University of Amsterdam

Research Project 1

Real time asset inventory in ICS

February 6, 2021

Student:
Artemis Mytilinaios
amytilinaios@os3.nl

Abstract

In ICS environments, it is important for organizations in multiple industries such as Oil and Gas, Chemicals, Transportation, and Consumer Goods to maintain the normal function of the ICS assets during the identification process. Setting up and maintaining a real-time asset inventory in ICS is a method to obtain information related to the assets that exist in the infrastructure. The well-known used techniques are active and passive. Conducted researches have shown that the active approach may harm the targeted asset by making it unresponsive. On the other hand, the passive technique utilizes a safer approach by collecting the information of assets' traffic, without harming them. In this research, we focused on the investigation of the hybrid approach of scanning, which is the combination of active and passive techniques, to achieve the collection of the best possible information from the assets without interrupting their operation. In order to minimize the possibility of malfunctioning of the assets, we needed to investigate the behavior of passive and hybrid techniques separately, so we could trace and find in which circumstances the identification of an asset leads to malfunctioning. To achieve this, we used some accessible tools compatible with each scanning technique, which provided information in depth about the assets. Also, we tested the behavior of assets during passive and hybrid asset identification by creating a Proof of Concept, including a virtualized Programmable Logic Controller (PLC) and Human Machine Interface (HMI). Our results showed that devices in each experiment reacted positively without any malfunction. Also, the experiments proved that hybrid scanning offered more information regarding the devices compared to passive.

***Keywords*— ICS, PLC, HMI, Open PLC, Scada Br, active scanning, passive scanning, hybrid scanning**

Contents

1 Introduction

An industrial control system (ICS) can be described as a group of different types of devices that exist in an industrial environment. One of the purposes that these devices have, is the control of crucial industrial processes. Industrial control systems can be utilized in different industries such as chemical, pharmaceutical, pulp and paper, food, and beverage[1]. A hypothetical example is a network of an energy company that consists of over one hundred ICS devices, separated into multiple sub-networks. Personnel responsible for administrating, maintain, and monitoring the environment must keep updated documentation of the infrastructure. The documentation contains information about the characteristics of devices, their purpose in the infrastructure, and the inter-communication with other devices[2]. Maintaining an up-to-date inventory is one of the top priorities that operators must have in mind. To achieve this, the first step is to identify the assets in the network. There are two well-known techniques that can be used for identification, the active and the passive scanning. Active scanning sends packets with each device in the network, with the purpose of identifying their status and information that can be collected[3]. On the other hand, passive scanning identifies services and hosts by collecting and analyzing network traffic. However, both techniques come with some drawbacks. Active scanning can be more effective in terms of the information that is collected, but it can cause disruptions and break downs to the host or service[3]. Passive scanning on the other side has been proven that it cannot harm devices and services, but the information that is obtained is not enough[4]. Except for these two techniques, a combination of them can be used to provide effective information from ICS devices with safety. This combination is called hybrid scanning and this paper mostly focused on it.

1.1 Research Questions

The reason behind this research topic originates from the problems that ICS administrators are facing when the identification of ICS devices occurs. Some of the most common problems are the outdated network diagrams and the fragility of ICS components. In addition to that, each scanning method either if it is active or passive, has its own drawbacks, as it was mentioned before. Based on that, the scope of this research project is to combine both techniques and evaluated it compared with passive. In more detail, the main research question is the following:

What are the added benefits of hybrid scanning compared to passive?

To effectively answer the main research question, the following sub-questions need to be answered:

- *What are the problems that can occur by using hybrid scanning in ICS environments?*
- *How do certain types of ICS devices behave under hybrid scanning and what are the problems that may arise in relation to these specific devices?*

2 Related Work

In 2015, Adam Wedgbury et al. researched the problems that exist during the identification process in ICS environments. According to the authors, in Information Technology (IT) environments the most common way to discover an asset is by actively probing packets to targeted devices. However, in the ICS infrastructure this can cause a lot of problems. ICS devices are more fragile compared to IT devices. It is common for ICS devices to become unresponsive when they receive unsupported types of packets. In their research, the authors tried to identify what is the scale of the problem, available tools that exist, and how they can be approached without interrupting devices' availability[5].

In 2018, Mohammed Abdulrazzaq et al. did a research trying to define the asset identification in ICS, using already existing methods. Apart from the scope of identification, the authors proposed an alternative method, the hybrid scanning. Hybrid tries to eliminate the disadvantages of active scanning by combining it with passive scanning. It was also found that the benefits of their combination can lead to better and safer results. However, not enough research has been done so far[6].

