WaterLeakage: A Stealthy Malware for Data Exfiltration on Industrial Control Systems Using Visual Channels[*]

A. Robles-Durazno[1], N. Moradpoor, J. McWhinnie, and G. Russell

Abstract— **Industrial Control Systems (ICS) have faced a growing number of threats over the past few years. Reliance on isolated controls networks or air-gapped computers is no longer a feasible solution when it comes to protecting ICS. It is because the new architecture of control networks requiring interaction with Internet technologies. Such connection has allowed businesses to access the data from Distributed Control Systems (DCS) or Programming Logic Controllers (PLC) from anywhere in a real-time manner. On the other hand, this connectivity exposes control networks, with low or poor security in place, to a wide range of new attacks such as ransomware, trojans and malware. Moreover, the human factor is one of the biggest threats on ICS given that unintentional mistakes or disgruntled employees can potentially cause hazardous changes/damages in the control process. In this paper, we present a stealthy malware named as WaterLeakage that exfiltrates information from an uninterrupted clean water supply system using a visual covert channel. For the experiment, we physically modelled such system using the Festo Rig MPA Compact Workstation. Our developed plug and play WaterLeakage malware is placed on a Raspberry Pi connected to the control network. The malware extracts vital information from the PLC such as CPU Model, Vendor, and Input Memory Values and then exfiltrates this information using two lamps connected to the output memory of the PLC. In our experiments, a receiver has been configured with two different resolutions to record the exfiltrated information and further decode them back to the original sensitive data. The results show that by using our WaterLeakage malware an attacker can successfully collect the important information from the control process, which can be used further to plan more sophisticated attacks on ICS.**

I. Introduction

Industrial Control Systems (ICS) can consist of several types of controllers and devices to monitor and control industrial processes. They can vary from small embedded control systems to large networked Distributed Control Systems with many sensors and actuators [1]. Critical Infrastructures such as transport, food, chemical and energy utilise ICS. These systems are considered critical and necessary for a country to function [2]. The evolution of technology and computing capabilities allowed control equipment such as Programming Logic Controller (PLC) to adopt network features with poor or no security measures [3]. Currently, ICS have become accessible over the Internet, which has led these systems to face the same threats as IT networks, increasing the attention of skilled hackers due to the greater impact of executing cyber-attacks [4]. For example, on April 6, 2018 [5], several critical infrastructure operations were affected after a large-scale attack was executed to Cisco IOS switches. This is a clear example that ICS face an increasing number of threats not only from the vulnerabilities found in control equipment but also for the ones present in IT equipment involved in control networks. Another large-scale attack was registered in 2017 with the spread of the ransomware WannaCry, which affected thousands of devices around the world. National Health Systems across the Europe were severely affected by this attack as critical patient data, and sensitive information was locked up [6-7]. The consequences of spreading this type of attack across critical control equipment would be devastating. For instance, real-time systems could be confronted with imminent shutdown or worse. Critical Infrastructures could face harmful threats because daily services used by modern society such as water, electricity or transport could be suspended due to ransomware attacks.

In 2010 a new form of cyber-attack to ICS emerged. A sophisticated malware called Stuxnet targeted Iranian nuclear facilities. The malware was introduced into the control network by a thumb drive. The malware was designed to exploit zero-day vulnerabilities in Windows Operating Systems and Siemens (S7-315, S7-417) devices [8]. The aim of this malware was to modify the PLC code and deviate its behaviour. According to the reports, it is believed that at least 984 centrifuges at Iran's uranium enriched facility were destroyed. Stuxnet caused a minor damage to the nuclear program compared with the potential damage that it could have had [9]. This is the first recorded attack that shows an adversary with detailed knowledge of the control process. It is thought that Stuxnet was planned five years prior to its release. An important lesson obtained from Stuxnet is that even air-gapped computers are not immune from cyber-attacks. In most of the cases, hackers find mechanisms for bypassing security controls and accessing to classified information.

Another attack vector used for hackers focuses on extracting sensitive information from specific ICS [10]. For instance, the malware Havex[11] and GreyEnergy[12] targeted ICS. Havex was discovered in 2013 and it was tailored for espionage in industries such as pharmaceutical, defence, energy and petrochemical. When a machine is infected, the malware Havex starts scanning the system and devices connected on the same network looking for information such as usernames, passwords, or files related

[*] Research supported by Edinburgh Napier University.

