

Final Project Report

Understanding the Usage of Industrial Control System Devices on the Internet

- Expand your final project proposal to include the following
 - Report on Implementation and Results
 - Report on Contribution

As the industrial control systems play an important role in the critical infrastructure. And they are using the SCADA to control the remote ICS devices so it is important to understand the usage of those devices. This paper is concentrating on understanding the usage of the ICS devices online by getting the data such as distribution features, functions and the users of ICS devices.

So they first started with visible ICS devices online. In this paper they first started discovering online ICS devices in the intrusive behaviour by analysing the 17 industrial protocols and trained the algorithm to improve the detection processes. And then reduced the negative effects caused by the honeypots using some trained algorithm.

To perform on the whole IPV4 space they deployed it on the EC2 instance and then they analysed the data based on the region and functions of those ICS devices.

In this Paper I have implemented the algorithm to scan the space for the ICS devices which are using the modbus protocol by pinging if they are alive or not.

I used the python programming language to ping the devices which are alive or not then with that list of devices I checked if they are using the modbus protocol or not. Then at last I got the list of devices which are alive and using modbus protocol.

Implementation and results

- Describe how you implemented your own simulations/experiments. Include details about data, code, or network setup that you used.

I have created a simulation of the device which is using the modbus protocol for the communication or to send some message.

I have to linux virtual machine in which one is having the simulation to the modbus slave simulator which will be leasing to the master by starting its server in the port 502 by creating some connection.

Then the other virtual linux machine will be having the modbus master simulator where you can connect to the slave by specifying the ip and port of the device. Then when the slave accepts the connection they can send the message between the two device using th e TCP modbus protocol.