In 2003, Sergei Bantseev et al. evaluated the available tools that exist for network scanning. They came up with different categories, but there was one that stood out, the active hybrid. At the time of the research, the available tools in this category were Fluke Optiview, Foundstone, Cheop-Ng, and Big sister. In their study, they concluded that there is not any available tool that can do it all[7].

3 Background

To carry out our research, we conducted several experiments using the knowledge of previous research. To get a better understanding of how our lab environment works, in this section we describe the appropriate tools that we used and their functionalities.

3.1 OpenPLC

A programmable logic controller (PLC) is a special type of computer that was invented to replace the sequential relay circuits for machine control[8]. PLCs can be used to handle instructions from users and perform their programmable actions[9]. This kind of device can be found in the industry section, but it is not unusual to be found in the IT environments too. OpenPLC is an open-source tool developed by Thiago Alves[10], aiming to emulate PLC programs in virtualized environments. OpenPLC has no cost for the consumer, making for everyone available the path to automation. During the creation of the tool, the developer followed the architecture's logic of the already existed PLCs. This means that the prototype of OpenPLC includes a Bus board, CPU card, Input, and Output card[10]. Finally, someone can use the Ladder Logic (LD), the fundamental programming language of PLC to develop a simple program. OpenPLC can be programmed by using any of the following four languages, which can be combined to develop a complete program[10].

- Instruction List (IL)
- Function Block Diagram (FBD)
- Sequential Function Chart (SFC)
- Structured Text (ST)

3.2 Scada Br

Scada Br is an open-source tool aiming to the development of automation, data acquisition, and human machine interfaces (HMI). This tool is a web browser platform allowing a user to interact with the monitoring, control, and automation of ICS equipment over different protocols, such as Modbus[11]. This tool is an ideal solution for developing communication channels between multiple ICS devices, creating HMIs, and developing automated processes. Scada Br is intended to be used by universities, automation professionals, technical schools, and companies[12]. Scada Br is available in a virtualized platform using any kind of available hypervisors.

3.3 Active and Passive scanning tools

Plcscan
Plcscan is an open-source scanning program, developed by Dmitry Efanov[13] using Python. It aims to discover PLCs by scanning the Modbus/TCP protocol[13].

ScadaScan Scadascan is a script written in Perl programming language, aiming to the identification of Modbus slaves and Distributed Network Protocol 3 (DNP3) slaves. It works by brute-forcing the ID field in the messages in order to discover the slave devices[14].

Nmap
Nmap or in other words Network Mapper is the most well-known open-source tool for network scanning and security auditing[15]. Nmap also includes the Nmap Scripting Engine (NSE) where a user can execute his own developed scripts. Professionals with expertise in ICS environments developed NSE scripts targeting ICS devices in order to enumerate and discover them. Some examples of these scripts are:

- Modbus-discover
- Modicon-info
- Dnp3-info
- Enip-enumerate

GRASSMARLIN
GRASSMARLIN is an open-source tool that passively discovers devices by sniffing network traffic. It provides a piece of detailed information about the devices that communicate with each other and the protocol they use[16].

3.4 VMware Workstation

VMware Workstation
VMware workstation is a type-2 hypervisor that runs directly in the software instead of the hardware like type-1 hypervisors do[17]. This tool offers the flexibility of deploying different operating systems separately in a safe and stable environment[18]. This tool was used as the base of the proof of concept topology, which is explained in the next chapter.

4 Methodology

4.1 Approach

This section outlines our methodology during the research. To identify the benefits of hybrid scanning, we evaluated the problems that may occur by using this technique and the behavior of the devices that existed in the ICS environment. We assessed any kind of anomaly that was produced by the scanning technique. We concluded our results by testing the functionality of each device at the end of each experiment.
With the help of a Proof of Concept, we were able to retrieve our results using a virtualized environment. Combining the following tools, we achieved a testbed of ICS infrastructure that could provide us with the expected results.

- OpenPLC Runtime
- Scada Br (HMI)
- VMware Workstation
- Kali Linux

- ICS active/passive scanning tools

The setup was built using VMware Workstation as hypervisor where the OpenPLC and Scada Br (HMI) were deployed and configured accordingly. OpenPLC was configured using a simple program that contains one rung, a button, a timer, and a coil. The main function of the program started after pressing the button. The coil was activated and stayed active for 2 seconds[19]. The Linux distribution, Kali, was used as the main platform for hosting the scanning tools. These programs could be divided into two groups, the active and the passive.
The whole setup is depicted in Figure 1.