[1] A. Robles-Durazno, N. Moradpoor and G. Russell are with the School of Computing, Edinburgh Napier University, Edinburgh, 10 Colinton Road EH10 5DT, Scotland, UK. a.roblesdurazno@napier.ac.uk

J. McWhinnie is with the School of Engineering & the Built Environment, Edinburgh Napier University, Edinburgh, 10 Colinton Road EH10 5DT, Scotland, UK.

to ICS or SCADA systems. After the information is collected, it establishes a connection to a remote server to exfiltrate and store the information. It is known that Havex is distributed to targeted users through phishing emails, exploits which is further injected to control program installers.

In this paper, we introduce a stealthy malware named as WaterLeakage capable of locating Siemens PLCs in the control network. It reads the information from the PLC such as sensors related data, CPU model and software version. Finally, it exfiltrates the information using lights as a covert channel. In our proposed scenario, we assumed that an insider aids the intruder during the execution of the attack. The results show that the attacker is capable of learning the control process operation, from the information collected after the execution of the attack. In a real scenario, an attacker with such information would be able to execute harmful attacks against a specific control process, for instance changing the dosing of chlorine in a water chlorination process that can affect the consumers enormously even endanger their lives.

The remainder of this paper is organized as follows. Section II provides a review of the related work on covert channel attacks. In Section III, our threat model for the ICS is proposed. Section IV describes the testbed implemented for our experiments and in Section V the results are presented and discussed. Section VI presents the conclusions which is followed by the acknowledgements and references.

II. Related Work

In this section, the work related to data exfiltration in different types of networks are discussed. We noticed that this issue has always been reviewed and measured based on the medium used for the data leakage. For example, electromagnetics, magnetic, optical (such as keyboard LEDs, hard drive activity LED, switch/router LEDs, and screen power LEDs), thermal, acoustic, and electric (power consumption). The electronic literature search was conducted on Google Scholar with 2011 to 2018-year filter applied.

For example, for electromagnetics medium, authors in [14], [15] demonstrated a malware named as AirHopper that exfiltrates sensitive data from a highly secure network to a nearby smartphone via radio signals emitted from the screen cable. Moreover, for magnetic medium, authors in [16] proposed a malware that can leak sensitive information from air-gapped computers using low frequency magnetic signals generated by the CPU cores. Furthermore, for optical medium, authors in [17] proposed LED-it-GO covert channel that uses hard drive LED to exfiltrate data from highly secure computers. Additionally, for thermal medium, authors in [18] established a bidirectional covert channel using temperature changes between two adjacent air-gapped computers to exfiltrate sensitive information. Also, for acoustic medium, authors in [19] used noise intentionally emitted form computer's fan located in a highly secure network for sensitive information leakage.

However, in this paper, we review the existing data exfiltration papers from a different viewpoint. This includes the three popular network categories of: traditional computer systems, Internet of Things (IoT), and smartphones. Additionally, in this paper, we introduce Industrial Control Systems (ICS), which are now being integrated into the IoT ecosystem, as a new emerging category of data leakage attack.

A. IoT

In [20], authors used IoT smart lights, whose functionality is to control the colour and strength of lights in a given room, to exfiltrate data from an office building. Their conducted attacks resulted in: 1) successfully using the smart lights as a covert Light Fidelity (LiFi) communication systems to exfiltrate sensitive data from a highly secure office building and 2) rapidly changing the light frequency to trigger seizures in people with photosensitive epilepsy. For the experiments, they used both high-end (i.e. an expensive Philips HUE system) and low-end (i.e an inexpensive smart light manufactured by LimitlessLED) IoT smart light systems in addition to an optical receiver placed in a safe distance from the target to receive the leaked data. Addressing their results, they were successful in covertly leaking sensitive data (e.g. passwords and private encryption keys) with a speed of several bits per second from over 100 meters. Additionally, they have proposed solutions for the vulnerabilities they found. This includes the essential need for penetration testing of IoT products and critically thinking about the way the IoT devices are integrated (e.g. in cities or in critical infrastructure networks) and separate the lights control networks from the Internet to protect against attacks such as blackout.

In [21], authors exploited the lack of authentication and identification in Infrared (IR) protocol by designing and developing a Malicious IR hardware Module (MIRM) in an air-gapped network to control nearby IoT devices in order to exfiltrate sensitive information. Their proposed MIRM can control a range of IoT devices from smart TV set-top boxes, to smart air-conditioners, smart electric fans, and robot sweepers. Using their proposed attack model, they successfully built a covert channel with a smart TV set-top box which is controlled by IR signals sent by their developed MIRM module embedded in a compromised keyboard. They used a conversion algorithm to send text data from the compromised keyboard to the TV set-top box through the covert channel. Addressing their results, the rate of the covert channel can reach 3.15 bits/sec. Additionally, they proposed some countermeasures for both IR remote control (e.g. avoid using IR remote control protocols or giving a prompt tone for every received command) and against covert channels (e.g. in terms of design, procedural or technical countermeasure).

