Detection of critical infrastructure devices on the public Internet

Martin Mladenov[1]

Supervisors: Georgios Smaragdakis[1]**, László Erdődi**[2]

[1]EEMCS, Delft University of Technology, The Netherlands
[2]IIK, Norwegian University of Science and Technology, Norway

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 24, 2023

Name of the student: Martin Mladenov
Final project course: CSE3000 Research Project
Thesis committee: Georgios Smaragdakis, László Erdődi, Alan Hanjalic

Abstract

Supervisory Control and Data Acquisition (SCADA) systems are sometimes exposed on the public Internet. It is possible to quickly and efficiently identify such exposed services. They are commonly part of critical infrastructure, so they need to be protected against cyber attacks. In the past, researchers have scanned the Internet to detect such systems. However, such data may be biased due to honeypots set up by other researchers, which are fake hosts mimicking real industrial systems in order to detect malicious attacks.

In this paper, we develop a methodology to discover SCADA systems, classify them as real or honeypots, and analyse the metadata collected from them. We show that a large part of all exposed SCADA services are in fact likely to be honeypots, and we find correlations between independent honeypot-related indicators.

1 Introduction

Industrial Control Systems (ICS) and *Supervisory Control and Data Acquisition* (SCADA) systems, which both collect and process data in order to control various critical processes, are commonly used in the industry, including critical infrastructure [1]. Sometimes, such devices can be intentionally or unintentionally exposed to the public Internet. To evaluate the security of SCADA/ICS devices and improve it, there is a need to efficiently detect such devices.

Cyber attacks against such devices are common and a successful attack can potentially have a devastating outcome [2–4]. If a device is accessible on the public internet, it can be targeted by adversaries. Furthermore, many such accessible devices have multiple known vulnerabilities which can leave them open for compromise by malicious actors [2].

In 2022, the *General Intelligence and Security Service of the Netherlands* (AIVD) observed cyber attacks with the aim of sabotage, some of which were successful [3]. The destruction of critical infrastructure via such attacks can have devastating consequences for a country [3]. European law defines critical infrastructure as "an asset, system or part thereof located in Member States which is essential for the maintenance of vital societal functions, health, safety, security, economic or social well-being of people, and the disruption or destruction of which would have a significant impact in a Member State as a result of the failure to maintain those functions" [5]. Thus, it is of utmost importance for every country to protect its critical infrastructure.

Researchers have analysed the state of the Internet in the past [2]. However, existing research has not considered the prevalence of honeypots, which mimic real SCADA/ICS devices in order to detect intrusion attempts [6].

The main contributions of this work are as follows:

- We develop a methodology for discovering exposed ICS/SCADA devices on the public Internet.
- We classify hosts as real devices or as honeypots that pretend to be real devices.
- We evaluate whether those devices could be part of critical infrastructure.
- We observe what metadata can be collected from those devices.

To achieve this, we make use of public data sources such as Censys [7] in order to find the *Internet Protocol* (IP) addresses of hosts with specific open ports, and then we connect to those hosts and retrieve metadata.

2 Background

In this section, we explain the basic concepts used in the paper. We also provide information on related work.

ICS devices commonly host various application-layer industrial protocols in order to perform their tasks. These protocols usually run on top of the standard TCP/IP protocols. [8]

Censys [7] is a search engine and data processing platform which constantly monitors the public Internet in order to index the open ports of every host exposed to the public Internet. It also has the ability to identify many services which could be accessible on those ports, including some industrial protocols. Censys is powered by ZMap [9], a state-of-the-art network scanner with the ability to scan the entire IPv4 address range in under one hour. Censys enables researchers to quickly and easily perform specific queries, for example, to retrieve a list of all hosts located in a particular country which are running a particular service on a particular port. Thanks to this, Censys significantly simplifies Internet-wide security research; without using such a search engine, building such a list would require a large amount of computational and network resources.

Siemens SIMATIC S7 *Programmable Logic Controllers* (PLCs) [10] are small industrial computers with programmable memory and are very popular in the industry; Siemens is the manufacturer of nearly a third of all PLCs worldwide [1, 11]. Due to their popularity, there are even third-party manufacturers which produce SIMATIC-compatible controllers, such as the Yaskawa VIPA series [12]. In many cases, PLCs have no password configured and the S7 communication protocol allows for programming the device, but it is "inherently insecure" [13]; a PLC which is intentionally or unintentionally exposed to the public Internet can be compromised by malicious actors. Siemens PLCs are often used in critical infrastructure, such as "power plants (including nuclear), pipelines, oil and gas refineries, hydroelectric dams, water and waste, and weapon systems" [13]. If a device is part of critical infrastructure, a successful cyber attack could cause significant damage [3]. The S7 communication protocol uses TCP/IP and usually runs on port `102`, which is indexed by Censys. It is possible to retrieve metadata about the device via this protocol.

Modbus [14] is a popular protocol developed by Modicon / Schneider Electric, which is used by a wide range of PLCs from multiple manufacturers. Modbus/TCP [15] is an Ethernet version of the protocol. However, it does not support authentication of requests. It is therefore not secure and should not be exposed on the public Internet. Modbus provides very little metadata, in comparison to the S7 communication pro-

tocol. Modbus/TCP traditionally runs on port `502`, which is indexed by Censys.

Honeypots are devices which mimic real services in order to detect the presence and activity of an attacker. There exist various honeypots for ICS devices, which are used by researchers to detect attempts to compromise critical infrastructure devices [6, 16, 17]. However, such honeypots also have the side effect of affecting legitimate research into exposed devices and skewing results.

To classify real ICS devices and honeypots, various indicators can be used. Those include the total number of open ports on the host, the *Autonomous System* (AS) of the host, the reverse *Domain Name System* (DNS) record of the IP address of the host as well as metadata retrieved from the industrial communication service itself. We will go into more detail about this in Section 4.

There are two types of honeypots: low-interactive and high-interactive. Low-interactive honeypots usually emulate only the basic parts of the protocol. High-interactive honeypots often have the ability to emulate a particular protocol better and more convincingly than low-interactive ones, making them harder to distinguish from real devices.

Conpot [6, 18] is a low-interactive honeypot which emulates SCADA systems. It has the ability to emulate multiple types of ICS devices via templates. The default template [19] mimics a Siemens S7-200 device by emulating the S7 communication protocol. It has various hardcoded values which it provides as responses to requests.

HoneyPLC [20] is a high-interactive honeypot with the ability to successfully trick reconnaissance tools into detecting it as a real device. It has the ability to detect attacks and collect malware for research purposes. For emulation of the S7 protocol, it uses the Snap7 framework as its server backend [16]. Like Conpot, this framework also has hardcoded default values, such as the device name, device serial number, and the serial number of the memory card [21].