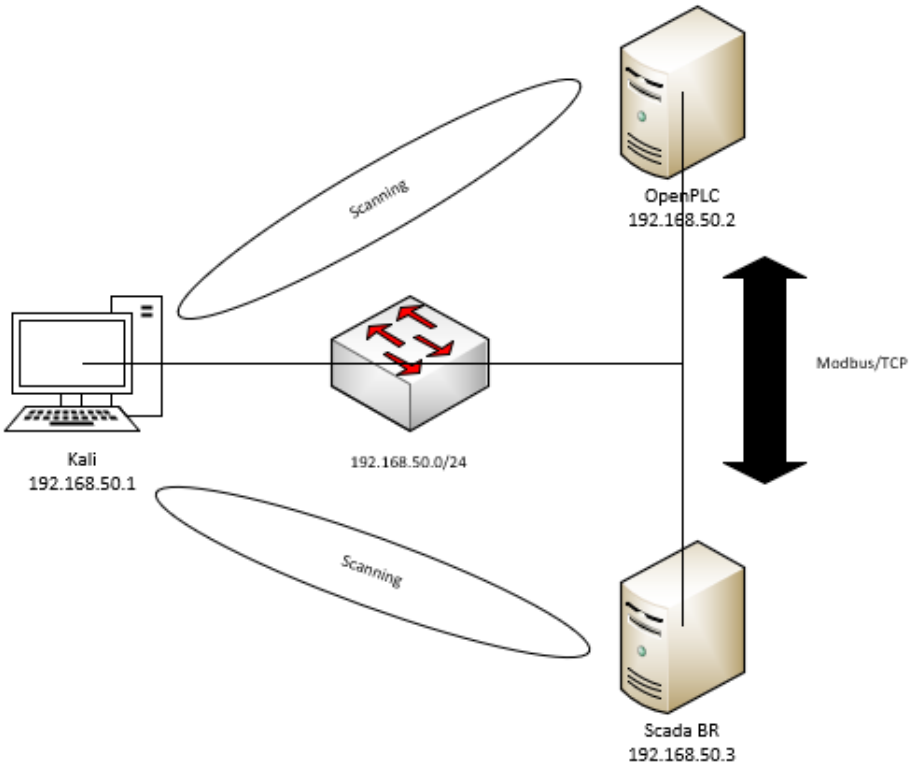

Figure 1: Proof of Concept.

4.2 Scope

The scope of this project is to understand the benefits, alongside the problems that may occur during a hybrid scanning approach in ICS environments. In order to accomplish this, a series of experiments had to be conducted for generating results regarding the behavior of the ICS devices. By getting appropriate results regarding the functionality of the devices, we were able to identify after the analysis what and when caused the malfunctions to the devices.
Unfortunately, due to the COVID-19, we could not test our approach on physical ICS devices. Consequently, the scope of this research focused on the virtualized ICS environments and literature research that already has been done.

5 Results

In this section, we describe the results of our experiments. To have a better understanding of the outcome, we present the results of the passive tool, as well as in combination with each active tool.

5.1 Passive

As it was mentioned before, we used a proof of concept environment. We monitored the communication channels between the OpenPLC and Scada Br (HMI), using the tool Wireshark. Also, we monitored the channels between the Kali and OpenPLC, Kali and Scada Br (HMI). As it was mentioned previously, the passive tool that was used was the GRASSMARLIN. The tool analyzed the traffic that was stored in a captured file (.pcap) and created a logical representation of the network. It also identified the protocols that were used during the communication of the active nodes. Figure 2 depicts a successful run of the tool.

