab steps.

3.1-3.2

Initially the pump is not running so the water level rises and the crops get flooded. On the management terminal, running "manage_plc status" shows the PLC is not running. Next I ran "manage_plc load plc config_wet.txt" to load the rainy season configuration, and the status this is time is that the PLC is running with the current water level and the minimum and maximum parameters for the pump settings.

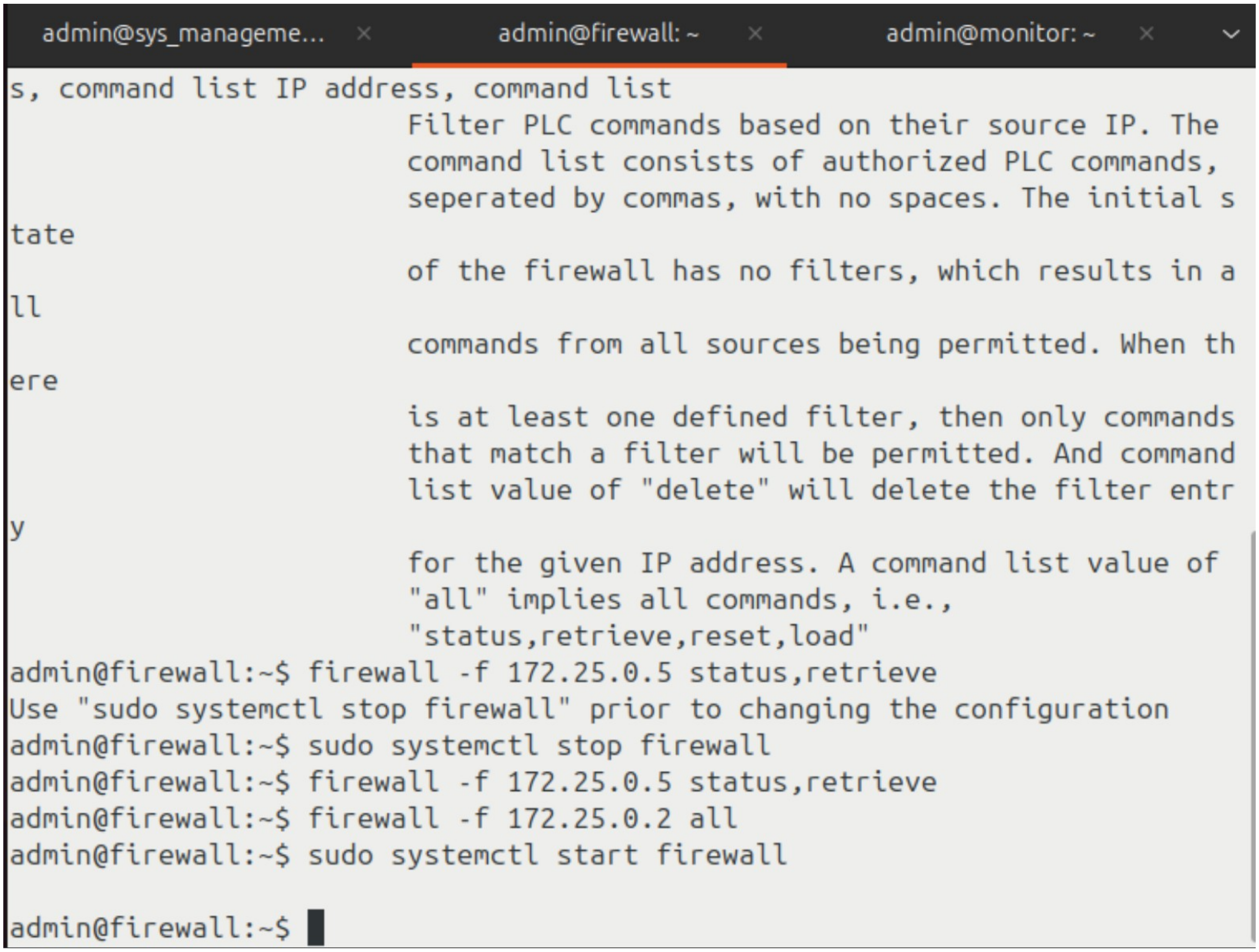


```
admin@sys_management:~$ manage_plc reset
connecting to 172.25.0.3 port 10000
admin@sys_management:~$ manage_plc status
connecting to 172.25.0.3 port 10000
status:PLC not running
admin@sys_management:~$ manage_plc load plc config_wet.txt
connecting to 172.25.0.3 port 10000
plc_code is plc data is config_wet.txt
Program and configuration loaded into PLC
admin@sys_management:~$ manage_plc status
connecting to 172.25.0.3 port 10000
status:PLC running
Water level: 24
Params: 15 25

admin@sys_management:~$ 
```