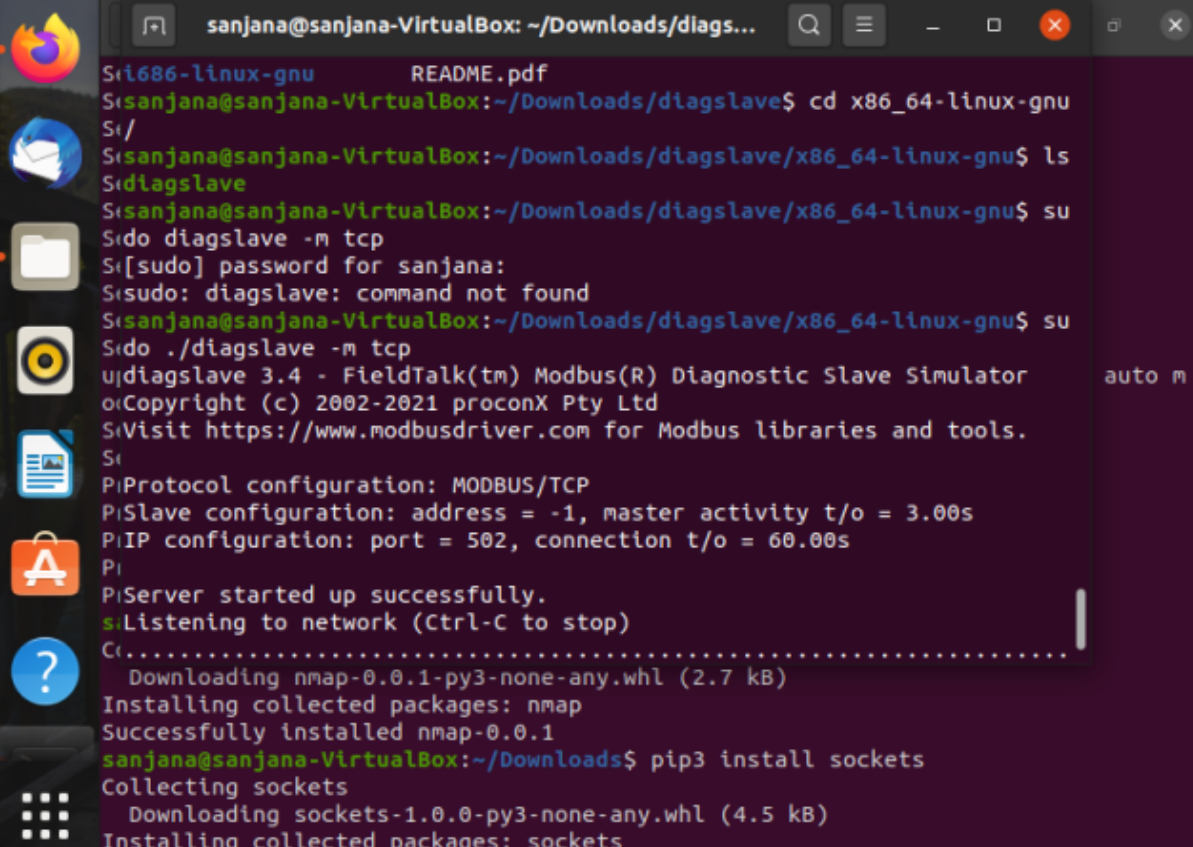
Then the other linux virtual machine will be having the algorithm to scan for the active ICS devices online. That algorithm will be pinning the devices and then it will be adding the IP address of those ICS devices to the list then it will check if that is using the modbus protocol that is port number 502. Then that will be printing the list of devices which are active and using the modbus protocol.

```
See "man sudo_root" for details.

sanjana@sanjana-VirtualBox:~$ cd Downloads/
sanjana@sanjana-VirtualBox:~/Downloads$ ls
modpoll-3.10.tgz
sanjana@sanjana-VirtualBox:~/Downloads$ tar xzf modpoll-3.10.tgz
sanjana@sanjana-VirtualBox:~/Downloads$ ls
modpoll  modpoll-3.10.tgz
sanjana@sanjana-VirtualBox:~/Downloads$ cd modpoll/
sanjana@sanjana-VirtualBox:~/Downloads/modpoll$ ls
aarch64-linux-gnu  i686-linux-gnu  LICENSE-FREE.txt  README.txt
arm-linux-gnueabihf  LICENSE-FREE.pdf  README.pdf        x86_64-linux-gnu
sanjana@sanjana-VirtualBox:~/Downloads/modpoll$ cd x86_64-linux-gnu/
sanjana@sanjana-VirtualBox:~/Downloads/modpoll/x86_64-linux-gnu$ ls
modpoll
sanjana@sanjana-VirtualBox:~/Downloads/modpoll/x86_64-linux-gnu$ sudo ./modpoll
-t4:float -r 100 -c 5 -1 10.0.2.15
[sudo] password for sanjana:
modpoll 3.10 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2021 proconX Pty Ltd
Visit https://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP, FC3
Slave configuration...: address = 1, start reference = 100, count = 5
Communication.....: 10.0.2.15, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 32-bit float, output (holding) register table
```

This is the screen shot of the linux virtual machine which is using the master simulation and it is connecting with the other virtual machine with the ip address of 10.0.2.15 with the port number 502.



```
sanjana@sanjana-VirtualBox: ~/Downloads/diags...  
Si686-linux-gnu README.pdf  
Si: sanjana@sanjana-VirtualBox: ~/Downloads/diagslave$ cd x86_64-linux-gnu  
Si:/  
Si: sanjana@sanjana-VirtualBox: ~/Downloads/diagslave/x86_64-linux-gnu$ ls  
Si: diagslave  
Si: sanjana@sanjana-VirtualBox: ~/Downloads/diagslave/x86_64-linux-gnu$ su  
Si: sudo diagslave -m tcp  
Si: [sudo] password for sanjana:  
Si: sudo: diagslave: command not found  
Si: sanjana@sanjana-VirtualBox: ~/Downloads/diagslave/x86_64-linux-gnu$ su  
Si: sudo ./diagslave -m tcp  
u: diagslave 3.4 - FieldTalk(tm) Modbus(R) Diagnostic Slave Simulator  
o: Copyright (c) 2002-2021 proconX Pty Ltd  
S: Visit https://www.modbusdriver.com for Modbus libraries and tools.  
S:  
Pi: Protocol configuration: MODBUS/TCP  
Pi: Slave configuration: address = -1, master activity t/o = 3.00s  
Pi: IP configuration: port = 502, connection t/o = 60.00s  
Pi:  
Pi: Server started up successfully.  
s: Listening to network (Ctrl-C to stop)  
C: .....  
   Downloading nmap-0.0.1-py3-none-any.whl (2.7 kB)  
Installing collected packages: nmap  
Successfully installed nmap-0.0.1  
sanjana@sanjana-VirtualBox: ~/Downloads$ pip3 install sockets  
Collecting sockets  
   Downloading sockets-1.0.0-py3-none-any.whl (4.5 kB)  
Installing collected packages: sockets
```