B. *Traditional Computer Networks*

In [22], authors proposed a malware named PowerHammer that uses power lines to exfiltrate sensitive data from a compromised air-gapped computer. An air-gapped computer is a computer located in a secured network/environment for which comprehensive security measures are taken into account to maintain both physical and logical separation from less secured computer/environment/networks. Air-gapped networks are used in sensitive and restricted environment/applications such as military, critical infrastructure, and finance sectors. Their proposed PowerHammer malware runs on a compromised air-gapped computer in which the sensitive data is transmitted on top of the computer's current flow and then encoded and exfiltrated out of the air-gapped environment using the power lines. They then introduced two types of attack to retrieve the exfiltrated data from the power lines: line level power-hammering where the attacker places a probe on computer power cables and phase level power-hammering where the probe is placed in the main power panel of the whole floor. Addressing their results, they were successful in processing the signals from the power lines in both attacks and decoding them back to the original sensitive data. Some detection and prevention techniques such as: host-based detection, signal jamming, and signal filtering are also discussed by the authors.

In [23], authors used a row of status LEDs on switches and routers to exfiltrate sensitive data such as: encryption keys, passwords, and files from a highly secure air-gapped network. For this, they developed and then executed their malicious code on a LAN switch and router which allowed them to have full control of the status LEDs. The malicious code then encoded and modulated the sensitive data over the blinking of the LEDs. The generated signals were then recorded by various receivers such as remote cameras, security cameras, smartphone cameras, and optical sensors. Addressing their captured results, they were successful in covertly exfiltrating the sensitive data from the highly secure air-gapped networks via the status LEDs on switches and routers from a bit rate of 10 bit/sec to more than 1Kbit/sec per LED. They have also discussed different prevention and detection techniques including camera ban from air-gapped networks, covering the status LEDs with black tape, device and cable shielding, randomly interrupting the LED signals using noise signals, mitigating firmware level attacks, using the device's JTAG debugging interface and extracting the memory for security purpose, as well as monitoring the LEDs on switches and routers to detect covert signals.

C. *Smartphones*

In [24], authors analyzed various covert channels on mobile phones with a particular focus on the available hardware resources that can be exploited and then maliciously used to leak data between applications on the same device. In their presented work, they have discovered two covert channels including the battery and the phone call component. Additionally, they proposed a new communication protocol which can be used between their two discovered covert channels in order to achieve high throughput. For their experiments, they used a Samsung Galaxy S phone running Android version 4.2.2 using the throughput metric (ratio of the input length and time taken) to evaluate their discovered concealed channels along with their proposed communication protocol. Their study showed that a high throughput (more than 30kbps) can be achieved with the use of phone call component as a covert channel.

In [25], authors proposed a malware named Soundcomber which is a stealthy and context-ware sound trojan for smartphones. The Soundcomber has access to on-board smartphone sensors (i.e. audio and microphone) for illicit collection of sensitive information such as credit card data and PIN numbers from both tone and speech-based interactions with the smartphone. For the experiments, they used two scenarios. In the first scenario, the Soundcomber used a legitimate application with network access such as a smartphone browser to exfiltrate sensitive information and in the second scenario, the malware used a paired Trojan application with network access for the sensitive data leakage. For evaluation, they considered effectiveness (i.e. service hotline detection, tone/speech recognition, detection by anti-virus applications, and reference monitor) and performance (i.e. service hotline detection, tone/speech recognition, covert channels, and reference monitor) all representing success for the Soundcomber malware. They have also designed and implemented a defense architecture to defeat Soundcomber such as: mute local playback of tones to prevent tone-based attacks, application isolation which disallows simultaneous access to smartphone resources between multiple applications, and fine-grained permission model for smartphone sensors (e.g. audio, video, and microphone).

In this paper, we introduce Industrial Control Systems (ICS), which are now being integrated into the IoT ecosystem, as a new category of data leakage attack given that ICS data exfiltration could have more devastating impacts in comparison with the other three categories mentioned above. ICS have been widely used in critical infrastructure and large industries in which career and wellbeing of nations is highly dependent on their continuity and operations. Thus, the protection of these utilities, that provide critical services to the nation, from attacks such as data exfiltration is vitally important given the difficulty imposed by any failure or damage to these systems and/or their services.