Figure 1: PLC status before and after loading a config

3.3

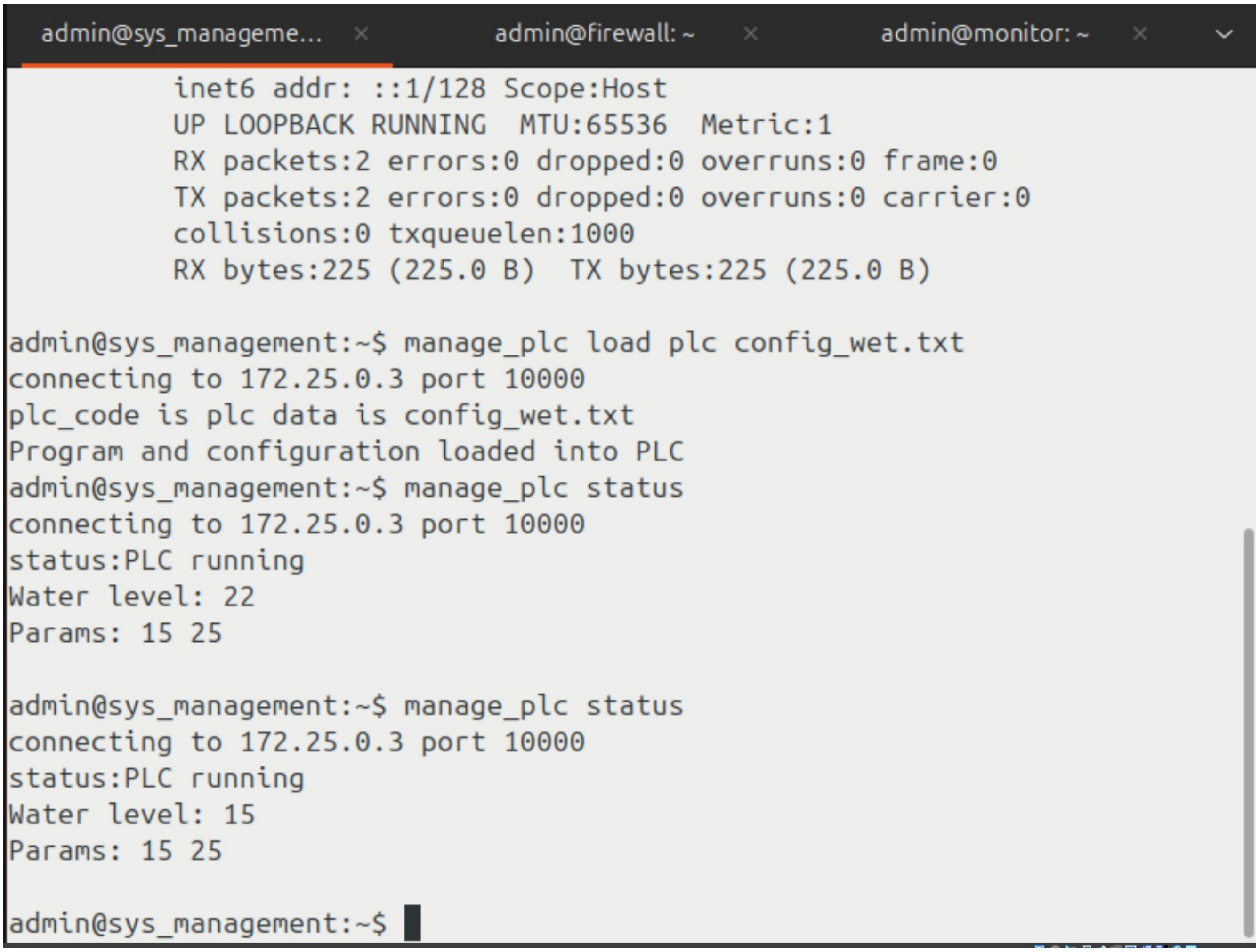
Next in the firewall terminal stopped the firewall with “sudo systemctl stop firewall” and used the -f flag to filter IP address capabilities when connecting to the PLC. I configured the monitor IP address to view the status and retrieve the details from the PLC but did not allow it to reset or load configurations to the PLC. I ran “firewall -f 172.25.0.5 status,retrieve” to limit the monitor device, and “firewall -f 172.25.0.2 all” to allow the management device all functionality.



```
s, command list IP address, command list
                        Filter PLC commands based on their source IP. The
                        command list consists of authorized PLC commands,
                        seperated by commas, with no spaces. The initial s
tate
                        of the firewall has no filters, which results in a
ll
                        commands from all sources being permitted. When th
ere
                        is at least one defined filter, then only commands
                        that match a filter will be permitted. And command
                        list value of "delete" will delete the filter entr
y
                        for the given IP address. A command list value of
                        "all" implies all commands, i.e.,
                        "status,retrieve,reset,load"
admin@firewall:~$ firewall -f 172.25.0.5 status,retrieve
Use "sudo systemctl stop firewall" prior to changing the configuration
admin@firewall:~$ sudo systemctl stop firewall
admin@firewall:~$ firewall -f 172.25.0.5 status,retrieve
admin@firewall:~$ firewall -f 172.25.0.2 all
admin@firewall:~$ sudo systemctl start firewall

admin@firewall:~$ 
```