This is the screen shot of the slave linux virtual machine with the IP address of 10.0.2.15 which is connecting with the ip address of

```

.....C:\
sanjana@sanjana-VirtualBox:~/Downloads/diagslave/x86_64-linux-gnu$ su
do ./diagslave -m tcp
[sudo] password for sanjana:
diagslave 3.4 - FieldTalk(tm) Modbus(R) Diagnostic Slave Simulator
Copyright (c) 2002-2021 proconX Pty Ltd
Visit https://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration: address = -1, master activity t/o = 3.00s
IP configuration: port = 502, connection t/o = 60.00s

Server started up successfully.
Listening to network (Ctrl-C to stop)
.
validateMasterIpAddr: accepting connection from 10.0.2.15
Slave 1: readHoldingRegisters from 100, 5 references
Slave 1: readHoldingRegisters from 100, 5 references
Slave 1: readHoldingRegisters from 100, 5 references
Slave 1: readHoldingRegisters from 100, 5 references
Slave 1: readHoldingRegisters from 100, 5 references
Slave 1: readHoldingRegisters from 100, 5 references

```

```

sanjana@sanjana-VirtualBox: ~/Downloads/modpoll/x8...
valCan't reach server/slave! Check TCP/IP and firewall settings.
sanjana@sanjana-VirtualBox:~/Downloads/modpoll/x86_64-linux-gnu$ sudo ./mo
Sladpoll -c 5 -r 100 -m tcp 10.0.2.15
Slamodpoll 3.10 - FieldTalk(tm) Modbus(R) Master Simulator
Slacopyright (c) 2002-2021 proconX Pty Ltd
Slavisit https://www.modbusdriver.com for Modbus libraries and tools.
Slaprotocol configuration: MODBUS/TCP, FC3
Slaslave configuration...: address = 1, start reference = 100, count = 5
1Slacommunication.....: 10.0.2.15, port 502, t/o 1.00 s, poll rate 1000 ms
1Sladata type.....: 16-bit register, output (holding) register table
1Slapolling slave... (Ctrl-C to stop)
1Sl[100]: 0
Sl[101]: 0
Sl[102]: 0
Sl[103]: 0
Sl[104]: 0
Slapolling slave... (Ctrl-C to stop)
Sl[100]: 0
.^[101]: 0
sar[102]: 0
[103]: 0
[104]: 0
-- Polling slave... (Ctrl-C to stop)
[100]: 0
[101]: 0

```

These are the screen shots for the device master and slave modbus simulator is connected to each other and leasing to the messages sent to them.

Present results on the performance of your considered algorithm, code, etc.

Input: The list of the detection range, $list$;

Output: The list of ICS devices, $list'$;

```
1: Using a random algorithm to resort the  $list^r = list$ ;  
2: for (each IP in  $list^r$ ) do  
3:   send one packet  
4:   each packet with stateless  
5:   add each live host into  $list'$   
6: end for  
7: for (each IP in  $list'$ ) do  
8:   using ICS protocols verifies it  
9:   add the quantified host into  $list'$   
10: end for  
11: for (each IP in  $list'$ ) do  
12:   if ( $p(y_i|X) > S_{th}$ ) then  
13:     send packet with packets  $FF$  with Algorithm 1.  
14:     if (get its responses & match the fingerprint) then  
15:       add it into  $list_{honeypots}$ , remove it from  $list'$   
16:     end if  
17:   end if  
18: end for
```

