

A Secure Network Scanner Architecture for Asset Management in Strongly Segmented ICS Networks

Matthias Niedermaier^a, Thomas Hanka, Florian Fischer^b and Dominik Merli
HSA_innos, Hochschule Augsburg University of Applied Sciences, Germany

Keywords: Network, Security, Scanning, ICS, SCADA.

Abstract: Industrial Control System (ICS) are essential for process automation and control in critical infrastructures, like smart grids, water distribution and also food production, in our modern world. These industrial devices will be even more connected, due to the trend of Industry 4.0 and Internet of Things (IoT), to provide additional functionality. An example for a use case is predictive maintenance, where sensor data is required, to e.g. replace defective parts before outage. While connectivity enables easier and more efficient process management, it also increases the attack surface for cyber-attacks. To provide secure operation for interconnected ICSs additional protection measures, like asset management should be applied, to observe and maintain assets within a control network. One of the first steps to improve cyber-security with asset management is device identification in ICS networks. A common method for device identification is active network scanning, which adds additional network traffic to the ICS network. Because of the common segmentation with firewalls of ICS networks, scanner nodes in each sub-network are necessary. The distribution of active scan nodes typically adds additional cross connections within segmented ICS networks. In this paper, we introduce a secure scanning architecture for fragile ICS networks. Our architecture is based on scanning nodes, which use the concept of hardware-based data diodes to e.g. separate the critical control network from the office network. To ensure a gentle scan on fragile ICS networks, the scan node provide a bandwidth limitation of the scan, to reduce risk of influences within ICS networks. We implemented a Proof of Concept (PoC) system and evaluated it within our industrial testbed, to show the feasibility of our architecture.

1 INTRODUCTION


Industrial Control Systems (ICSs) control our daily life, unnoticed by the majority of the population. A typical field of application for ICS devices is process automation within building automation or production lines in the area of chemistry and automotive. But ICS devices are also applied to critical infrastructures, such as electricity or water supply, to automate and control critical tasks. In these environments, a disturbance of the regular execution may have severe consequences and e.g. could lead to a blackout.


Historically, ICSs were operating in isolated network and field-bus environments, due to lack of necessity for data exchange to the outside of the ICS. In contrast to this, in recent years, the connectivity of ICSs and distribution of information within industrial networks is increasing and will further increase

in the future, due to the trend of Industry 4.0. A use case, which relays on the networking ability and interconnection of ICS is predictive maintenance, where outages can be foreseen, based on the continuous observation of certain parameters. This data exchange between ICSs and data processing systems in modern networks is often Internet Protocol (IP)-based, which also allows an interaction with common Information Technology (IT) systems.

But next to the advantages of interconnected devices, this connectivity leads to an enlarging attack surface of ICS devices. The main reason for this is, that the network interfaces introduce the possibility of remote attacks on the Operational Technology (OT) environment. This leads to the point, that a proper security strategy must be applied to reduce the new possible attack vectors.

Important measures to increase the IT-Security are proper update and vulnerability management, to react on new vulnerabilities as fast as possible. In order to identify which versions of hardware and software

^a  <https://orcid.org/0000-0003-4550-7422>

^b  <https://orcid.org/0000-0002-4532-731X>

are currently in use, asset identification is essential to perform a proper asset management (Vanier, 2001). This information is required, to determine currently existing vulnerabilities in the products in use, or if e.g. security updates are available from the manufacturer.

There are various options for gathering the data necessary for asset management. Next to manual gathering with e.g. inventory lists, it is possible to access necessary information technically via passive or active network scanning. While with passive network scanning the data traffic is used, that already exists in the network, and with active scanning additional network traffic is brought into the network. This is also one of the main differences, which usually leads to a poorer detection rate for passive network scanning. One of the main reasons for the mostly lower scan result quality of passive scanning is, that e.g. no information about the software version used by the devices is transmitted in regular communication. Both technologies have their right to exist and, depending on the application, show their strengths. However, in this work, the focus is on active network and vulnerability scanning as a technology for asset management, because of the mostly better detection rate.

Due to the strong segmentation in industrial networks, active scanning is usually implemented with scanner nodes in each sub-network. These nodes often struggle the problem, that if they contain vulnerabilities themselves, additional attack vectors are created.

In this paper we present a secure scanning architecture for asset management within industrial networks with the following contributions:

- Safe and secure scanner architecture for ICSs, which ensure a slow and therefore gentle device scanning in fragile networks.
- ICS scanner node based on a bidirectional Intelligent Data Diode (IDD) with strong segmentation.
- Evaluation of the concept with a low-cost prototype, build with common off-the-shelf components.