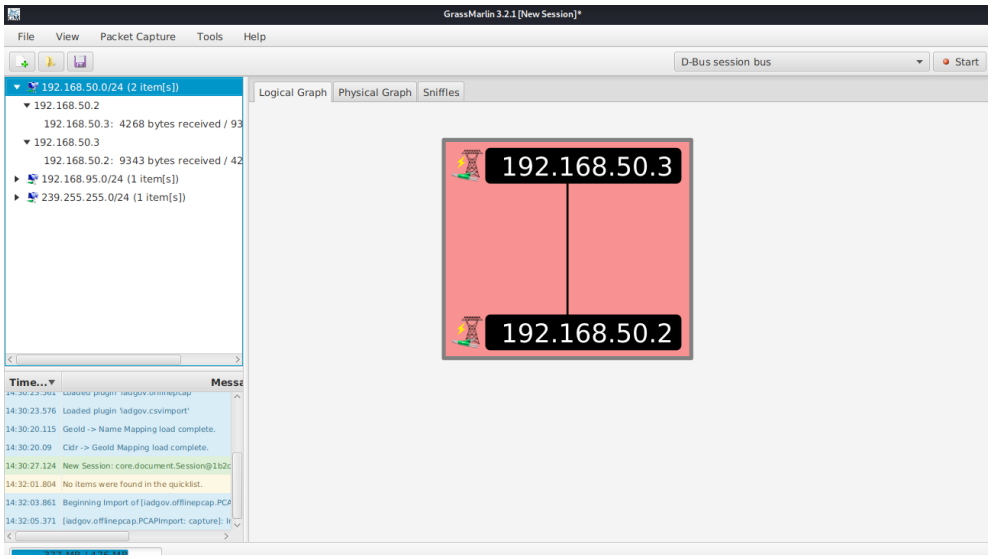

Figure 2: Successful run of GRASSMARLIN.

The results showed two nodes, the OpenPLC with an IP address of 192.168.50.2 and the Scada Br (HMI) with an IP address of 192.168.50.3. Figures 3 and 4 depict the collected information of OpenPLC and Scada Br (HMI).

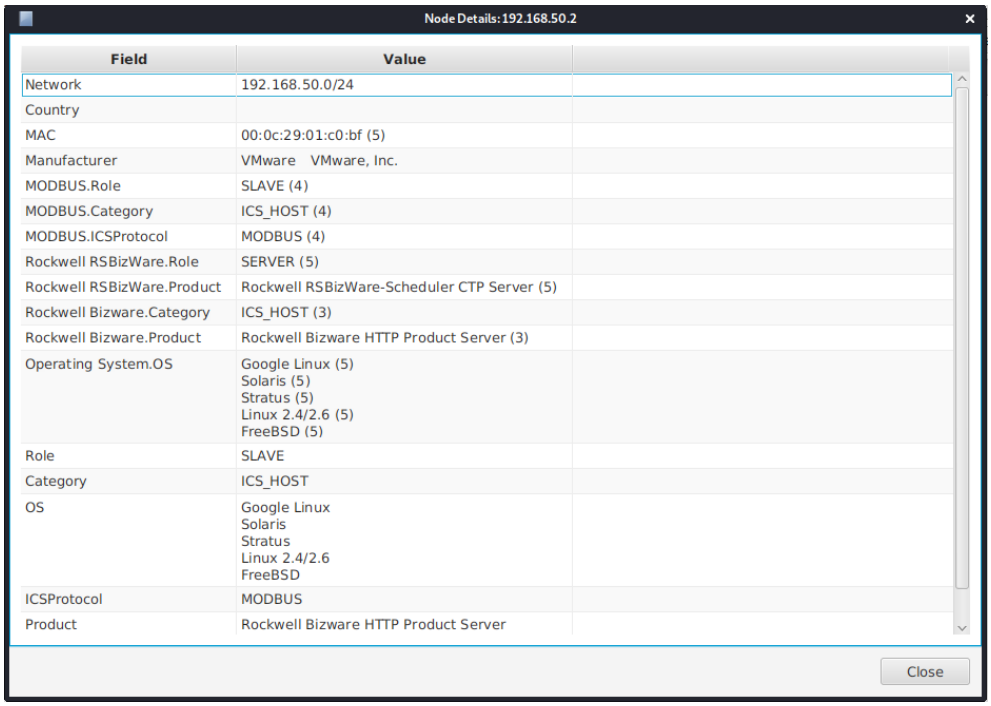

Figure 3: Collected information from OpenPLC.