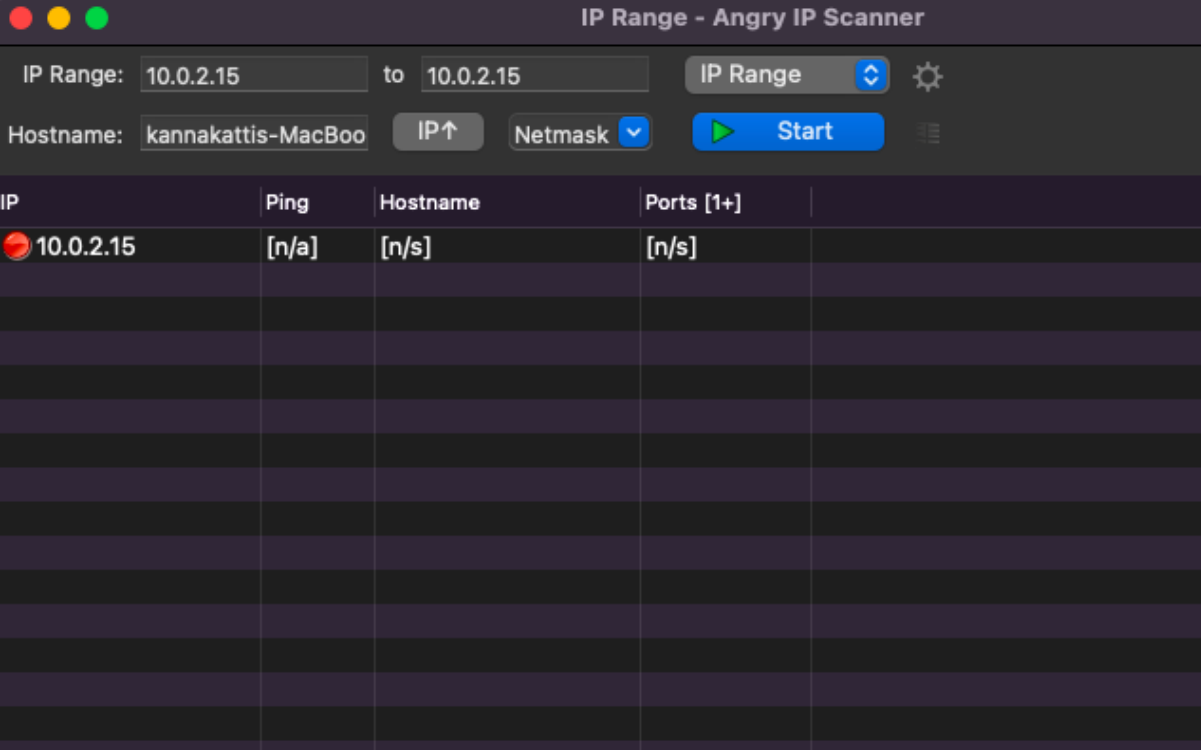
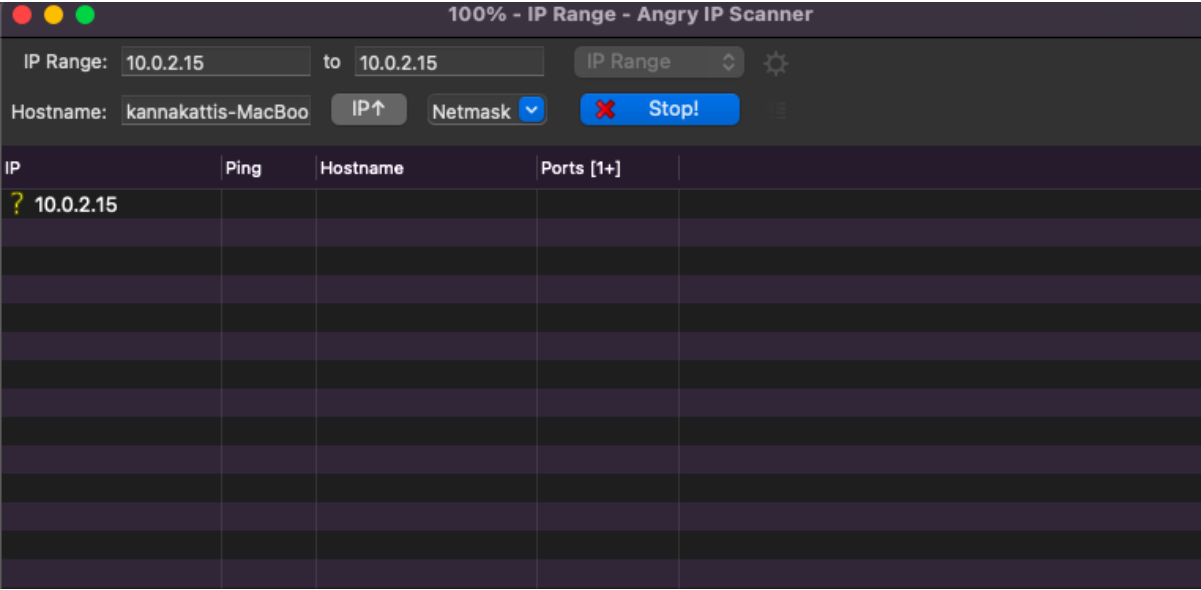
The algorithm is used to scan the active devices which is used in the paper.