The paper is structured as follows. In Section 2 the current state and challenges of asset management and industrial network scanning is explained. The methodology behind the here presented network scanner is introduced in Section 3. Afterwards, the implementation is described and evaluated in Section 4. At the end, in Section 5, a conclusion and outlook is given.

2 ASSET MANAGEMENT AND NETWORK SCANNING IN ICS NETWORKS

Asset management has various specifics and challenges in industrial networks. These are taken up and explained in the following.

2.1 Network Segmentation in ICS

First and one of the most important measures within a defense-in-depth strategy is the zoning or segmentation of assets within sub-networks. This segmentation is mostly done according to their application scenario or logical relationship to each other. This results commonly in sub-networks for IT assets and one or multiple sub-networks for OT assets. Between these sub-networks, there are often firewalls (Nivethan and Papa, 2016) placed between the IT, OT and other sub-networks. These are used to actively protect against attacks against the ICS network.

In addition to the obvious advantages of segmentation, it is often implemented because of recommendations or requirements in various guidelines and standards. For instance “NIST SP800-82 – Guide to ICS Security” (Stouffer et al., 2015) recommends the segmentation of control networks and corporate networks. In addition to that, the IEC 62443-3-3 (IEC, 2020) in particular “SR 5.1 Network Segmentation”, describes, that control system networks should be logically segmented from non-control system networks.

Figure 1 shows an example of a common network structure with a defense-in-depth strategy, as used in manufacturing industry (Kuipers and Fabro, 2006). In this example, the ICS network contains Programmable Logic Controllers (PLCs), which interacts with the physical world, a Supervisory Control and Data Acquisition (SCADA) system for monitoring and controlling the process, as well as a Human Machine Interface (HMI) for user interactions. In the corporate network, often Manufacturing Execution System (MES)/Enterprise Resource Planning (ERP) systems are located, which allows planning of the production. Furthermore, standard computer and other standard components like printers are also located in the corporate network. In this case, the corporate network is protected from the untrustworthy Internet with a firewall. In addition, the ICS network is furthermore protected by a firewall from the corporate network. If an external attacker ❶ wants to take over the control network, two firewalls must be compromised. An internal attacker or virus in the office network ❷ must also compromise the industrial firewall in order to get access to the ICS network. Only an

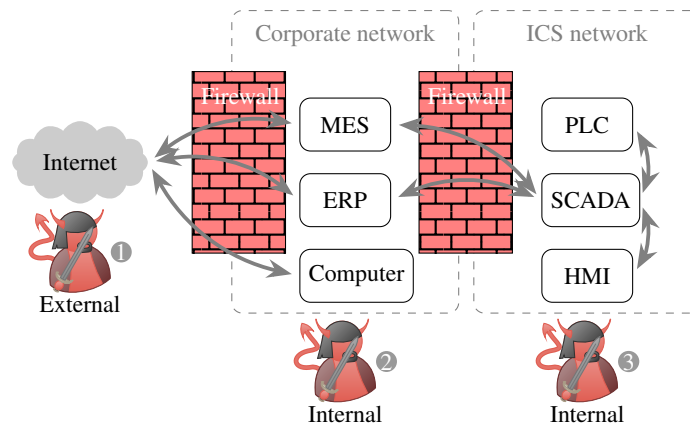


Figure 1: Example network architecture and data flow following the principal defense in the depth for segmented ICS infrastructures.

attacker who has already access to the ICS network ③ does not have to overcome perimeter protection.

2.2 Asset Management and Protection Strategies in ICS

For the subject of asset and vulnerability management there are similar techniques used, how the required information is gathered (Knapp and Langill, 2014). Beside manual gathering the information by hand, there exist two common strategies for identifying assets and related metadata for vulnerability detection within an industrial network environment.

The first variant is the complete passive approach, which analyzes the existing network traffic within an OT network, e.g. by listening to a mirror port. This approach causes no additional network traffic to the observed devices and the network. On the downside the detection potential is limited to the information, which can be extracted from the existing network traffic.

The second variant is based on actively scanning the assets within the network, which generates additional traffic within the OT network. The targeted and active information gathering increases the quality of results and therefore the detection accuracy. But the additional network traffic also can cause disturbance to the targeted assets (Niedermaier et al., 2018). Due to the fragility of common off-the-shelf industrial components, active scanning must be executed with caution and the amount of generated network traffic must be chosen with care. Thus it is important, to limit the scan rate and therefore the traffic generated by the scanning process to a predefined and uncritical threshold value.