3 Related Work

In [13], Beresford explains that Siemens SIMATIC S7 PLC as well as other industrial systems use protocols which are "inherently insecure". The reason for this is that those protocols were not designed with security in mind - they are intended to be used in closed (air-gapped) networks, where only authorised users would be able to connect to those industrial devices over the network. Such devices are not designed to withstand cyber attacks. However, this assumption is not always correct; many networks in which ICS devices are deployed are not, in fact, air-gapped, and the devices in those networks could potentially be targeted by malicious actors remotely. Hui et al. have also investigated the security of the S7 communication protocol and also found it to be vulnerable [22, 23]. Therefore, it is important to take care not to expose ICS devices on public networks such as the Internet.

In 2020, Ceron et al. summarised the number of vulnerable ICS devices in the Netherlands [2]. To achieve this, they constructed an exhaustive list of ICS/SCADA protocols and port numbers, which they used to discover devices exposed to the public Internet. They considered all such exposed devices as vulnerable. Furthermore, they identified the list of known vulnerabilities for each exposed device. To identify device types and hardware/software revision numbers, they retrieved the *Transmission Control Protocol* (TCP) banner from each device. Furthermore, they used the banner to detect honeypots. However, most honeypots do not identify themselves as such. Therefore, it is possible that the results may be inaccurate due to the large number of honeypots (as explained in Section 5).

We extend these studies by developing a methodology to classify discovered hosts as real devices or honeypots. We do this based on network attributes and service metadata.

4 Host Detection and Classification

Our methodology consists of 6 main steps, as shown in Figure 1. We retrieved a list of hosts from Censys, collected metadata from those hosts, and assigned suitable labels to each host (Section 4.1). Then, based on those labels, we determined the likelihood of each host being a honeypot or a real device (Section 4.2). Finally, we analysed the results (Section 4.3).

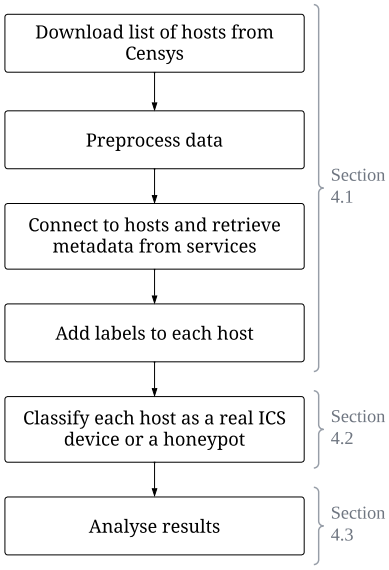

Figure 1: The steps of our methodology.

4.1 Assignment of indication labels

First, we retrieved a list of hosts from Censys [7]. We did this using the Censys *Command Line Interface* (CLI) tool [24]. Censys (and by extension the CLI tool) supports complex search queries. It provides a response in *JavaScript Object Notation* (JSON) format, which includes the following data for each host (if available):

- the IP address;
- information about the *Autonomous System* (AS), such as the *AS Number* (ASN) and the AS name;
- location information;

- a reverse DNS lookup of the IP address;
- the operating system;
- a list of detected open ports and identified services.

To retrieve such data, a Censys query such as the following may be used:
`censys search 'services.service_name=S7 and location.country_code=NL and services.port=102' --pages -1 -o hosts_0.json`.
This query returns a list of hosts located in the Netherlands which are identified by Censys to be running the S7 communication protocol on port `102`.

We made use of the feature of Censys to select only the subset of hosts identified to be running an ICS service, because there are hosts which have those ports open for unrelated reasons, for example for another service or by accident. There also exist simple generic honeypots, which do not attempt to imitate a particular service but only listen for connections to specific ports. Hence, while it is possible to perform classification and analysis on all hosts with the ports in question open, this could significantly inflate the number of discovered hosts. We decided to focus only on hosts which are running a real or fake SCADA/ICS service.

Then, we simplified the data by removing unnecessary details. In particular, we discarded the full list of ports and services and only kept information about the potential ICS service as well as the number of open ports.

After this, we performed *active probing* by completing a service handshake with each host and collecting metadata, such as the model number, the serial number, and other details. It is important to do this step shortly after retrieving data from Censys in order to reduce the chance of a host going offline or moving to a different network location. As explained in Section 4.2, this step may be skipped if collecting metadata is impractical.

Afterwards, we added labels to each host based on the data collected from Censys and from the service, which we then used to classify hosts. We assigned each device zero or more of the following indication labels:

- **Large number of open ports.** The host has more than t open ports (defined in Section 5.2).
- **Datacentre or a university network.** The AS of the host is associated with a datacentre or an educational/research institution.
- **Mobile network.** The host is on a mobile network.
- **No response from service.** The host does not respond to our attempts to fetch service metadata or the response does not match the protocol specifications.
- **Default honeypot configuration.** The host replied with one or more of the preset values in the default template of a known honeypot. We chose to use only fields which are unlikely to have those values in real devices in order to avoid false positives where we label a real device as a honeypot.

4.2 Classification

After labelling all hosts with suitable indications, we classified them based on the likelihood of each host being a honeypot. For this, we used the classification algorithm shown in Figure 2, which makes use of the indications assigned in the previous step (Section 4.1).

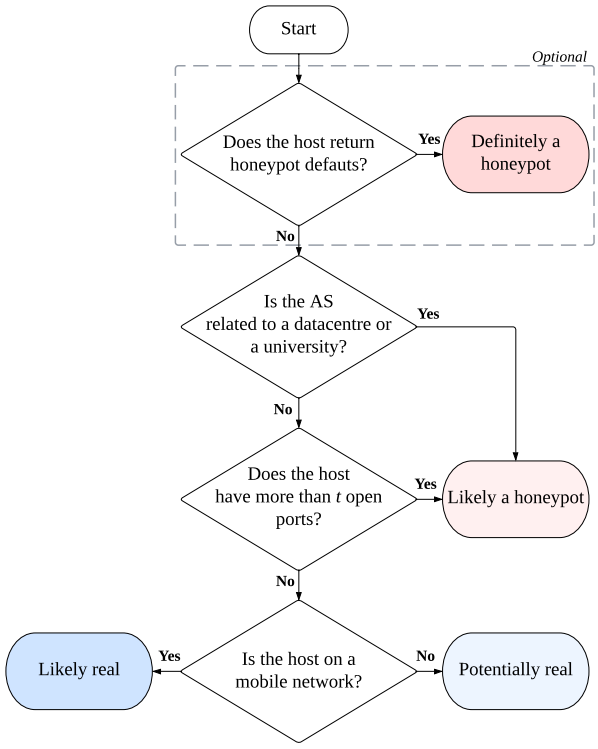

Figure 2: Our honeypot classification algorithm.