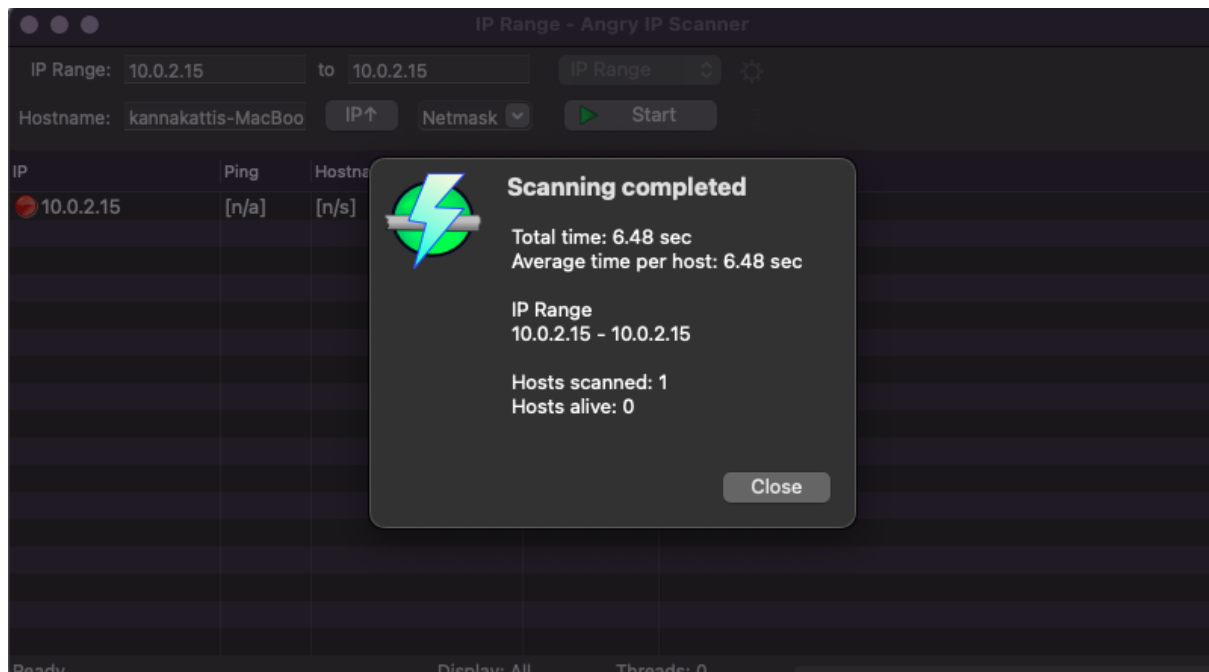


```
1 import nmap
2 import sockets
3 np = nmap.portScanner()
4 scan_range = np.scan(hosts="10.0.2.15/502")
5 list1 = scan_range['scan']
6 for i in range(list1):
7     s = socket.socket.AF_INET, socket.SOCK_STREAM)
8     s.connection(list1, 502)
9     s.send(Message)
10    data = s.recv()
11    if(data!=null):
12        list1[1] = list2[]
13    s.close()
14 print(list2)
```

```
sanjana@sanjana-VirtualBox:~/Downloads$ rm scanning_script.py
sanjana@sanjana-VirtualBox:~/Downloads$ ls
  acess-control.pdf      ModbusPal.jar          modpoll-3.10.tgz
  diagslave              modpoll                rxtx-2.1-7-bins-r2.zip
  diagslave-3.4.tgz      'modpoll-3.10(1).tgz' scanning.py
sanjana@sanjana-VirtualBox:~/Downloads$ vi scanning_script.py
sanjana@sanjana-VirtualBox:~/Downloads$ python3 scanning_script.py
10.0.2.15
sanjana@sanjana-VirtualBox:~/Downloads$
```