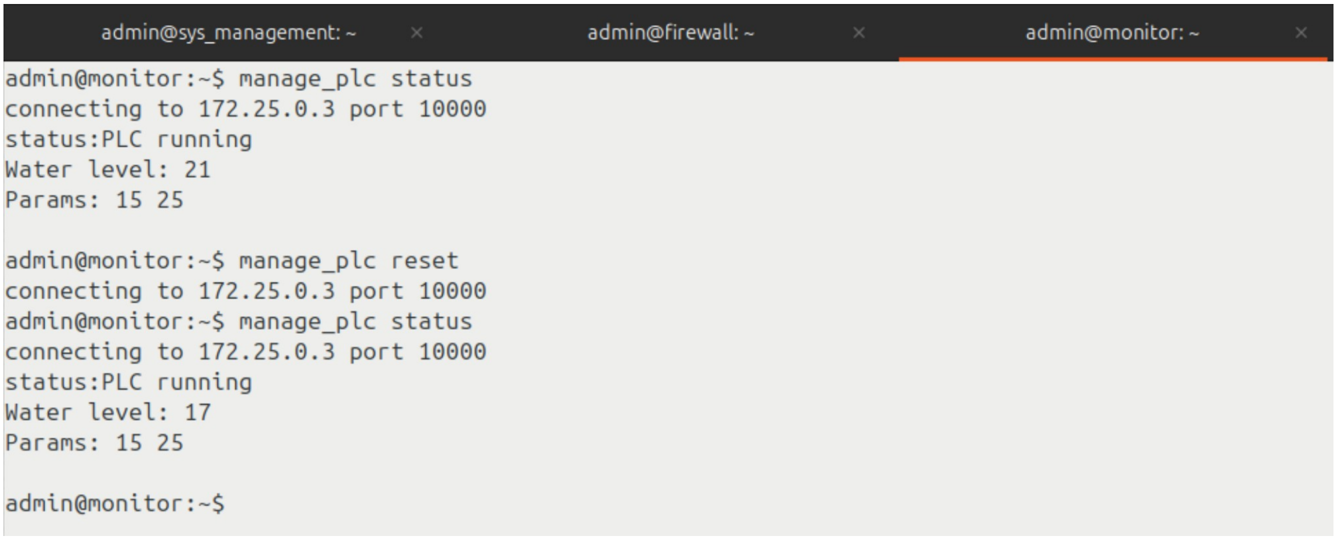
Figure 2: Setting firewall filters for the management and monitor devices