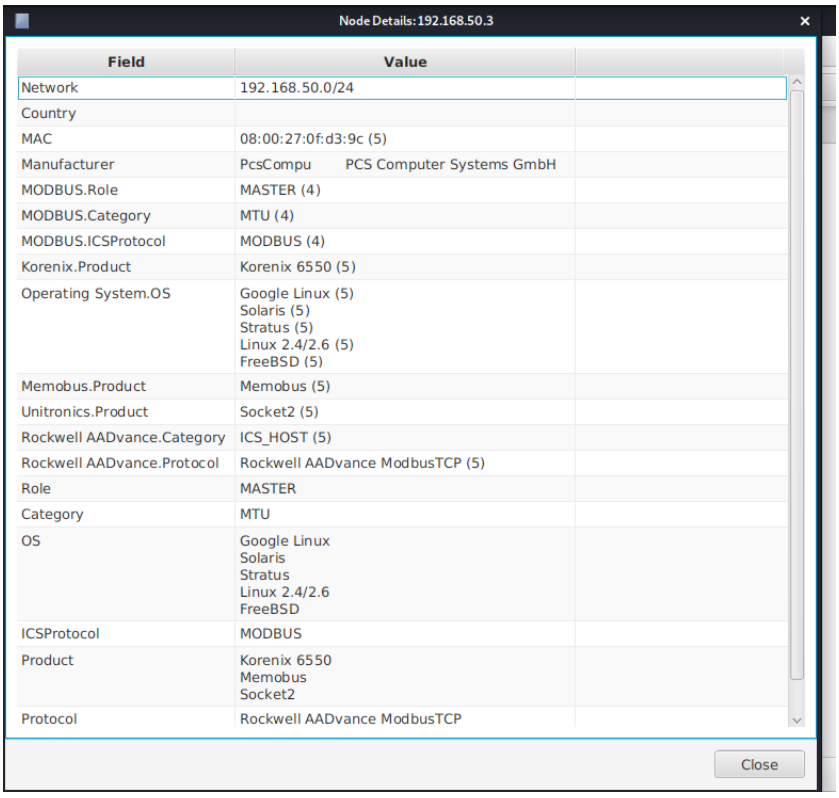

Figure 4: Collected information from Scada Br (HMI).

From the conducted experiments, we concluded that the information collected, provided details regarding the manufacturer, the role of the device, the used protocol, and the operating system of the device. The tool automatically categorized the devices as ICS_HOST and at the same time detected the Modbus protocol that was used for their communication.

5.2 Hybrid

With the combination of the passive tool with an active one, we achieved a hybrid scanning approach.
Hybrid Plcscan
Combining the script Plcscan with the passive tool resulted in a successful scanning without any indication of service interruption. The approach was tested on both devices. The information we retrieved was about the open port number and the unit id of the device, as well as all the passive information. However, the Scada Br (HMI) device did not respond to the scan, because the targeted port was not open. The following commands were executed:

plcscan –ports=502 192.168.50.2
plcscan –ports=502 192.168.50.3

The following Figure illustrates the results of the scan.

```
┌──(root㉿ kali-isc)-[/home/artemis]
└─# plcscan --ports=502 192.168.50.2 && plcscan --port=502 192.168.50.3
Scan start ...
192.168.50.2:502 Modbus/TCP
  Unit ID: 0
    Device info error: ILLEGAL FUNCTION
  Unit ID: 255
    Device info error: ILLEGAL FUNCTION
Scan complete
Scan start ...
Scan complete
```

Figure 5: Hybrid Plcscan results.

PLCscan used TCP three-way handshake to establish the connection from a client (192.168.50.1) to a server (192.168.50.2) at port 502. Then the client sent a Modbus request packet, including the Function code of 43. The code was used for the identification of the targeted device[20]. Figure 6 presents the request packet.

```
▾ Modbus/TCP
    Transaction Identifier: 0
    Protocol Identifier: 0
    Length: 5
    Unit Identifier: 0
▾ Modbus
    .010 1011 = Function Code: Encapsulated Interface Transport (43)
    MEI type: Read Device Identification (14)
    Read Device ID: Basic Device Identification (1)
    Object ID: VendorName (0)
```

Figure 6: PLCscan Modbus request packet.

The device replied with a Modbus respond packet that contained an exception of Illegal Function. This means that the targeted device did not support this kind of function, as shown in Figure 7.

```
Modbus/TCP
    Transaction Identifier: 0
    Protocol Identifier: 0
    Length: 3
    Unit Identifier: 255
Function 43:  Encapsulated Interface Transport.  Exception: Illegal function
    .010 1011 = Function Code: Encapsulated Interface Transport (43)
    Exception Code: Illegal function (1)
```

Figure 7: PLCscan Modbus reply packet.

Hybrid ScadaScan
The next tool that was tested was the ScadaScan.pl, written in Perl programming language, aiming to for the active identification of Modbus slaves and Distributed Network Protocol 3 (DNP3) slaves. It worked by brute-forcing the ID field in the messages in order to discover the slave. The following command was executed for the scanning.

perl scadascan.pl -d 192.168.50.0/30

The results proved that the targeted devices were not affected by hybrid scanning, as shown in Figure 8

```
┌──(root kali-isc)-[/home/artemis/Desktop]
└─# perl scadascan.pl -d 192.168.50.0/30

Working on 192.168.50.0
Processing DNP ...
DNP not running
Working on 192.168.50.1
Processing DNP ...
DNP not running
Working on 192.168.50.2
Processing DNP ...
DNP3 running
Working on 192.168.50.3
Processing DNP ...
DNP not running
```

Figure 8: Hybrid Scadascan resutls.