Another security measure within the defense-in-depth approach is the inclusion of devices and ser-

vices to recognize ongoing attacks on the assets with the ability to react towards these attacks. To detect attacks in progress, Intrusion Detection Systems (IDSs) and network monitoring systems (Maynard et al., 2018) are commonly used, while Intrusion Prevention System (IPS) systems are used to prevent influences of attack attempts and preempting the attacker.

In addition to find devices, network scanners (Cofey et al., 2018) can also be used to uncover unnecessary open ports and services, as well as other misconfigurations. Furthermore, with network scanner it is also possible or detect changes of the network configuration, e.g. according to an attack. The use case of active network scanning for asset identification, vulnerability detection and asset management in general in ICS is where the work of this paper focuses.

2.3 Current Challenges of Active Network Scanning in ICS

The segmentation also means that active scanners that can be used for asset management must be able to interact with segmented sub-networks. There are therefore two common ways of realizing this for active scanners: on the one hand, the firewall can be set up so that it is open for scanners and, on the other hand, scan nodes can be placed in every sub-network. The use of scan nodes in the sub-networks is usually preferred, as this does not weaken the segmentation concept through firewalls. However, it is important, that the scan nodes itself does not bring in additional vulnerabilities.

One of the most known and used network scanners is `nmap` (Lyon, 2009). An off-the-shelf network scanner, which is extensible by scripts (Nmap Scripting Engine (NSE)) to use e.g. ICS specific protocols. `Nmap` is limited by the point, that it is not able to make

distributed scans by itself. However, `nmap` is an excellent and well-tested network scanner and is used as a pure scanner for this work. The required architecture for network segmentation around `nmap` in this work is an in-house development to circumvent the weaknesses of `nmap`.

Another common scanner framework is the commercial product `Nessus` from Tenable (Tenable, 2020). This vulnerability scanner supports some industrial protocols and is able to scan distributed with agents. However, it is not open source and does not offer any hardware segmentation in the scanner nodes.

An open-source alternative to this framework is Greenbone Vulnerability Management (GVM), formerly known as `OpenVAS` which was split off from `Nessus`, when it switched to a proprietary license in 2005. However, the OT security feed is only available in the professional Greenbone Security Feed (GSF), but not in the Greenbone Community Feed (GCF). Furthermore, no hardware segmentation in the scanner nodes is currently implemented.

`Powler` (See et al., 2017) is an open-source distributed network vulnerability scanner, which uses `nmap` as a scan engine. However, the scan node is not segmented and also not designed for the usage within ICSs.

For a distributed network scanner architecture (Bidaud, 2003) and hardware separation (Goldring, 2013; ?), there are already concepts available on the market. Though, to the best of the author's knowledge, there are no concepts of distributed scanners with hardware separation, aiming especially for fragile ICS networks.

(Niedermaier et al., 2019) introduced a network scanning architecture on low-performance Microcontroller Units (MCUs). This scanner is based on a basic port scan, without any service detection carried out by e.g. sensors or actuators. However, this is not suitable for use on proprietary devices and does not offer any segmentation concept.

The problems within existing approaches show that not all challenges are solved to implement a secure and feasible asset management process in ICSs (Wedgbury and Jones, 2015).

3 A SAFE AND SECURE NETWORK SCANNING ARCHITECTURE

For the use of an industrial network scanner, a number of requirements were set up specifically for the ICS

environment as part of this work. These requirements are extensions, that distinguish the here presented network scanner from common off-the-shelf scanners.

- Safe and secure distributed scanning: The scanning architecture must operate in a highly segmented network.
- Secure without updates: The network scanner itself must not introduce new vulnerabilities to an ICS in any way.
- No influences by the scan: The scan must not influence the ICS devices.
- No unintended access to scan jobs and results: The scan jobs should only be given in order by authorized persons and the scan results must be securely stored and also should only be accessible by authorized persons.

Therefore, in this paper we present an active scanning architecture, which place individual scan nodes into each sub-network and a central master node. One of the most important aspects is the hardware-based segregation of each scan node. This keeps the security measure of firewall segregated networks effective as is and limits the potential abuse of vulnerabilities within the scanner itself. Furthermore, the active scanner is optimized for use with fragile assets within OT networks, with restricted scan traffic and a limited scan rate. The configuration of the scanner is protected against manipulation, by the hardware-based segregation. This avoids the creation of network traffic, different from the pre-configured possibilities.

A common scanning architecture is shown in Figure 2, where a central server controls a number of scanning nodes. This is often necessary, when ICS networks are separated into different sub-networks, which prevents scanning from a central scan node.