We based much of our classification methodology on generic attributes related to the network of the device instead of a particular industrial service. This makes our research applicable to many types of ICS devices and easily reproducible by everyone.

We classified as "certain honeypots" all devices which return metadata known to be part of the default configuration of known honeypot software. For this purpose, we decided to use only default honeypot configuration values which are impossible to be present in the configuration of a real device without being intentionally configured in such a way, such as a preset serial number, a nonexistent model number, or specific content of a user-specifiable text field. This step is optional. In case such data is not available or impractical to obtain, it may be skipped. As discussed in Section 5, while this step increases confidence in the results by classifying some hosts as certain honeypots, there is a high chance they would also fit one of the other criteria to be classified as likely honeypots due to their network attributes, regardless of whether metadata is available.

We classified as "likely honeypots" all ICS devices on datacentre or university networks. While it is possible for a real exposed controller to be present on such networks, it is more likely for such hosts to be honeypots set up for research.

We also classified as "likely honeypots" hosts with more than t open ports. Hosts with a large number of open ports are

Table 1: Dates of data collection and number of hosts.

Country	Protocol	Date	Size
The Netherlands	S7	24 May 2023	n=56
The Netherlands	Modbus	3 June 2023	n=407
Norway	S7	5 June 2023	n=17
Norway	Modbus	5 June 2023	n=130

very likely to be honeypots [25]. The threshold t is defined in Section 5.2.

As "likely real" we classified all ICS devices on a mobile network. Many PLCs have a modem with the ability to connect to a mobile network by inserting a SIM card, therefore it is reasonable to assume that such hosts may well be real devices. While it would be possible to deploy a honeypot on a mobile network, it would involve additional hardware and software, and it would very likely be significantly more expensive than deploying a honeypot on a fixed network. Finally, we classified all hosts which did not match any of the aforementioned classification rules as "potentially real".

4.3 Analysis

In our analysis, we considered all hosts which are classified as "certain" or "likely" honeypots, as honeypots, and all devices classified as "likely" or "potentially" real, as real. It is important to note that we did not classify any devices as "certainly real" devices, as it is impossible to detect all honeypots as such with complete accuracy. In other words, while it is possible to detect the presence of a honeypot, its absence can never be proven.

Some protocols allow for the retrieval of a large amount of metadata, such as model and serial numbers. This metadata allowed us to observe what ICS device models and manufacturers are popular as well as what types of honeypots are commonly used.

Furthermore, we could see what autonomous systems honeypots and real devices are commonly located in. We could also establish how many open ports most hosts have, in order to help future research.

Finally, based on the information above, we could manually review the metadata collected from each real device and its network information. This allowed us to find notable hosts which could be part of critical infrastructure and notify the operators.

5 Experimental Setup and Results

The results presented below are related to hosts located in the Netherlands and in Norway, which were identified by Censys to be running the S7 Communication protocol on port `102` or the Modbus protocol on port `502`. The data was collected during working hours in order to maximise the number of online devices. The dates of collection for each protocol and country as well as the number of discovered hosts can be found in Table 1. All code used in this research can be found on our GitHub repository[1].

[1]https://github.com/martinmladenov/critical-infrastructure-detection

Table 2: Autonomous Systems associated with datacentres and educational institutions.

ASN	AS Name	Classification
224	UNINETT UNINETT, The Norwegian University & Research Network	University
1101	IP-EEND-AS IP-EEND BV	University
1103	SURFNET-NL SURFnet, The Netherlands	University
8075	MICROSOFT-CORP-MSN-AS-BLOCK	Datacentre
14061	DIGITALOCEAN-ASN	Datacentre
20473	AS-CHOOPA	Datacentre
39647	REDHOSTING-AS	Datacentre
46844	SHARKTECH	Datacentre
202448	MVPS www.mvps.net	Datacentre
396982	GOOGLE-CLOUD-PLATFORM	Datacentre

Table 3: Domains associated with mobile network providers.

Domain	Provider
*.mobile.kpn.net	KPN (NL)
*.kpn-gprs.nl	KPN (NL)
*.telenormobil.no	Telenor (NO)
*.netcom.no	Telia (NO)

5.1 Assignment of indication labels

As described in Section 4, we first downloaded the list of hosts from Censys using the CLI tool and processed the data. We used information about the network of the device as well as service metadata (if available) to assign indication labels to each host, classify devices, and analyse the results. Based on the network information of each host, we assigned the following labels:

- `many_open_ports`. We added this indication to all hosts with at least t=*10* open ports (as explained in Section 5.2).
- `datacenter_as`. We placed this indication on hosts whose AS is associated with a datacentre. Table 2 lists such ASes.
- `university_as`. We added this indication to hosts on a university network. This includes ASes which are not directly associated with universities but are related to service providers which only supply educational institutions, such as SURF [26]. Table 2 lists such ASes.
- `mobile_network`. Hosts on a mobile network were labelled with this indication. This was detected by checking the domain name associated with the IP address of the host. For example, the Dutch provider KPN assigns domain names in the format `X-X-X-X.mobile.kpn.net` (where `X.X.X.X` is the IP address), making it possible to distinguish fixed and mobile connections. Table 3 shows the domain names we used for this purpose as well as the providers they belong to.

Additionally, we assigned the following indications to hosts

Table 4: Default values used for recognising hosts running the Conpot honeypot. [19]

Field	Default value
Plant identification	Mouser Factory
Serial number of module	88111222

Table 5: Default values used for recognising hosts running the Snap7 server framework. [21]

Field	Default value
Name of the PLC	SAAP7-SERVER
Serial number of module	S C-C2UR28922012
Serial number of memory card	MMC 267FF11F

running the S7 communication protocol (port `102`):

- `honeypot_defaults_conpot`. Hosts which reply with one or more of the preset values in Conpot's default template [19] were given this label. The values we used are listed in Table 4.
- `honeypot_defaults_snap7`. Hosts which reply with one or more of the default values returned by the Snap7 framework [21] were given this indication. This framework is used as the backend of honeypots such as HoneyPLC [16, 20]. The used values are listed in Table 5.
- `manufacturer_vipa`. We gave this label to hosts identifying as third-party Yaskawa VIPA *Programmable Logic Controllers* (PLCs) [12]. We classified them based on whether their S7 communication service response contained the substring `VIPA` in field 129.
- `no_plcscan_results`. If our attempts to fetch metadata using `plcscan` from a host were unsuccessful, it was given this label.

For devices running the S7 communication protocol, we used `plcscan` [27], a tool with the ability to retrieve metadata via the S7 Communcation protocol. It can retrieve the manufacturer, the model number, the serial number, the serial number of the memory card, a plant identification string (if set by the operator), and other details. While `plcscan` is much slower than ZGrab2 [28], another tool used for similar purposes, we decided to use the former, as it provides a wider range of fields.