A client sent a DNP3 request packet to the targeted device that contained a Request link status. The client issued this request with the self-address of 0xFFFC. Afterwards, it tried to identify the slave to be associated with this connection. Consequently, only DNP3 slave devices responded to this kind of request with self-address[21].

Figure 9 depicts a DNP3 request to the OpenPLC.

```
> Frame 5: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
> Ethernet II, Src: VMware_be:fa:d9 (00:0c:29:be:fa:d9), Dst: VMware_01:c0:bf (00:0c:29:01:c0:bf)
> Internet Protocol Version 4, Src: 192.168.50.1, Dst: 192.168.50.2
> Transmission Control Protocol, Src Port: 56220, Dst Port: 20000, Seq: 1, Ack: 1, Len: 10
v Distributed Network Protocol 3.0
  v Data Link Layer, Len: 5, From: 5, To: 65521, DIR, PRM, Request Link Status
      Start Bytes: 0x0564
      Length: 5
    v Control: 0xc9 (DIR, PRM, Request Link Status)
        1... .... = Direction: Set
        .1.. .... = Primary: Set
        ..0. .... = Frame Count Bit: Not set
        ...0 .... = Frame Count Valid: Not set
        .... 1001 = Control Function Code: Request Link Status (9)
      Destination: 65521
      Source: 5
      Data Link Header checksum: 0xd2aa [correct]
      [Data Link Header Checksum Status: Good]
```

Figure 9: Scadascan DNP3 request.

Hybrid Nmap
The modbus-discover script tried to enumerate Modbus device information. More specifically, it tried not only to scan legal slave IDs from Modbus devices, but also to export information regarding the vendor and the firmware of the device. By conducting the experiments, the results revealed that the targeted devices continued their operation normally after the scanning. Unfortunately, the Scada Br (HMI) did not respond to the particular scan. Bellow are presented the executed commands of the experiment.

nmap –script Modbus-discover.nse -p 502 192.168.50.2
nmap –script Modbus-discover.nse -p 502 192.168.50.3

Figure 10 shows the results of the scan.

```
┌──(root㉿ kali-isc)-[/home/artemis/Desktop/nmap-scripts]
└─# nmap --script modbus-discover.nse -p 502 192.168.50.2
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-05 17:54 CET
Nmap scan report for 192.168.50.2
Host is up (0.00041s latency).

PORT    STATE SERVICE
502/tcp open  modbus
| modbus-discover:
|   sid 0×1:
|_    error: ILLEGAL FUNCTION
MAC Address: 00:0C:29:01:C0:BF (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.49 seconds

┌──(root㉿ kali-isc)-[/home/artemis/Desktop/nmap-scripts]
└─# nmap --script modbus-discover.nse -p 502 192.168.50.3
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-05 17:54 CET
Nmap scan report for 192.168.50.3
Host is up (0.00059s latency).

PORT    STATE  SERVICE
502/tcp closed mbap
MAC Address: 08:00:27:0F:D3:9C (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.43 seconds
```

Figure 10: Hybrid Nmap Modbus-discover results.

Modbus-scan used the Modbus request packets by requesting the slave ID from a device. Figure 11 depicts the request packet

```
> Frame 14: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
> Ethernet II, Src: VMware_be:fa:d9 (00:0c:29:be:fa:d9), Dst: VMware_01:c0:bf (00:0c:29:01:c0:bf)
> Internet Protocol Version 4, Src: 192.168.50.1, Dst: 192.168.50.2
> Transmission Control Protocol, Src Port: 51658, Dst Port: 502, Seq: 1, Ack: 1, Len: 8
v Modbus/TCP
    Transaction Identifier: 0
    Protocol Identifier: 0
    Length: 2
    Unit Identifier: 1
v Modbus
    .001 0001 = Function Code: Report Slave ID (17)
```

Figure 11: Modbus-discover request.