In our work in this paper, we introduce a new stealthy malware named WaterLeakage which in spite of the existing work targets the ICS and more specifically a PLC in a water treatment system. Unlike the existing work, our WaterLeakage malware is a plug and play malicious software that does not need any configurations nor code installations on the susceptible PLC. It targets the PLC vulnerabilities currently available in the market to exfiltrate its sensitive information such as IP address, sensor related data, CPU model, software version, and different memory spaces (e.g. PLC Input and Output memory). This sensitive exfiltrated information can be obtained during the reconnaissance phase which can further be used by the hacker/malicious insider to launch more devastating cyber-attacks on ICS.

III. Threat Model

The threat model involves gaining access to the control network. Previous attacks to ICS demonstrated the feasibility of infecting computers inside control networks with the intention of executing malicious attacks or gathering sensitive information [8], [13], [26]. In this paper, we assume that the attacker managed to get access to the environment and connect a Raspberry Pi [27] to the control network. The Raspberry Pi contains a malware that is programmed to scan and find out PLCs connected to the control network. When a PLC is located, the Raspberry Pi crafts network packets and send requests to the PLC aiming to gather sensitive information such as IP address, serial number, CPU status. This information is then exfiltrated from the system using visual channels. This threat model also involves a slow-motion camera as a receiver to record the exfiltrated information. Finally, the attacker can decode this information by applying image processing techniques on the captured video. This threat model is applicable to any lighting control system.

The ICS security frameworks [1], [28] indicates that the network switches that belong to the control network should shutdown the unused ports. However, we find out this threat model applicable in a scenario where the hacker is an insider who has access to the organisation [29]. We use a testbed implemented in our previous work [30] to demonstrate the effectiveness of this data exfiltration scenario.

A. Insider Threat

ICS can be under different threat in various ways and the source of the attack could come from: terrorist groups, skilled hackers, outsourcing companies, natural disasters and insiders. The insider is a malicious threat originated within the organisation and its actions poses a considerable harm to equipment, financial or reputation[29]. A disgruntled insider could be motivated by money, steal data or cause damage to the company's reputation. Although it should also be considered that unintentional employee's actions such as negligence or recklessness could also lead to security breaches[1]. In this paper, we consider that an insider who has access to the control network is involved in the attack.

IV. Design and Implementation

In this section, we describe the testbed and methods implemented to demonstrate in practice the data exfiltration on ICS using visual channels.

A. Testbed

In this paper, we use a model of a clean water supply system using the Festo MPS PA Compact Workstation Rig [31]. The operation of this control system is fully described in our previous work [30]. Fig 1 shows the testbed implemented for this paper. The Festo Rig simulates an essential utility such as an uninterrupted clean water supply system. Its components (sensors and actuators) are hardwired to the Simatic S7-1500 PLC [32]. The control strategies and operator interfaces are programmed and configured using TIA Portal V14. The PLC and the Supervisor Console are able to communicate by means of a network switch. This communication link represents the control network. The Raspberry Pi connected to the control network represents the component placed by the adversary and it is programmed to collect and exfiltrate the information obtained from the control process using two lamps attached to the Festo Rig. The visual information is captured by a video receiver and then processed by a computer in order to decode the original sensitive data.

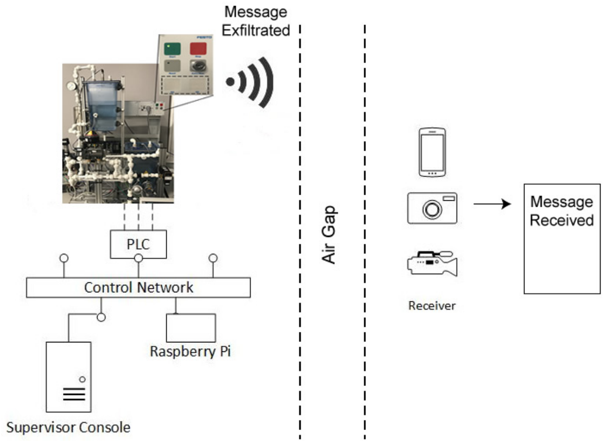


Figure 1. Visual Covert Channel.