For devices using the Modbus protocol, we did not collect any information from the service. Modbus provides a lot less information than the S7 communication protocol and none of the fields provided can be used to identify honeypots, as the values are not specific enough not to result in false positives. As discussed in Section 5.3, we found that network-related attributes are largely sufficient to classify ICS devices as real or as honeypots.

5.2 Classification

We used the algorithm described in Section 4.2 as well as the previously assigned labels. We classified each host into one of four categories: "certain honeypots", "likely honeypots", "potentially real devices", and "likely real devices".

One of the indicators used for classifying a device as a likely honeypot is whether the number of open ports exceeds a threshold t. We decided to classify hosts which have more than $t=10$ open ports as such. According to Surnin et al., hosts with more than 5 open ports are likely to be honeypots [25]. However, as we use multiple independent indicators in order to classify a host as a honeypot, we decided to increase this threshold to 10 in order to reduce the chance of falsely classifying a real ICS device as a honeypot based on this metric.

5.3 Analysis

We used two types of data in order to analyse our results. Those are the metadata retrieved from the industrial service and the network information of the host. In total, we discovered 607 hosts in the Netherlands and Norway running the S7 communication or Modbus protocols (or both). The analysis consists of three parts - based on the classification label, the metadata, and the network.

5.3.1 Classification Label

We discovered that 30.4% of all S7 communication protocol hosts in the Netherlands are honeypots. Figure 3 shows the distribution of the classification labels assigned to those hosts.

Modbus protocol hosts were less likely to be honeypots; 10.1% were found to be honeypots. However, this could be explained by the sheer popularity of Modbus in comparison to the S7 communication protocol, as the absolute number of Modbus honeypots (41) was higher than the number of S7 honeypots (17).

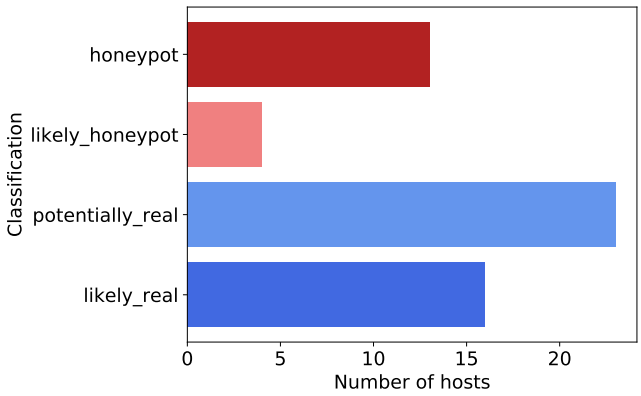

Figure 3: Discovered S7 Communication protocol hosts in the Netherlands by classification label.

5.3.2 Metadata

As described in Section 5.1, we collected metadata from hosts running the S7 communication protocol. Among the metadata collected is the model number of the module. We summarised the popularity of model numbers of hosts identified as real devices in Table 6.

Furthermore, we discovered that evaluating service metadata for classification is largely unnecessary. All hosts classified as a certain honeypot based on S7 communication service metadata also fit the criteria to be classified as a likely honeypot based on their network attributes - all of them are located in a datacentre or a university. Additionally, 77% of them

Table 6: Popularity of device models among devices in the Netherlands and Norway running the S7 communication protocol and identified as real.

Model number	Number of devices
6ES7 312-5BE03-0AB0	5
6ES7 215-1HG40-0XB0	5
6ES7 214-1AG31-0XB0	3
6ES7 214-1HG40-0XB0	3
6FC5 317-2FK14-0AB0	2
6ES7 214-1AG40-0XB0	2
6ES7 214-1HG31-0XB0	2
Other	4

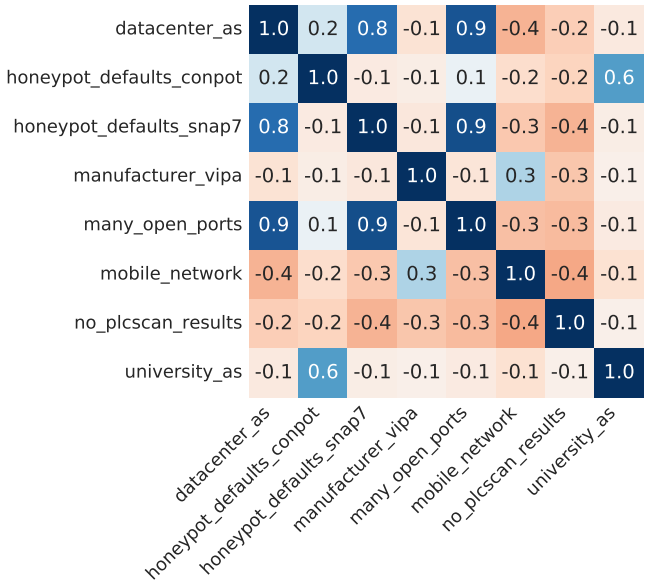

Figure 4: Correlation heatmap of the labels assigned to hosts in the Netherlands and Norway running the S7 Communication Protocol.

also have more than 10 open ports, which is an independently sufficient indicator to label them as likely honeypots. This correlation can be seen in the correlation matrix in Figure 4.

Finally, we did not discover any honeypots identifying as Yaskawa VIPA devices [12]. In fact, we observed that most such devices were on mobile networks (see Figure 4).

5.3.3 Network

We considered the Autonomous System, the number of open ports, and reverse DNS lookup results for each host. In Section 5.3.2, we found a correlation between a host having a large number of open ports and being in a datacentre or a university. We investigated this further and analysed the number of open ports of devices in datacentres and compared them to the overall statistics. We summarised these results in Figures 5 and 6. We discovered that hosts which have more than 10 ports are outliers. While only 4.1% of all discovered hosts had more than 10 ports open, this proportion increases significantly to 26.1% when considering only hosts on datacentre and university networks. This further shows that such networks are commonly used to host honeypots. This indicator supports the decision described in Section 5.2 to classify hosts with more than *t=10* open ports as likely honeypots. A summary of the number of hosts found in each Autonomous System is available in Tables 7, 8, 9, and 10 in the appendix.

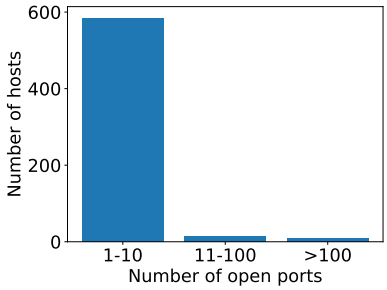

Figure 5: Number of open ports of all discovered hosts on any network in the Netherlands and Norway running Modbus or the S7 Communication protocols.