And also I was using the scanning app which can scan for the active devices by pinging the devices and it will check if the device is active or not and also there is a option to check if the device is using the certain protocol or not if we specify the port number there.





- Analysis of results

- Provide a brief justification of why you think you are obtaining the observed results.

The result I am getting is the list of the devices online which are using the modbus protocols. The Paper has analysed the 17 industrial protocols which are commonly used by the ICS devices. I have implemented one of the TCP protocols in which it needs to establish the three way handshake to connect with the devices and then send some message to it. So that the paper is using the same approach. And then the data which is the list of the devices and the some sort of the basic information based on the protocols they are using is used to analyse the functions and usage of the ICS devices online so that by understanding those functionality it could be more easy to protect them.

- Explain what are the real-world implications of your result.

The paper is trying to understand the usage of the ICS devices by their functions, features and their usage. So it is scanning the IPV4 space to collect the ICS devices data and also it is deploying the code in the EC2 instance to run the code for a long time and get the list of devices which are online in the whole IPv4 space.

- If possible, make a head to head comparison between your results and your chosen paper's Results

My result and the Papers result is similar as they are getting the list of the devices which are using the 17 ICS protocols But i am getting the devices which are just using the modbus TCP, one of the 17 ICS protocols. Then they are analysing that data based on the function, feature and the usage according to the region and the area.

In the paper they are scanning the IPV4 space and they are sending the messages to the ICS devices and getting the response from the active devices and then analysing the data based on the collected data. I am using the simulation which is using the one of the same protocol and getting the response back by pinging that using some scanning tool for that and list those devices in the format which are using the particular modbus protocol.

- Do your results match the paper's results? Explain why or why not.

The results don't somewhat match with the results of the paper as they are doing it live online as I am using the simulation to perform the task which is providing the different data with that of the actual data. In the paper while scanning for the devices they informed the system administrators that they are scanning their devices for the research purpose and they will be just getting the null response back to their system to know if the device is alive or not. And they even deployed that in the EC2 to run the code for the 3 months to scan the IPV4 space. So I had the simulation to perform that task and I did it for the single protocol. As that is the smallest part of the paper which will match with my results.

**** Describe what novel knowledge you obtained during this project.**

The paper's main aim was to understand the usage of online ICS devices which have an important role. And also the biggest task was to perform the scanning of the devices without intruding on the functionality of the working of those devices and also keeping track of those devices when they change their IP addresses. In the paper they performed those tasks well. And also I was able to get the importance of the simulation and protocols they are using in the particular ports while doing this project and also the working of the ICS honey pots they are used to monitor the behaviour of the attackers online.