After limiting capabilities with the firewall, I loaded a config and checked the status of the PLC on the management device and was able to do so. I tried resetting the device from the monitor, which shouldn’t be allowed, and it did not reset the device.



```
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:225 (225.0 B)  TX bytes:225 (225.0 B)

admin@sys_management:~$ manage_plc load plc config_wet.txt
connecting to 172.25.0.3 port 10000
plc_code is plc data is config_wet.txt
Program and configuration loaded into PLC
admin@sys_management:~$ manage_plc status
connecting to 172.25.0.3 port 10000
status:PLC running
Water level: 22
Params: 15 25

admin@sys_management:~$ manage_plc status
connecting to 172.25.0.3 port 10000
status:PLC running
Water level: 15
Params: 15 25

admin@sys_management:~$
```

Figure 3: Management device can load configs and check status after filtering in the firewall

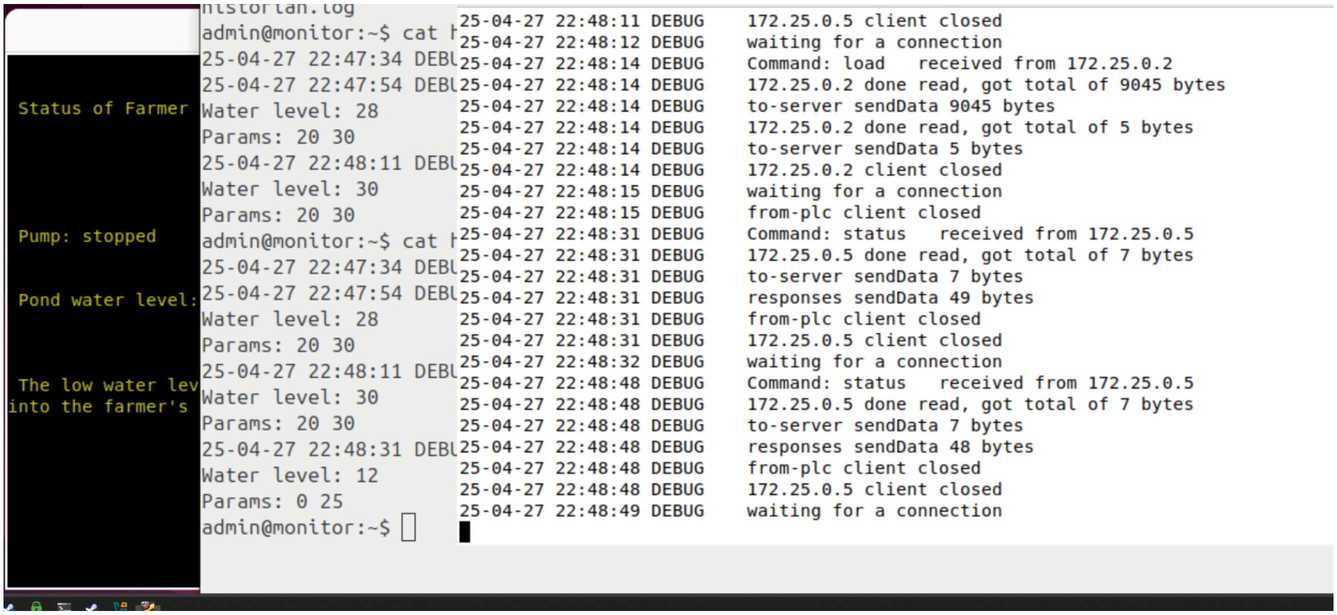


```
admin@monitor:~$ manage_plc status
connecting to 172.25.0.3 port 10000
status:PLC running
Water level: 21
Params: 15 25

admin@monitor:~$ manage_plc reset
connecting to 172.25.0.3 port 10000
admin@monitor:~$ manage_plc status
connecting to 172.25.0.3 port 10000
status:PLC running
Water level: 17
Params: 15 25

admin@monitor:~$
```