The architecture, presented in Figure 2, differs from the central scan node approach and integrates well in firewall segregated sub-network topologies. First distinction of this approach is the point-to-point connection between the scan master and the single scan nodes (Network Interface Card (NIC) IT) as well as the second network card (NIC OT). Only the network card for the OT network has access to this network. Next, the network card for the IT communication is physically not connected to the OT network.

The scan master is used to control the scan tasks within the connected scan nodes and also gathers the scan results for a centralized evaluation. A user, which interfaces with the scan master, is able to configure and create certain predefined scan jobs, which are then propagated to the required scan node.

The communication between the scan master and each scan node is protected against manipulation with

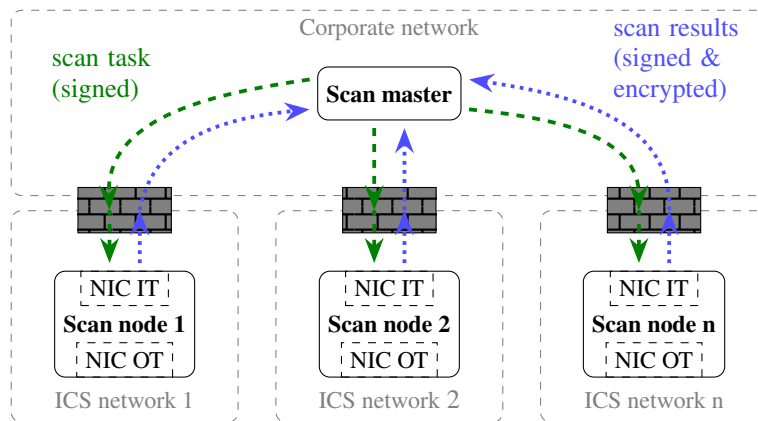


Figure 2: Distributed scanner architecture with a point-to-point connection through the firewall from the scan master to the scan nodes.

the use of signatures and the scan job messages can also be encrypted on demand. This ensures, that scan jobs on the scan nodes can only be executed from authorized users. After a scan job has been received, the scan node starts the network scan independently from the master.

A scan of a node within a sub-network results in a scan report. The scan node encrypts and signs these results, which include a timestamp, and transmits them to the scan master. This means that attackers cannot bring in historical or manipulated results into the scan master and is also not able to read the results.

The concept of the hardware separation is shown in Figure 3. On the left, the scan task comes from the scan master, this scan job/task online contains the start time of the scan. The rest of the configuration, such as which IP address spaces are scanned and the packet rate, are already stored on the OT scanner, immutable over scan job messages. These configurations could for example be saved on a SD card used by the OT scanner. This is important, because if an attacker succeeds, e.g. by taking over the scan master, the attacker can only perform regular network scans, but is not able to perform a Denial of Service (DoS) attack. The IT controller, which must be able to handle Ethernet, receives the scan task and forwards it to the OT IDD, which serves as a Gateway (GW). The IDD is capable of filtering and blocking traffic, which passes through it. The level of detail with which the IDD can do this depends on the implementation on the IDD and the application, e.g. filter rules. Afterwards, the OT IDD interprets the message and checks whether it is valid and thus serves as a packet filter and data diode. If the scan task is valid, the message is forwarded to the OT scanner, otherwise the message is dropped. If the OT scanner receives a valid signed scan job, it starts a network scan as config-

ured in the locally saved scan configuration file. After completion of the scan, the scan results are encrypted and signed and sent back to the IT IDD. If the data is valid, it forwards the results to the IT controller. The IT IDD serves as a packet filter and data diode in the same way as the OT IDD. The IT controller sends the results to the scan master, where the signature is checked and the results gets decrypted.

The scan architecture is designed to limit the attack surface for the ICS network to a minimum. The following barriers are provided within the security concept for the attack scenario, which does not directly impact the ICS network. In the following, it is assumed that an attacker has already access to the corporate network:

- ① The scan master is compromised (1st layer barrier compromised): If the scan master is taken over, by worst it is possible to send regular scan jobs which, cannot trigger a DoS, because the scan configuration is already stored on the OT scanner. Furthermore, if the attacker can manipulate or access the cryptographic private keys, he can view and manipulate scan results on the master.
- ② The IT controller is compromised (1st layer barrier compromised): Since the scan jobs are signed and the scan results are signed/encrypted, an attacker who has taken the IT controller can only interrupt the connection.
- ③ The OT IDD is compromised (2nd layer barrier compromised): If the first barrier level failed and the OT IDD is overtaken by an attacker, scan jobs can be intercepted, similar to the IT controller.
- ④ The IT IDD is compromised (2nd layer barrier compromised): If the first barrier level failed and the IT IDD is compromised by an attacker, scan results can be intercepted, similar to the IT controller.