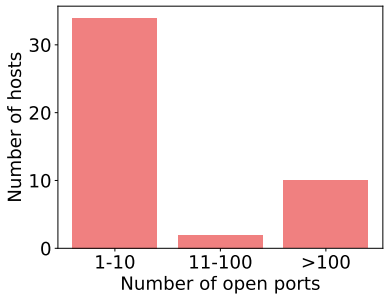

Figure 6: Number of open ports of hosts on datacentre or university networks in the Netherlands and Norway running Modbus or the S7 Communication protocols.

5.3.4 Critical Infrastructure and Responsible Disclosure

During our research, we discovered one ICS device which could be part of critical infrastructure. It was on a network operated by one of Norway's largest power grid companies.

If an ICS device is exposed on the public Internet, it could be compromised by malicious actors, as industrial protocols are often insecure by design [13]. We looked into exposed hosts which could be part of critical infrastructure. To do this, we checked whether any of the discovered hosts are on Autonomous Systems associated with critical infrastructure providers. We used the classification labels assigned earlier in Section 5.2 to review only devices classified as real in order to decrease the workload.

We found one such host on AS8542. This device was running the Modbus protocol on port `502` and was classified as potentially real by our algorithm. AS8542 belongs to one of the largest Norwegian energy companies. After the discovery, we immediately contacted the company. To safely disclose this vulnerability, we followed the Coordinated Vulnerability Disclosure model (also known as Responsible Disclosure). After our report, the Norwegian InfraCERT Incident Response Team [29] was involved and the company immediately took action to resolve the issue.

```
$ nmap -sV -Pn -p- [IP redacted]
Starting Nmap 7.80 ( https://nmap.org ) at
2023-06-10 14:55 UTC
Nmap scan report for [IP redacted]
Host is up (0.027s latency).
Not shown:  65523 filtered ports
PORT STATE SERVICE VERSION
113/tcp closed ident
502/tcp open mbap?
503/tcp open intrinsa?
504/tcp open citadel?
2000/tcp open cisco-sccp?
5060/tcp open sip?
8008/tcp open http
8010/tcp open ssl/xmpp?
8015/tcp open ssl/cfg-cloud?
8020/tcp open intu-ec-svcdisc?
47808/tcp closed bacnet
47809/tcp closed presonus-ucnet
```

Figure 7: Nmap scan of an exposed PLC on a network operated by one of Norway's largest energy companies. Nmap was unable to identify most services (indicated by "?").

Furthermore, we also reported other discovered devices to their operators. We made reports to a Dutch mobile carrier, a Dutch research network operator, and a Norwegian university. Some of the devices included in our disclosure were then taken offline as a result.

5.4 Evaluation

We analysed a few of the discovered hosts further in order to evaluate our classification results. We did this by performing port scans using Nmap [30], a tool for scanning all open ports of a host and identifying services, as well as by connecting to other services offered by those hosts.

5.4.1 Devices classified as real

As mentioned in Section 5.3, we discovered a Modbus device which could be part of the Norwegian critical infrastructure network. We confirmed via ZGrab [28] that the service running on port `502` is indeed Modbus. We performed an Nmap port scan on this host. The results can be seen in Figure 7. The Nmap results showed the presence of an HTTP server running on port 8008 as well as a lot of unidentified services. We discovered that there are also HTTP services behind ports 8010, 8015, and 8020 using `curl`, but we could not connect to them, as our connections were blocked by an *Intrusion Prevention System* (IPS), a system that monitors traffic and automatically blocks possible threats. After we submitted a report to its operator, we were informed that the device was indeed a real exposed PLC and not a honeypot.

We also selected a Modbus host classified as real on a Dutch mobile network for further analysis. We performed an Nmap scan and discovered that there is an HTTP server hosted on port 80. After connecting to it, we were presented with the login page shown in Figure 8, which further increases the probability that the device is likely not a honeypot.

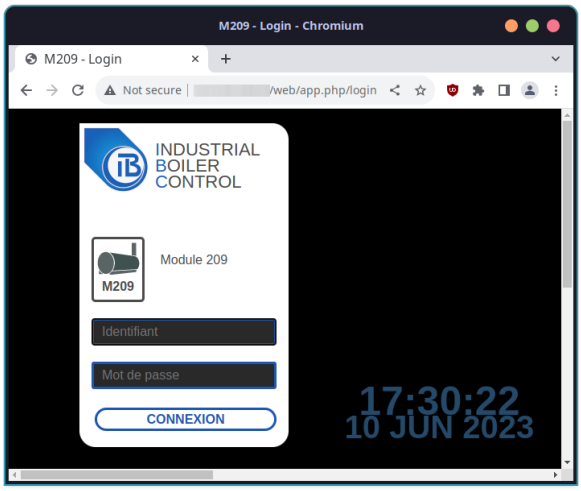

Figure 8: Login page of an exposed Modbus ICS device on a Dutch mobile network.

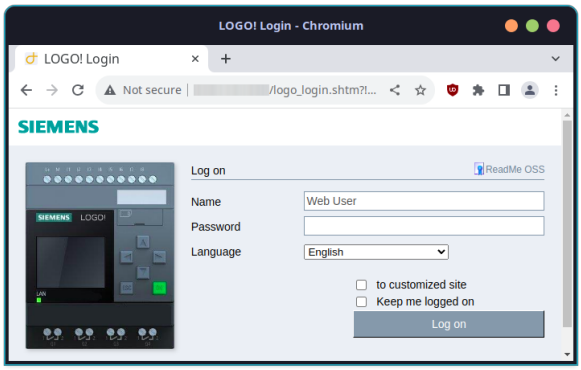

Figure 9: Login page of an exposed Siemens ICS device on a Dutch mobile network.

Then, we performed the same steps with a likely real S7 communication protocol host on a mobile network in the Netherlands. Again, we discovered a web interface on port 80, as shown in Figure 9.

5.4.2 Hosts classified as honeypots

We also performed the steps described in Section 5.4.1 on a host identified as a honeypot. It was located on a datacentre network in the Netherlands and was running the S7 communication protocol. After performing a port scan, we discovered that ports 80 and 443 were open. On port 80, we found a WordPress installation with no posts, where all links except the administrator login page did not work (shown in Figure 10). This suggests that this installation (and by extension the host itself) is a honeypot. The host had an *Secure Socket Layer* (SSL) certificate installed on the HTTP server running on port 443. SSL certificates contain information about the owner of the certificate as well as other details. The SSL certificate we found on the host (shown in Figure 11) had a Common Name set to "Nepenthes Development Team". This confirmed with certainty that this host is a honeypot, as *Nepenthes* is a honeypot platform [31].