Figure 4: Monitor device cannot reset the PLC after filtering in the firewall

3.4

After loading the dry config, the parameters change in the historian.log. This means that something changed the configuration on the PLC after the config had been loaded and caused the water level to be too low. There are lines in the firewall log around the time the config changes in the historian.log, showing a command to load a config from the management device IP.



```
Status of Farmer

Pump: stopped

Pond water level:

The low water lev
into the farmer's
```

```
nistorian.log
admin@monitor:~$ cat h
25-04-27 22:47:34 DEBU
25-04-27 22:47:54 DEBU
Water level: 28
Params: 20 30
25-04-27 22:48:11 DEBU
Water level: 30
Params: 20 30
admin@monitor:~$ cat h
25-04-27 22:47:34 DEBU
25-04-27 22:47:54 DEBU
Water level: 28
Params: 20 30
25-04-27 22:48:11 DEBU
Water level: 30
Params: 20 30
25-04-27 22:48:31 DEBU
Water level: 12
Params: 0 25
admin@monitor:~$ 
```

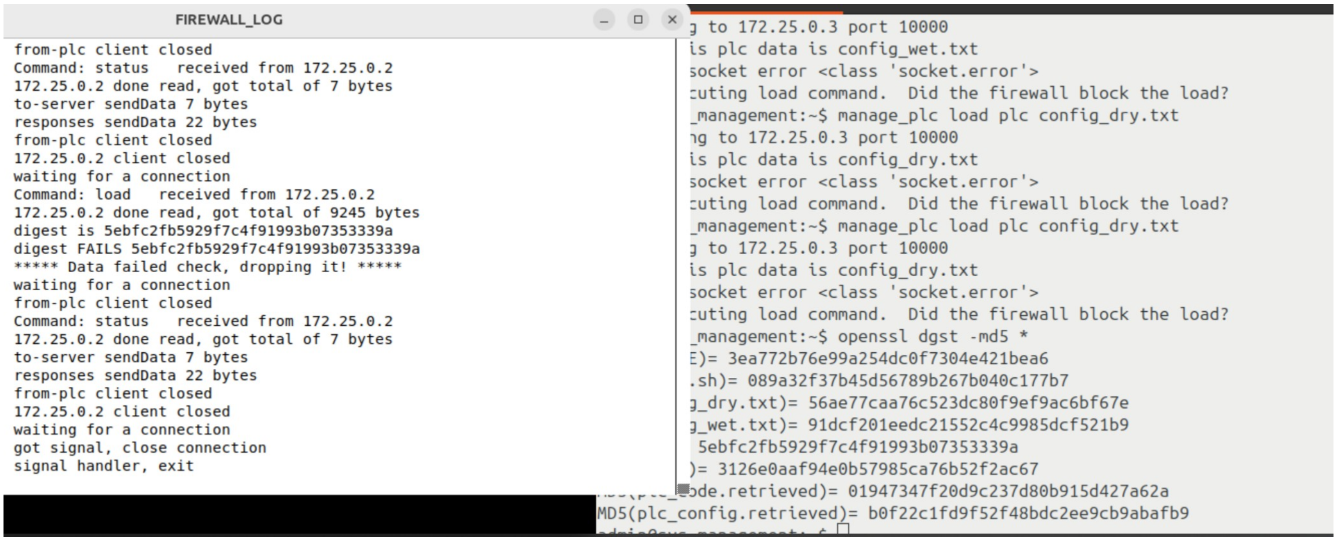
```
25-04-27 22:48:11 DEBUG    172.25.0.5 client closed
25-04-27 22:48:12 DEBUG    waiting for a connection
25-04-27 22:48:14 DEBUG    Command: load   received from 172.25.0.2
25-04-27 22:48:14 DEBUG    172.25.0.2 done read, got total of 9045 bytes
25-04-27 22:48:14 DEBUG    to-server sendData 9045 bytes
25-04-27 22:48:14 DEBUG    172.25.0.2 done read, got total of 5 bytes
25-04-27 22:48:14 DEBUG    to-server sendData 5 bytes
25-04-27 22:48:14 DEBUG    172.25.0.2 client closed
25-04-27 22:48:15 DEBUG    waiting for a connection
25-04-27 22:48:15 DEBUG    from-plc client closed
25-04-27 22:48:31 DEBUG    Command: status   received from 172.25.0.5
25-04-27 22:48:31 DEBUG    172.25.0.5 done read, got total of 7 bytes
25-04-27 22:48:31 DEBUG    to-server sendData 7 bytes
25-04-27 22:48:31 DEBUG    responses sendData 49 bytes
25-04-27 22:48:31 DEBUG    from-plc client closed
25-04-27 22:48:31 DEBUG    172.25.0.5 client closed
25-04-27 22:48:32 DEBUG    waiting for a connection
25-04-27 22:48:48 DEBUG    Command: status   received from 172.25.0.5
25-04-27 22:48:48 DEBUG    172.25.0.5 done read, got total of 7 bytes
25-04-27 22:48:48 DEBUG    to-server sendData 7 bytes
25-04-27 22:48:48 DEBUG    responses sendData 48 bytes
25-04-27 22:48:48 DEBUG    from-plc client closed
25-04-27 22:48:48 DEBUG    172.25.0.5 client closed
25-04-27 22:48:49 DEBUG    waiting for a connection
```