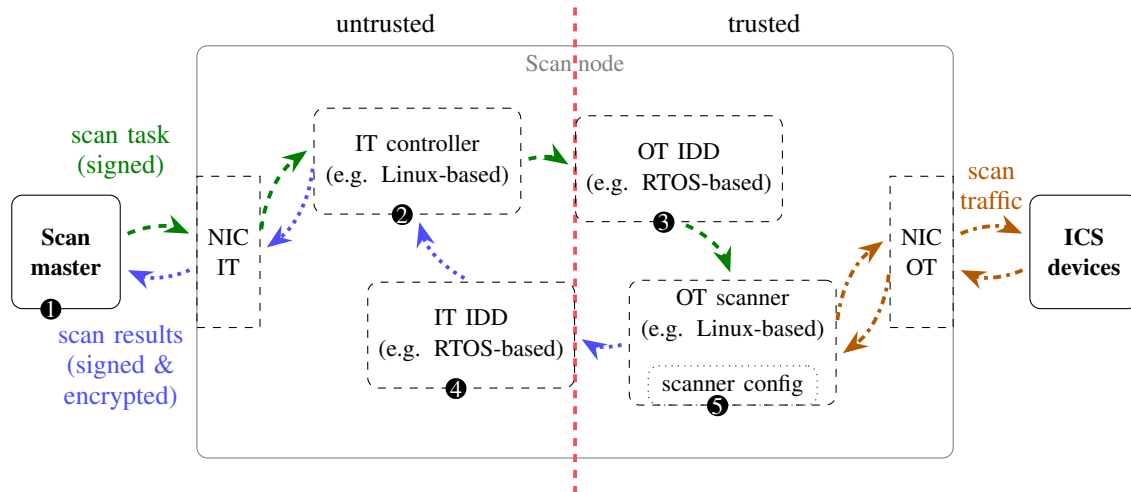


Figure 3: Methodology of the hardware concept.

- 5 Only if the 1st and 2nd barrier are hacked (3rd layer compromised), the OT scanner could be attacked. This of course allows DoS or other attacks to be carried out on the OT network. This requires several serious errors in various implementations of the defense-in-depth architecture of our scanner.

The multiple barrier security concept shown here shows that an IT security risk for the OT network only arises through multiple errors. However, the possibility for this to happen is quite low.

4 PROOF OF CONCEPT IMPLEMENTATION

To show the feasibility of the secure network scanner concept and its practicability, a Proof of Concept (PoC) implementation was built.

4.1 Hardware

The hardware of our PoC is based on two Raspberry Pis Model 3B (Raspberry Pi Foundation, 2015). One is connected to the critical ICS network side and the other on the untrustworthy office network side. The two Raspberry Pis are electrically connected via data diodes (STM32). For the data diodes, two STM32F030CCT (STMicroelectronics, 2020) based on Cortex®-M0 are used. Electrically the MCUs are connected to the Raspberry Pi via Serial Peripheral Interface (SPI). A Printed Circuit Board (PCB) (Figure 4) was designed for our PoC, which implements this hardware segmentation.

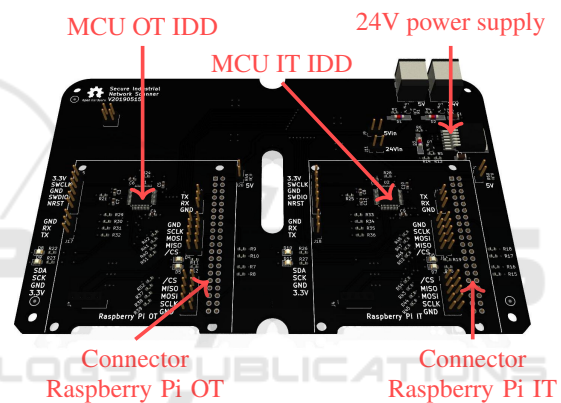


Figure 4: Pictures of the PCB of the network scanner rendered in KiCad (Charras, 2012). On the bottom, two Raspberry Pis get mounted. On the top, two displays controlled by the Raspberry Pis can be attached to prompt the current status.

Figure 5 shows the complete network scanning device within a 3d printed case for IEC/EN 60715 rail mount for appliance in industrial environment. On the bottom of the custom PCB, there are the two Raspberry Pis mounted. Furthermore, the two Organic Light Emitting Diode (OLED) displays are installed, which show the current status of the IT and OT side. The PoC hardware part is implemented as described in the concept and is about € 100 in component costs.