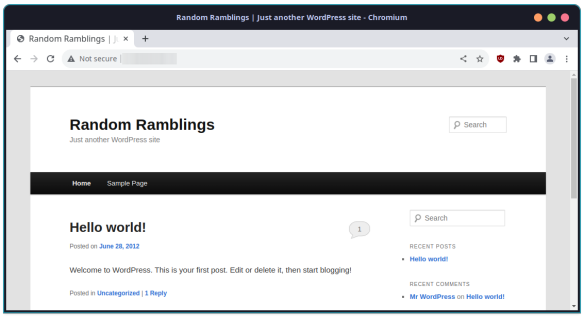

Figure 10: WordPress installation which we discovered on an S7 communication service host identified as a honeypot on a Dutch datacentre network. None of the links on this page work, except for the one leading to the administrator login form.

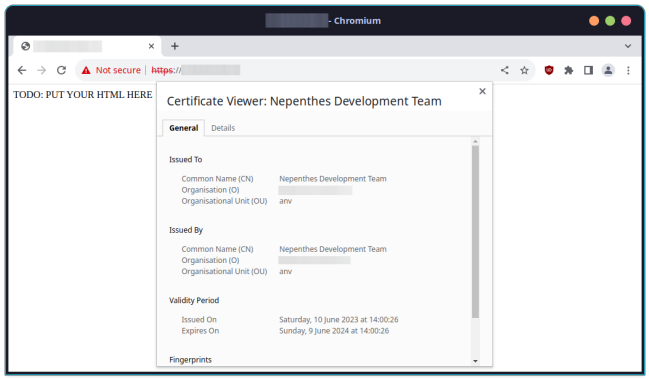

Figure 11: The SSL certificate discovered on an S7 communication service host identified as a honeypot. The Common Name of this certificate confirms with certainty that this host is a honeypot [31].

6 Responsible Research

In Section 6.1 we will reflect on the reproducibility of this research. In Section 6.2 we will discuss the ethical considerations.

6.1 Reproducibility

This research was designed with reproduction in mind. We have published all code used for labelling, classification, and analysis in our GitHub repository[1]. Furthermore, we have thoroughly described the methods we used to retrieve data from sources such as Censys [7] in Section 4.1.

Due to the nature of the research, if the data collection steps are repeated at a later date, the results have some differences, due to the constantly changing state of the Internet. Hosts may go offline or change network locations, or new hosts may appear. This research provides a methodology to assess the state of the Internet at a particular point in time. That being said, provided that the list of hosts and the collected metadata are identical, the research results are fully reproducible by following the steps outlined in our methodology.

In Section 5, we have provided information about the dates when we retrieved the data. However, we are not allowed to redistribute this data, according to the Censys Terms of Service [32]. Furthermore, while redistribution of the data collected from the hosts themselves would be possible, as it was publicly accessible at the time of collection, publishing raw non-aggregated data would be highly unethical, as such data concerns real devices whose operators have not given us explicit permission to distribute such identifying information.

[1] https://github.com/martinmladenov/critical-infrastructure-detection

6.2 Good Internet Citizenship and Ethical Implications

As with any security-related research, there are some ethical questions which need to be answered. Those can be grouped into two categories - the initiation of uninvited requests, and the implications of the research as a whole.

6.2.1 Initiation of requests

Part of our research involves making connections from our server to the discovered hosts in order to retrieve metadata. Hence, we need to discuss whether such request initiation is ethical.

Our requests only did the bare minimum necessary in order to retrieve metadata from devices. As soon as this was completed or the attempt was unsuccessful, connections were terminated. We took care not to accidentally cause damage to any exposed device or to cause service disruption. As described in Section 7, we avoided aggressive honeypot detection techniques for this exact reason.

We took into account that our activity may be misinterpreted as malicious by third parties. To prevent this, we followed the Recommended Practices described by Durumeric et al. [9]. Namely, we did the following:

- Before performing any such activities, we coordinated with the network administrators, both at the faculty level and at the university level. We informed them about the type of traffic that we would be producing as well as of the potential for complaints to be sent to the university by third parties and only proceeded after receiving permission.
- For any type of active connections, we used a server specifically for this purpose. This server had a dedicated IP address, which was not shared with any other hosts. We set a DNS entry to explain that the activity is for research purposes - `researchscan.ewi.tudelft.nl`.
- Behind the source IP address, we hosted a simple website on ports 80 and 443 which explains the purpose of the connections and provides our contact information.
- On the website, we explained that individuals and organisations can contact us by email if they would like their IP ranges to be excluded from any of our research.
- We designed our methodology and algorithms to be modular. It is not necessary to connect to devices again in order to improve the classification algorithm or to perform further analysis - an attempt to retrieve metadata from each device is done once and the response is stored, so that the data can be reused later without the need to contact devices again.

We received two complaints on the university's main abuse address - one from another university and one from a datacentre. Both of them appeared to be sent by automated intru-

sion detection systems. We replied to each of those promptly and asked the organisations to provide their IP ranges so that we can exclude them from our research. We did not receive replies. Furthermore, we were not sent any opt-out requests on the email address listed on the website hosted on our server.

6.2.2 Implications of this research

As mentioned in Section 2, honeypots are often used by researchers in order to detect and research malicious activity. If our methodology is adopted by malicious actors in order to avoid uploading malware to honeypots, this could frustrate research involving malware analysis. Once a honeypot is identified, it is likely to be blacklisted by adversaries and its data collection value will be reduced significantly [33].

Honeypots could be employed by critical infrastructure operators as part of Intrusion Detection Systems (IDS). However, as discussed in Section 7, our methodology would likely be unable to recognise such honeypots based on network information, as they would be placed within critical infrastructure networks and likely would not have many open ports. Regardless, as shown in Section 5.3, we informed operators of potentially real ICS devices on critical infrastructure networks following the Coordinated Vulnerability Disclosure model.

As mentioned in [3], attacks against critical infrastructure are commonly carried out by nation-states. Such *advanced persistent threats* (APTs) likely already have capabilities matching and far exceeding what this research paper achieves. Therefore, this research is not expected to aid the efforts of APTs aiming to compromise a country's critical infrastructure.

This research may be able to assist researchers aiming to detect and analyse attacks which target exposed ICS/SCADA devices. They can use our methodology and findings to design and deploy better honeypots that are more resilient to detection by adversaries.

7 Discussion

An unexpectedly high number of honeypots. Our study shows that a large number of the ICS/SCADA devices which are exposed on the public Internet are in fact honeypots. It is therefore very important for any research in the area to consider this when analysing results. Otherwise, it is possible that the results of such research could be misleading. Based on our results, we believe that previous studies are likely to have overestimated the number of exposed ICS/SCADA devices by up to 45%.

Many honeypots use a default configuration. We discovered a surprising number of honeypots which use a default template. This makes them trivial to detect and classify as honeypots with certainty. Preventing this is easy - honeypot operators can simply change the default configuration before deployment.