The hex value 0x11 corresponded with number 17. The Modbus function code 17 was a diagnostic function to Report Slave ID. The reply that came from the targeted device indicated that the requested function was not supported (Figure 12).

```
> Frame 16: 75 bytes on wire (600 bits), 75 bytes captured (600 bits)
> Ethernet II, Src: VMware_01:c0:bf (00:0c:29:01:c0:bf), Dst: VMware_be:fa:d9 (00:0c:29:be:fa:d9)
> Internet Protocol Version 4, Src: 192.168.50.2, Dst: 192.168.50.1
> Transmission Control Protocol, Src Port: 502, Dst Port: 51658, Seq: 1, Ack: 9, Len: 9
v Modbus/TCP
    Transaction Identifier: 0
    Protocol Identifier: 0
    Length: 3
    Unit Identifier: 1
v Function 17:  Report Slave ID.  Exception: Illegal function
    .001 0001 = Function Code: Report Slave ID (17)
    Exception Code: Illegal function (1)
```

Figure 12: Modbus-discover reply.

To sum up, after the analysis of the gathered information, we came up with the following results. Firstly, both devices remained stable and available throughout the whole process of the three hybrid scanning experiments. Each experiment was conducted multiple times to make sure that we received valid results. As it is illustrated in the following graph (Figure 13), in each case both devices had 100% availability.

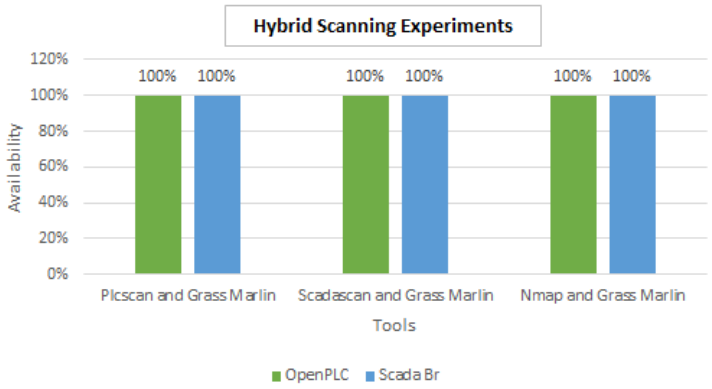

Figure 13: Hybrid scanning Graph

Also, the results indicated a difference in the provided information between each type of scanning that was used. To be more specific, passive scanning provided information that could be collected by analyzing network traffic. On the other hand, through the hybrid scanning (including active scanning tools), we gathered additional information in each experiment. One of the differences that these two techniques had, was that passive collected

less information than hybrid scanning. On the other side, none of the techniques made the devices unresponsive.

6 Discussion

In this chapter we discuss the results and the limitations of this research project. In our passive experiment we collected information regarding the devices. We expected that the information would be limited due to the nature of passive scanning. On the other hand, in our hybrid experiments we collected information which depended on the active tool that was used, combined with the information retrieved from the passive tool. Overall, our results demonstrate that there was a difference of the provided information between hybrid and passive scanning. In addition, in this experiment we obtained extra information regarding the operation of the devices. It came out that the hybrid scanning approach did not arise any fragility on the targeted devices.
Besides, we should take into consideration that the environment that the experiments were conducted in, was virtualized due to the inaccessibility to physical devices. As a result, if the same experiments were conducted in a different environment, the results might differ.

7 Conclusion

In this research, the following research question was answered *What are the added benefits of hybrid scanning compared to passive?*. In order to answer our research question, we firstly needed to answer the following sub-questions *What are the problems that can occur by using hybrid scanning in ICS environments?* and *How certain types of ICS devices behave under hybrid scanning and what are the problems that may arise in relation to these specific devices?* To answer our sub-questions, we performed experiments in order to get the necessary results for answering the sub-questions.
Regarding the first sub-question, the results showed that no problems arose regarding the performance, availability, and responsiveness of the devices by using hybrid scanning. About the last sub-question, OpenPLC and Scada Br remained stable without any interruption during the experiments. Concluding, the answer to our main research question based on the experiments is that hybrid scanning is superior according to the provided information, compared to passive scanning. Moreover, through hybrid scanning, a variety of tools can be chosen, which gives a sense of flexibility. This depends on the characteristics of the targeted devices.

7.1 Future Work

As part of future work, further investigation of this technique is needed. The investigation should be focused on the physical PLCs and HMIs. Only some devices could be examined in this research project because the devices' options to examine are unlimited. For this reason, further experiments using different PLC's vendors should be conducted. Furthermore, the percentage of service's availability should also be examined. The next step could be to conduct the experiments on a live ICS environment and document the results. Further work should also be conducted by using different offered tools from both passive and active groups. There is a variety of options that are available, which can influence the results of this approach. Another topic for future investigation is the vendor tools. Taking as an example the Totally Integrated Automation (TIA) tool[22] that Siemens provides, deeper research should be conducted to identify the techniques that these kinds of tools are using. By identifying the techniques used by these tools, someone can say that it will be beneficial to integrate them into the hybrid approach. This could result in a new era of scanning tools for ICS environments.