4.2 Software

The use of Linux-based (Torvalds, 1997) controllers on the one hand and minimal Real-time Operating System (RTOS) systems on the other offers various advantages from a security point of view. The system was prototypically implemented with

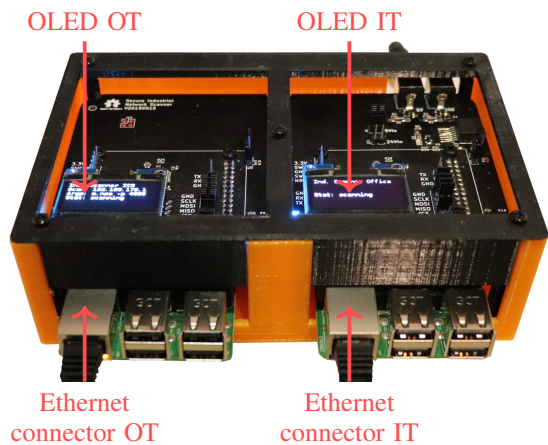


Figure 5: Picture of the complete device within a 3d printed IEC/EN 60715 rail mountable case.

Python (Rossum, 1995) on the Linux devices and the widely used and tested `nmap` network scanner with OT NSE scripts is used. These scripts offer the possibility to access information about ICS-specific and proprietary protocols. Currently, the following NSE scripts are in use `s7-info`, `pcworx-info`, `modbus-discover`, `knx-gateway-discover`, `snmp-info`, `codesys-v2-discover`, `enip-info`, `bacnet-info`, `dnp3-info`, `fox-info`, `iec-identify`, `moxa-enum`, `omrontcp-info`, and `omronudp-info`. Using Linux provides the advantage of an updateable and secure base system. The prototypical Python implementation within the PoC could be replaced in a real product by a secure programming language such as Rust (Matsakis and Klock, 2014). This could further increase the security of the scanning architecture and prevent e.g. buffer overflows.

Furthermore, the MCUs based on FreeRTOS are a part of the security concept. The firmware of the RTOS is written in C (Kernighan et al., 1988) and the implementation of the data diodes differs from the more powerful OT scanner and IT controller. These MCUs only pass the data further if they have a correct format and thus serve as a data diode and packet filter. E.g. the length and the message type are verified and if this does not meet the predefined structure, the message is not forwarded.

The results can be displayed via a website or otherwise e.g. further processed in a Security Information and Event Management (SIEM).

4.3 Minimizing the Scan Impact

As previously mentioned, it is important that the generated scan traffic can be configured. This is done through the configuration file on the OT scanner, so

that attackers cannot change it without physical access. This prevents attackers from performing a DoS attack on components in the OT network.

The scan duration depends on the selected packet rate and could be reduced by a higher packet rate. However, higher packet rates increase the risk that scanned devices get influenced by the increased traffic. As an example configuration, a slow packet rate of maximum 20 packets per second was selected for the test. This is configured on the one hand in `nmap` and on the other hand in `iptables` (Andreasson et al., 2001), so that the Linux operating system does not cause additional packets. This limitation of the network traffic leads to a high scan duration, which is, however, safer for the industrial network.

4.4 Testbed

Table 1 lists the components that were used to evaluate the network scanner. This testbed provide a solid basis for checking whether the pipeline from sending a scan task to receiving the scan results is working.

Figure 6 shows the laboratory setup on a stand, with the various PLC solutions and one scanner node 7. Here the OT test network is completely independent and the scan jobs can be received over the second wired interface (IT NIC) or over Wi-Fi. The results are also sent back this way.

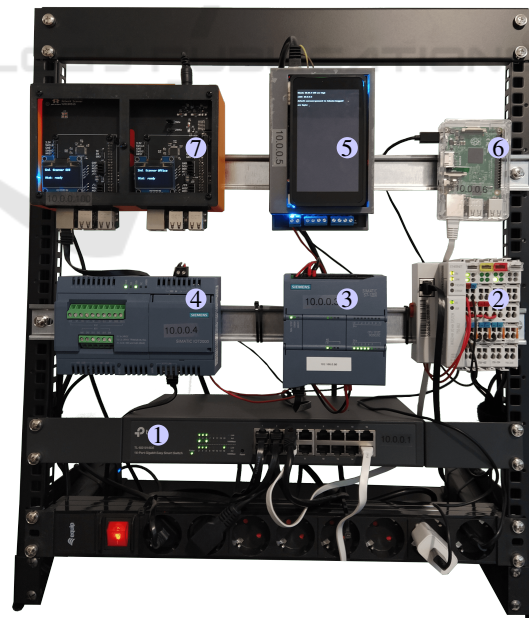


Figure 6: Picture of the testbed used for the evaluation, with the scanner node on the top left and five ICS test devices.