Limitations. A limitation of our study is that we relied on data provided by Censys. After performing a secondary scan via ZMap [9] for hosts in the Netherlands running the S7 communication protocol in order to evaluate the results, we discovered 11 new hosts which had not been indexed by Censys. Scanning for hosts via ZMap may produce more results than Censys, but active scanning is much more difficult and expensive.

Furthermore, we performed honeypot classification based on metadata retrieved from the service and network information. The detection mechanism could be made more accurate using more aggressive honeypot detection techniques similar to the ones described in [25], which involve storing files on devices and later making another connection to verify their presence. In PLCs, this could be done by storing a value in a particular memory location and verifying it has been persisted in a separate connection. However, this would pose legal and ethical questions, as such aggressive techniques could inadvertently cause service disruption or even damage to a PLC.

Finally, due to its nature, our methodology is unable to classify honeypots deployed within critical infrastructure networks as such, unless they use a default configuration. Such honeypots are designed to look very similar to real devices and they could be part of an *Intrusion Detection System* (IDS) deployed by a critical infrastructure provider. However, such honeypots are rare and should not have a significant effect on any further research.

8 Conclusion

In this paper, we develop a methodology to discover ICS/SCADA devices on the public Internet, fetch service metadata, classify them as real devices or honeypots based on metadata and network information, and analyse the results. We show that a large part of all exposed ICS/SCADA devices are certain or likely to be honeypots. Our analysis also provides insight into the distribution of devices and manufacturers. Furthermore, we found that nearly a third of all exposed hosts running the S7 Communication protocol were honeypots, and we show correlations between independent honeypot classification indicators. We observe that hosts in datacentres and educational institutions as well as those with a large number of open ports are likely to also have other indicators associated with honeypots.

We hope that our work is useful for other researchers aiming to perform large-scale Internet studies. Future work includes extending the study to more protocols and more countries. Also, host detection and metadata gathering could be improved. ZMap [9] could be utilised to detect services on ports less frequently indexed by Censys. ZGrab [28] could be used to improve service detection and metadata retrieval speed. To detect more advanced honeypots, different detection techniques could be implemented, for example checking the error messages provided in responses after sending invalid packets, or by analysing the responses byte by byte [33]. Confidence in detected real ICS devices may be increased using more aggressive scanning techniques, but care must be taken to stay within legal and ethical boundaries. To improve honeypot classification, machine learning could be utilised, provided that sufficient training data is available. Finally, the process could be automated further and a tool could be created which constantly scans the entire Internet, detects exposed ICS/SCADA devices, and sends an automatic notification to the operator if the device is not considered a honeypot.

Acknowledgements

I take this opportunity to express gratitude to my research supervisors, Prof. László Erdődi and Prof. Georgios Smaragdakis. Without them, this thesis would have never become a reality. Furthermore, I would like to thank Prof. László Erdődi for teaching me the core concepts of ethical hacking and penetration testing during my time abroad at the Norwegian University of Science and Technology. I would like to show gratitude to Prof. Georgios Smaragdakis for his support and useful advice throughout the entirety of this thesis. I would also like to thank all of my colleagues for their feedback and encouragement.

I would like to thank my family, in particular my mother and grandmother, for their consistent support during my entire education.

Finally, I would like to thank the Norwegian energy company operating AS8542, the InfraCERT Incident Response Team, the Norwegian educational institution, the Dutch mobile carrier, and the Dutch research network for their quick responses after our vulnerability disclosure.

References

[1] K. Stouffer, S. Lightman, V. Pillitteri, M. Abrams, and A. Hahn, "Guide to Industrial Control Systems (ICS) Security," *NIST special publication*, vol. 800, no. 82, pp. 16–16, 2011.

[2] J. M. Ceron, J. J. Chromik, J. Santanna, and A. Pras, "Online discoverability and vulnerabilities of ICS/SCADA devices in the Netherlands," *arXiv preprint arXiv:2011.02019*, 2020.

[3] General Intelligence and Security Service of the Netherlands, "Internationale dreigingen — aivd.nl." https://www.aivd.nl/onderwerpen/jaarverslagen/jaarverslag-2022/internationale-dreigingen, 2022.

[4] K. E. Hemsley and R. E. Fisher, "History of Industrial Control System Cyber Incidents," tech. rep., Idaho National Lab (INL), Idaho Falls, ID (United States), 2018.

[5] Council of European Union, "Council Directive 2008/114/EC of 8 December 2008 on the identification and designation of European critical infrastructures and the assessment of the need to improve their protection," 2008. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32008L0114.

[6] A. Jicha, M. Patton, and H. Chen, "SCADA honeypots: An in-depth analysis of Conpot," in *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, pp. 196–198, IEEE, 2016.

[7] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, "A Search Engine Backed by Internet-Wide Scanning," in *22nd ACM Conference on Computer and Communications Security (CCS)*, Oct. 2015.

[8] X. Feng, Q. Li, H. Wang, and L. Sun, "Characterizing industrial control system devices on the Internet," in *2016 IEEE 24th International Conference on Network Protocols (ICNP)*, pp. 1–10, 2016.

[9] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast Internet-wide Scanning and Its Security Applications.," in *USENIX Security Symposium*, vol. 8, pp. 47–53, 2013.

[10] Siemens, "Industrial Automation Systems SIMATIC." https://www.siemens.com/global/en/products/automation/systems/industrial.html.

[11] T. Owens, "Global PLC market share as of 2017, by manufacturer," Jan 2023.

[12] Yaskawa, "Products: Yaskawa VIPA controls." https://vipa.com/en/products/.

[13] D. Beresford, "Exploiting Siemens Simatic S7 PLCs," *Black Hat USA*, vol. 16, no. 2, pp. 723–733, 2011.

[14] Schneider Electric, "What is Modbus and How does it work?." https://www.se.com/us/en/faqs/FA168406/.

[15] A. Swales, "Open Modbus/TCP Specification," *Schneider Electric*, vol. 29, pp. 3–19, 1999.

[16] E. López-Morales, C. Rubio-Medrano, A. Doupé, Y. Shoshitaishvili, R. Wang, T. Bao, and G.-J. Ahn, "HoneyPLC: A Next-Generation Honeypot for Industrial Control Systems," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 279–291, 2020.

[17] F. Xiao, E. Chen, and Q. Xu, "S7commTrace: A High Interactive Honeypot for Industrial Control System Based on S7 Protocol," in *Information and Communications Security: 19th International Conference, ICICS 2017, Beijing, China, December 6-8, 2017, Proceedings 19*, pp. 412–423, Springer, 2018.

[18] Honeynet Project, "Conpot." http://conpot.org/.

[19] Conpot, "template.xml." https://github.com/mushorg/conpot/blob/f0e6925fb9632172922abe41b293d7ee438fa60b/conpot/templates/default/template.xml.

[20] SEFCOM, "HoneyPLC." https://github.com/sefcom/honeyplc.