Figure 5:

Left: pump lets water get too low,
Middle: historian.log shows configuration changes between 22:48:11 and 22:48:31,
Right: firewall log shows a load command and 9045 bytes sent from management device at 22:48:14 which changed the config

3.5

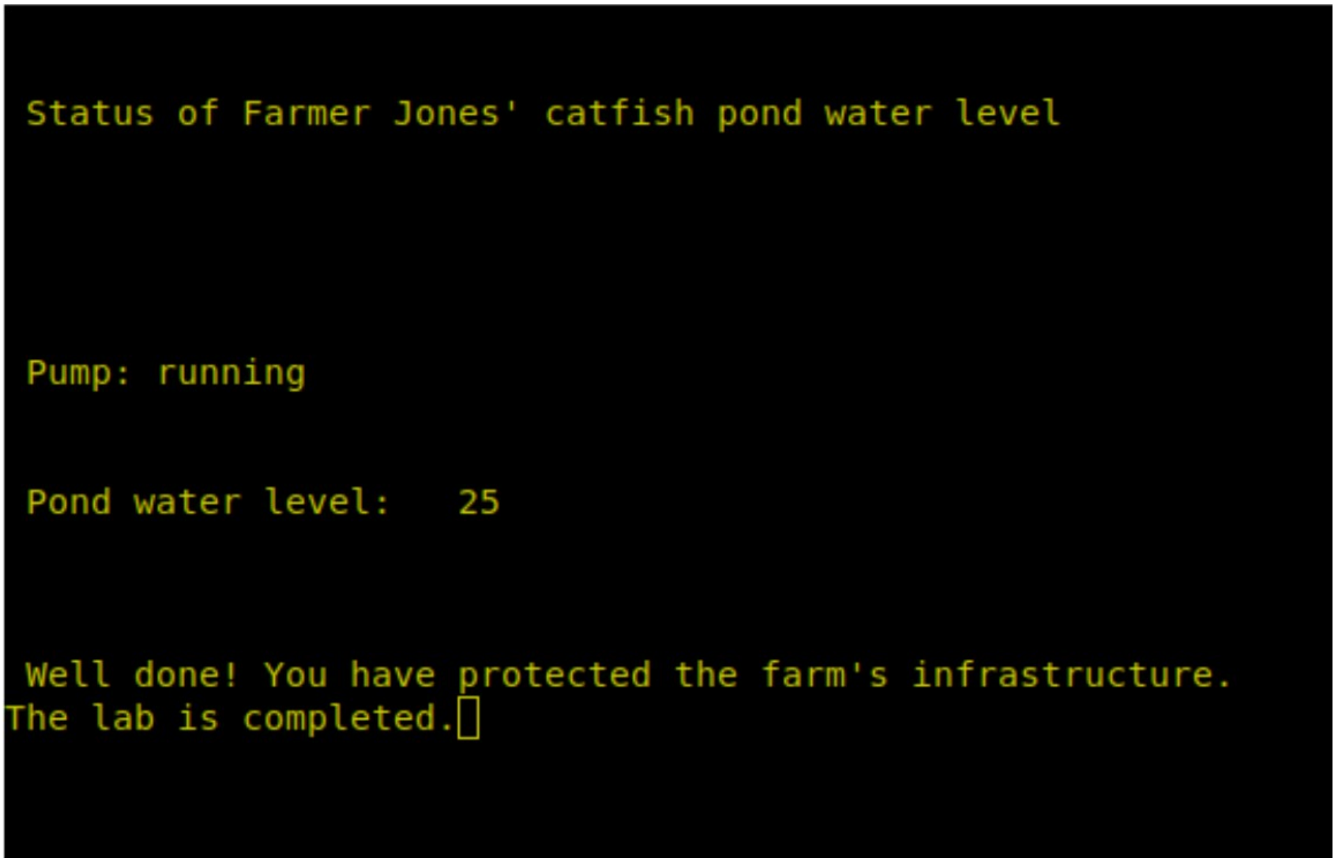
Using "openssl dgst -md5" I determined the md5 hash values for the files being loaded. This ended up being the "plc" file not the "config_wet.txt" or "config_dry.txt" files. I found out which file hash was needed by looking at the firewall log and then running "openssl dgst -md5 *" to find the hash of the files in the folder and find out which of them was the one that matched the hash being sent through the firewall. I added the hash value to the firewall allow list by running "firewall -a <hash>". Then I reset and reloaded the dry config on the PLC again and saw the success message.



```
from-plc client closed
Command: status   received from 172.25.0.2
172.25.0.2 done read, got total of 7 bytes
to-server sendData 7 bytes
responses sendData 22 bytes
from-plc client closed
172.25.0.2 client closed
waiting for a connection
Command: load   received from 172.25.0.2
172.25.0.2 done read, got total of 9245 bytes
digest is 5ebfc2fb5929f7c4f91993b07353339a
digest FAILS 5ebfc2fb5929f7c4f91993b07353339a
***** Data failed check, dropping it! *****
waiting for a connection
from-plc client closed
Command: status   received from 172.25.0.2
172.25.0.2 done read, got total of 7 bytes