Table 1: Overview of the devices in the testbed.

No.	Device	IP	Software	Open pots
1	TL-SG1016DE	10.0.0.1	1.0.1 Build 20180629 Rel.58355	80
2	Wago 750-842	10.0.0.2	05.05.02(19)	80, 502, 2455
3	Siemens S7-1212	10.0.0.3	V 3.0.2	80, 102
4	Siemens IoT 2020	10.0.0.4	CODESYS Control for IOT2000 SL 3.5.16.10	22, 1217, 1534, 4840, 11740
5	STM32MP157C-DK2	10.0.0.5	OpenPLC v3 (Alves et al., 2014)	22, 502, 8080, 20000, 44818
6	Raspberry Pi 3B	10.0.0.6	OpenPLC v3 (Alves et al., 2014)	22, 502, 8080, 20000, 44818
7	Scannerbox	10.0.0.100	Raspberry Pi OS	-

4.5 Illustration of Results

Figure 7 shows the scan results of the PoC network scanner for the testbed in the web browser. All devices and open ports were found. Furthermore, through the use of NSE scripts, the industrial protocols could read out the versions of the components.

On the mirror port of the switch in the testbed, a packet capture was generated over several scans over the complete subnet (10.0.0.1/24). None of the captures shows more than the previously set maximum packet rate. A section from a scan can be seen in Figure 8.

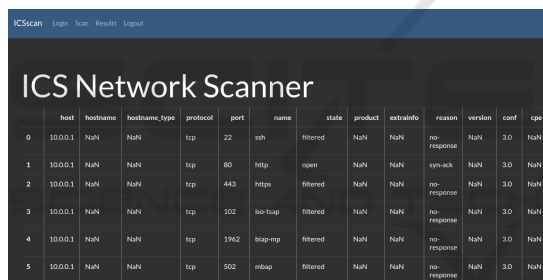


Figure 7: Results of the network scanner within the web-based dashboard.

In the plot, the traffic caused is divided into Address Resolution Protocol (ARP) and Transmission Control Protocol (TCP) traffic. Since the scanner does not have a fixed ARP table and the ARP cache is empty at the beginning of the scan, ARP requests are used to find out whether the scan target is available and responding. Only when this request has been answered, the TCP scan can be executed. In total, scanning the subnet took about 4 hours with this configuration. This mainly depends on the slow packet rate of 20 packets per second and the scan of a full /24 net with up to 254 hosts. Since the scan is carried out sequentially and each IP is tried to reach, the number of active hosts plays a minor role. In contrast, the number of open ports per IP have an impact on the scan duration.

5 CONCLUSION AND OUTLOOK

In this paper, we introduced a secure architecture for network scanning in ICS networks. Special about this concept is the strict hardware segmentation, which does not break up segmented networks. Furthermore, the scanner can be configured in such a way that undesired high packet rates cannot occur and DoS attacks are unlikely. Supplemental, the MCUs act as an IDD and can filter the communication.

The feasibility of the concept was shown with a PoC implementation. This includes a PoC implementation of the hardware and software, as well as the possibility of sending scan tasks and receiving the results. In addition, a testbed with industrial devices and software was set-up to test the PoC implementation. Moreover, the concept of the scanner architecture and especially the scanner hardware was tested and evaluated. As shown in this paper, the concept presented here is ideal for segmented and fragile ICS networks. However, this concept is clearly not limited to ICS applications and can also be used for other highly segmented or fragile systems.

In the future, this concept needs to be further developed and tested in a real world ICS. Besides, an additional feature, which could be provided by the scanner nodes, would be the checking of the network segmentation. This could e.g. be verified by attempting to establish a connection between the scanner nodes. If this connection is possible between two scanner nodes in different network segments, this could be an indication of a faulty segmentation.

AVAILABILITY

Additional material like software and network captures are available on GitHub – <https://github.com/hsainnos/ICSscannerDiode>.