[21] Snap7, "server.c." https://github.com/SCADACS/snap7/blob/f6ff90317ca5d54250f4dcd29209689a74e26d82/examples/plain-c/server.c.

[22] H. Hui and K. McLaughlin, "Investigating Current PLC Security Issues Regarding Siemens S7 Communications and TIA Portal," in *5th International Symposium for ICS & SCADA Cyber Security Research (ICS-CSR) 2018*, pp. 67–73, 2018.

[23] H. Hui, K. McLaughlin, and S. Sezer, "Vulnerability analysis of S7 PLCs: Manipulating the security mechanism," *International Journal of Critical Infrastructure Protection (IJCIP)*, vol. 35, p. 100470, 2021.

[24] Censys, "Censys CLI Documentation." https://censys-python.readthedocs.io/en/stable/usage-cli.html.

[25] O. Surnin, F. Hussain, R. Hussain, S. Ostrovskaya, A. Polovinkin, J. Lee, and X. Fernando, "Probabilistic Estimation of Honeypot Detection in Internet of

Things Environment," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, pp. 191–196, 2019.

[26] SURF, "SURFinternet: een snelle, betrouwbare internetverbinding." https://www.surf.nl/surfinternet-een-snelle-betrouwbare-internetverbinding.

[27] D. Efanov, "plcscan." https://code.google.com/archive/p/plcscan/.

[28] ZMap Project, "ZGrab2." https://github.com/zmap/zgrab2.

[29] InfraCERT, "InfraCERT." https://www.kraftcert.no/en/.

[30] G. Lyon, "Nmap: the Network Mapper - Free Security Scanner." https://nmap.org/.

[31] P. Baecher, M. Koetter, T. Holz, M. Dornseif, and F. Freiling, "The Nepenthes Platform: An Efficient Approach to Collect Malware," in *Recent Advances in Intrusion Detection* (D. Zamboni and C. Kruegel, eds.), (Berlin, Heidelberg), pp. 165–184, Springer Berlin Heidelberg, 2006.

[32] Censys, "Terms of Service - Censys." https://censys.io/terms-of-service/.

[33] A. Vetterl and R. Clayton, "Bitter Harvest: Systematically Fingerprinting Low- and Medium-interaction Honeypots at Internet Scale," in *12th USENIX Workshop on Offensive Technologies (WOOT 18)*, (Baltimore, MD), USENIX Association, Aug. 2018.

Appendix

This appendix contains the autonomous systems we discovered and the number of hosts associated with each, split by country and protocol.

Table 7: Number of discovered S7 communication protocol hosts per AS in the Netherlands.

ASN	AS Name	Count
1136	KPN KPN National	24
14061	DIGITALOCEAN-ASN	11
1101	IP-EEND-AS IP-EEND BV	3
33915	TNF-AS	2
15542	ZEELANDNET DELTA Fiber Nederland	2
28685	ASN-ROUTIT	2
12414	NL-SOLCON SOLCON	1
34373	XXLNET	1
39647	REDHOSTING-AS	1
207375	FIBO	1
20473	AS-CHOOPA	1
49033	CRITICALCORE	1
8075	MICROSOFT-CORP-MSN-AS-BLOCK	1
42707	EQUEST-AS	1
38919	NETREBEL	1
39686	ASN-EUROFIBER	1
41960	NEXTPERTISE Nextpertise	1
15435	KABELFOON DELTA Fiber Nederland	1

Table 8: Number of discovered Modbus protocol hosts per AS in the Netherlands.

ASN	AS Name	Count
1136	KPN KPN National	226
33915	TNF-AS	60
31615	TMO-NL-AS	19
49033	CRITICALCORE	13
28685	ASN-ROUTIT	11
8075	MICROSOFT-CORP-MSN-AS-BLOCK	9
12414	NL-SOLCON SOLCON	9
39647	REDHOSTING-AS	7
14061	DIGITALOCEAN-ASN	6
1103	SURFNET-NL SURFnet, The Netherlands	4
38930	FIBERRING Amsterdam, Netherlands	3
15542	ZEELANDNET DELTA Fiber Nederland	3
28788	UNILOGICNET-AS	2
6830	LIBERTYGLOBAL Liberty Global formerly UPC Broadband Holding, aka AORTA	2
196640	ASPIDER	2
13127	T-MOBILE AS for the Trans-European T-Mobile IP Transport backbone	2
21221	INFOPACT-AS The Netherlands	2
50673	SERVERIUS-AS	2
39686	ASN-EUROFIBER	2
207375	FIBO	2
15435	KABELFOON DELTA Fiber Nederland	2
49784	NL-NETVISIT	1
20847	PREVIDER-AS	1
206894	WHOLESALECONNECTIONS	1
207456	PANGEA-CONNECTED	1
396982	GOOGLE-CLOUD-PLATFORM	1
51088	A2B	1
35224	PLINQ	1
15480	VFNL-AS Vodafone NL Autonomous System	1
42707	EQUEST-AS	1
50554	NCBV-BACKBONE	1
201975	UNISCAPEB IT-Services & Hosting	1
50522	POCOS	1
57795	NGNETWORKS	1
5524	BREEDBANDNEDERLAND	1
201290	BLACKGATE	1
30925	SPEEDXS-AS	1

43995	NL-KABELTEX Kabeltex B.V.	1
50266	TMOBILE-THUIS	1
202448	MVPS www.mvps.net	1

Table 9: Number of discovered S7 communication protocol hosts per AS in Norway.

ASN	AS Name	Count
2119	TELENOR-NEXTEL Telenor Norge AS	5
12929	NETCOM-AS Oslo, Norway	4
57660	COM4-AS	3
203424	TIKT	1
29695	ALTIBOX_AS Norway	1
25400	TELIA-NORWAY-AS Telia Norway Core Networks	1
203995	ICENET	1
31264	STIM-COMPUTING-AS Peering: peering@visolit.no	1

Table 10: Number of discovered Modbus protocol hosts per AS in Norway.

ASN	AS Name	Count
2119	TELENOR-NEXTEL Telenor Norge AS	36
57660	COM4-AS	33
2116	GLOBALCONNECT-	25
29695	ALTIBOX_AS Norway	11
15659	NEXTGENTEL NEXTGENTEL Autonomous System	8
12929	NETCOM-AS Oslo, Norway	4
43568	VEV-ALO1	2
16185	RINGNETT-NORWAY RingNett AS Autonomous System	2
203995	ICENET	2
8542	EVINY-AS8542 Norway	1
224	UNINETT UNINETT, The Norwegian University & Research Network	1
41164	GET-NO GET Norway	1
29492	EIDSIVA-ASN	1
8478	ASN-GIGNETWORKS	1
49082	ZONES-AS	1
25400	TELIA-NORWAY-AS Telia Norway Core Networks	1