8 Acknowledgments

We would like to thank our supervisors Michel van Veen and Pavlos Lontorfos from Deloitte for their supervision and guidance for this research project.

Acronyms

DNP3 Distributed network protocol 3.

HMI Human machine interfaces.

ICS Industrial control system.

IT Information technology.

LD Ladder logic.

NSE Nmap scripting engine.

PLC Programming logic controller.

TCP Transmission control protocol.

References

[1] K. Stouffer, J. Falco, and K. Scarfone, "Guide to industrial control systems (ics) security," *NIST special publication*, vol. 800, no. 82, pp. 16–16, 2011.

[2] C. Mavrakis, "Passive asset discovery and operating system fingerprinting in industrial control system networks," *Wayback archive: http://web. archive. org/web/20190307110951/https://pure. tue. nl/ws/files/46916656/840171-1. pdf*, pp. 840171–1, 2015.

[3] S. Webster, R. Lippmann, and M. Zissman, "Experience using active and passive mapping for network situational awareness," in *Fifth IEEE International Symposium on Network Computing and Applications (NCA'06)*, pp. 19–26, IEEE, 2006.

[4] G. Bartlett, J. Heidemann, and C. Papadopoulos, "Understanding passive and active service discovery," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pp. 57–70, 2007.

[5] A. Wedgbury and K. Jones, "Automated asset discovery in industrial control systems-exploring the problem," in *3rd International Symposium for ICS & SCADA Cyber Security Research 2015 (ICS-CSR 2015) 3*, pp. 73–83, 2015.

[6] M. Abdulrazzaq and Y. Wei, "Industrial control system (ics) network asset identification and risk management," 2018.

[7] S. Bantseev and I. Labbé, "Study of tools for network discovery and network mapping," tech. rep., COMMUNICATIONS RESEARCH CENTRE OTTAWA (ONTARIO), 2003.

[8] W. Bolton, *Programmable logic controllers*. Newnes, 2015.

[9] B. T. Frederick, P. A. Hickok, and D. S. Kougel, "Method for programming a programmable logic controller," Oct. 3 2006. US Patent 7,117,043.

[10] T. R. Alves, M. Buratto, F. M. De Souza, and T. V. Rodrigues, "Openplc: An open source alternative to automation," in *IEEE Global Humanitarian Technology Conference (GHTC 2014)*, pp. 585–589, IEEE, 2014.

[11] M. S. Almas, L. Vanfretti, S. Løvlund, and J. O. Gjerde, "Open source scada implementation and pmu integration for power system monitoring and control applications," in *2014 IEEE PES General Meeting — Conference Exposition*, pp. 1–5, 2014.

[12] "Scadabr."

[13] M. S. Javate, "Study of adversarial and defensive components in an experimental machinery control systems laboratory environment," tech. rep., NAVAL POSTGRADUATE SCHOOL MONTEREY CA, 2014.

[14] K. C. Wiberg, "Identifying supervisory control and data acquisition (scada) systems on a network via remote reconnaissance," tech. rep., NAVAL POSTGRADUATE SCHOOL MONTEREY CA, 2006.

[15] P. Calderon, *Nmap: network exploration and security auditing cookbook : a complete guide to mastering Nmap and its scripting engine, covering practical tasks for penetration testers and system administrators, 2nd edition*. PACKT Publishing, 2nd ed. ed., 2017.

[16] N. Upanavage and P. Orr, "Utilizing virtual network taps to increase visibility into virtualized control system networks,"

[17] S. Simic, "How to implement validation for restful services with spring," Dec 2020.

[18] D. T. Vojnak, B. S. orđević, V. V. Timčenko, and S. M. Štrbac, "Performance comparison of the type-2 hypervisor virtualbox and vmware workstation," in *2019 27th Telecommunications Forum (TELFOR)*, pp. 1–4, IEEE, 2019.

[19] T. Alves, "The openplc project," 2018.

[20] D. J. W. Dube, "Modbus encapsulated transport interface," Aug. 11 2009. US Patent 7,574,512.

[21] "Dnp3 opc server configuration guide."

[22] C. A. Bejan, M. Iacob, and G.-D. Andreescu, "Scada automation system laboratory, elements and applications," in *2009 7th international symposium on intelligent systems and informatics*, pp. 181–186, IEEE, 2009.