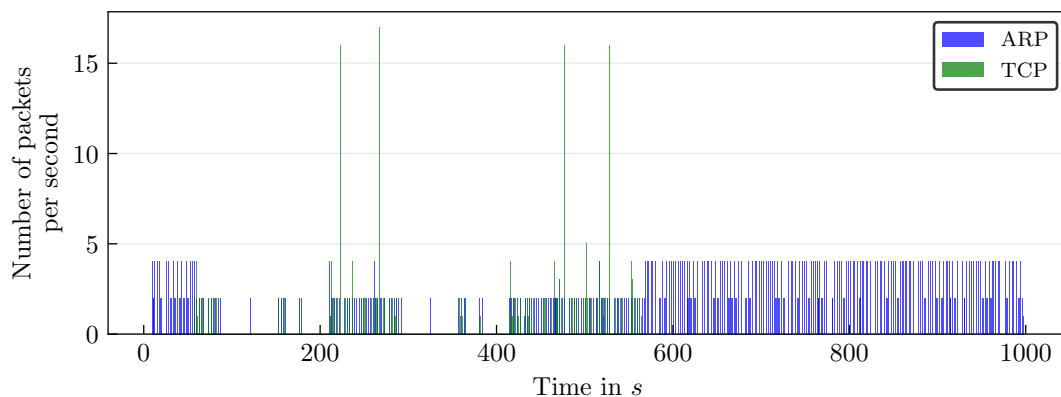


Figure 8: Packets per second caused by the PoC implementation with a rate set to a maximum of 20 packets per second.

REFERENCES

- Alves, T. R., Buratto, M., De Souza, F. M., and Rodrigues, T. V. (2014). Openplc: An open source alternative to automation. In *IEEE Global Humanitarian Technology Conference (GHTC 2014)*, pages 585–589. IEEE.
- Andreasson, O. et al. (2001). Iptables Tutorial 1.2. 2. Copyright© 2001–2006 Oskar Andreasson, GNU Free Documentation License.
- Bidaud, O. (2003). Distributed Network Architecture Security System. US Patent App. 10/118,632.
- Charras, J.-P. (2012). KiCad: GPL PCB Suite.
- Coffey, K., Smith, R., Maglaras, L., and Janicke, H. (2018). Vulnerability Analysis of Network Scanning on SCADA Systems. *Security and Communication Networks*, 2018.
- Goldring, B. A. (2013). Data diode. US Patent 8,380,913.
- IEC, D. (2020). Industrial Communication Networks - Network and System Security - Part 3-3: System Security Requirements and Security Levels (DIN EN IEC 62443-3-3:2020).
- Kernighan, B. W., Ritchie, D. M., et al. (1988). *The C Programming Language*, volume 2. prentice-Hall Englewood Cliffs, NJ.
- Knapp, E. D. and Langill, J. T. (2014). *Industrial Network Security: Securing critical infrastructure networks for smart grid, SCADA, and other Industrial Control Systems*. Syngress.
- Kuipers, D. and Fabro, M. (2006). Control Systems Cyber Security: Defense in Depth Strategies.
- Lyon, G. (2009). Nmap-Free Security Scanner for Network Exploration & Security Audits.
- Matsakis, N. D. and Klock, F. S. (2014). The Rust Language. *ACM SIGAda Ada Letters*, 34(3):103–104.
- Maynard, P., McLaughlin, K., and Sezer, S. (2018). Using Application Layer Metrics to Detect Advanced SCADA Attacks. In *ICISSP*, pages 418–425.
- Niedermaier, M., Fischer, F., Merli, D., and Sigl, G. (2019). Network scanning and mapping for iiot edge node device security. In *2019 International Conference on Applied Electronics (AE)*.
- Niedermaier, M., Malchow, J.-O., Fischer, F., Marzin, D., Merli, D., Roth, V., and von Bodisco, A. (2018). You Snooze, You Lose: Measuring PLC Cycle Times under Attacks. In *12th USENIX Workshop on Offensive Technologies (WOOT 18)*.
- Nivethan, J. and Papa, M. (2016). On the use of Open-source Firewalls in ICS/SCADA Systems. *Information Security Journal: A Global Perspective*, 25(1-3):83–93.
- Peichl, T., Ehrenberg, T., and Schriefer, J. (2015). Serial peripheral interface having a reduced number of connecting lines. US Patent 9,042,274.
- Raspberry Pi Foundation (2015). Raspberry PI 3 Model B.
- Rossum, G. (1995). Python Reference Manual.
- See, F., Seng, W. C., and Liu, T. (2017). Prowler - Distributed Network Vulnerability Scanner.
- STMicroelectronics (2020). STM32F0x0 Value Line.
- Stouffer, K., Falco, J., and Scarfone, K. (2015). Guide to Industrial Control Systems Security. *NIST SP 800-82 Rev. 2*.
- Tenable, N. (2020). Nessus Vulnerability Scanner.
- Torvalds, L. (1997). Linux: A Portable Operating System. *Master's thesis, University of Helsinki, dept. of Computing Science*.
- Vanier, D. D. (2001). Why Industry Needs Asset Management Tools. *Journal of computing in civil engineering*, 15(1):35–43.
- Wedgbury, A. and Jones, K. (2015). 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*, pages 73–83.