to-server sendData 7 bytes
responses sendData 22 bytes
from-plc client closed
172.25.0.2 client closed
waiting for a connection
got signal, close connection
signal handler, exit
```

```
g to 172.25.0.3 port 10000
is plc data is config_wet.txt
socket error <class 'socket.error'>
cuting load command.  Did the firewall block the load?
_management:~$ manage_plc load plc config_dry.txt
ng to 172.25.0.3 port 10000
is plc data is config_dry.txt
socket error <class 'socket.error'>
cuting load command.  Did the firewall block the load?
_management:~$ manage_plc load plc config_dry.txt
g to 172.25.0.3 port 10000
is plc data is config_dry.txt
socket error <class 'socket.error'>
cuting load command.  Did the firewall block the load?
_management:~$ openssl dgst -md5 *
E)= 3ea772b76e99a254dc0f7304e421bea6
.sh)= 089a32f37b45d56789b267b040c177b7
g_dry.txt)= 56ae77caa76c523dc80f9ef9ac6bf67e
g_wet.txt)= 91dcf201eedc21552c4c9985dcf521b9
 5ebfc2fb5929f7c4f91993b07353339a
)= 3126e0aaf94e0b57985ca76b52f2ac67
ode.retrieved)= 01947347f20d9c237d80b915d427a62a
MD5(plc_config.retrieved)= b0f22c1fd9f52f48bdc2ee9cb9abafb9
```

Figure 6: Finding the hash value needed for the allowlist in the firewall log for “plc”



```
Status of Farmer Jones' catfish pond water level



Pump:  running


Pond water level:   25



Well done! You have protected the farm's infrastructure.
The lab is completed.
```

Figure 7: Success message after loading the dry config without issues