B. Stealthy WaterLeakage Malware

The Raspberry Pi connected to the control network contains our stealthy WaterLeakage malware which is developed in Python 2.7.15. The malware starts its operation and targets devices connected to the control network that are listening to port 102 for incoming connections, due to this is the port used by default in the Siemens PLCs. To achieve this, the malware inside the Raspberry Pi executes a network scan using the NMAP tool [33] against that specific port aiming to obtain the list of PLC's IP Addresses in the control network. The second stage of the malware is to start obtaining information from the different memory spaces from the PLC. This goal is feasible as Siemens supports a wide range of protocols over Ethernet. Thus, it is possible to collect information from the Siemens PLC by crafting ISO 8073/X.224 COTP packets [1]. Figure 2 shows the structure of the crafted packet.

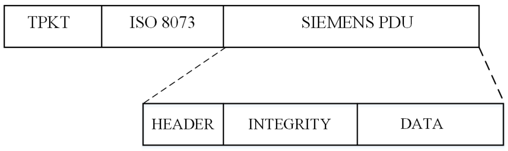


Figure 2. ISO-COTP Packet Structure.

Another advantage for the attacker is that Siemens PLCs use fixed spaces of memory for addressing the Inputs and Outputs. Thereby, at this stage, the attacker may also read the values provided by the sensors connected to the Input memory and the values sent to the actuators through the Output memory. It should be noted that from the Simatic S7-1200 model onwards Siemens introduced a new feature called memory optimisation with the intention of allocating data-blocks and function-blocks in a given space of memory. This feature makes it difficult for the attacker to obtain information from that specific space of memory. On the other hand, it is possible to obtain such information in older models where the Input, Output and Working Memory are fixed.

In the third stage of the attack, the information collected from the PLC memory is transformed from text to binary using the binascii module available in Python [34]. Finally, to exfiltrate the information we use the lamps Q1 and Q2 attached to the Festo Rig and showed in Figure 3. Lamp Q1 (on the left in Figure 3) represents 0 and lamp Q2 (on the right Figure 3) represents 1. These lamps are connected to the digital output of the PLC. It can be argued whether in a real ICS the PLC is used for driving lamps because in most of the scenarios the PLC interface uses touch screen or computer technology, however it is still common practice to drive some status indication in this way, and in this paper we want to demonstrate the feasibility of gathering and disclosing sensitive information from an ICS with this simple procedure.

C. Sender.

Before exfiltrating the information from the Control Process through the Lamps, depends on the attack scenario, there are many parameters that need to be considered by the attackers. Visual channels are complex to analyse because it varies reliant on every person. It is well-known that in average the human eye can perceive flickers that occur at about 60Hz [35], [36]. For that reason, the attacker should consider the parameters involved in the human vision before planning the data exfiltration. Figure 4 shows an analysis where X-axis represents the delay between each packet before it is exfiltrated through the corresponding lamp and the Y-axis represents the time taken to exfiltrate 942 bits of data. It can be seen that the shorter the delay the lesser time is taken to transmit the message. It takes 9.21 seconds to transmit the 942 bits of data when no delay is expected between the packets. Table I shows the values used to plot Figure 4.

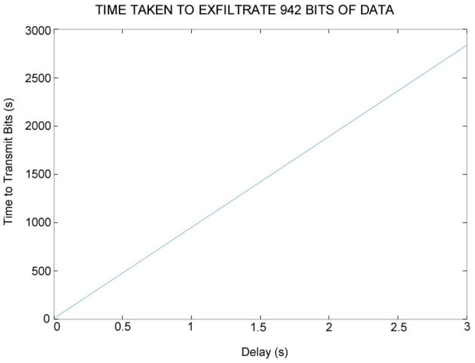


Figure 4. Data Exfiltrated.

TABLE I.

Delay Between Data Packets (s)	Time taken to Exfiltrate 942 bits of data (s)
0	9.21
0.01	18.96
0.05	58.76
0.1	104.95
0.5	481
1	951.92
1.5	1422.7
2	1893.47
2.5	2364.25
3	2835.02

D. Receiver.

In order to decode the data exfiltrated, it is required to place a receiver, for instance a video camera, near or with a line of sight with the visual channel. As mentioned before, it is the attacker's choice whether to use high frequency to send the data in a shorter time or use lowe frequency which needs longer time. It mostly depends on the scenario. However, sending the information at high frequency requires a receiver capable of capturing the exfiltrated data. In average, the default configuration for a video recording camera is 30 frames per second with a resolution of 720 pixels. This configuration allows to receive the exfiltrated sensitive data through the lamps but only when the data is sent at a low frequency. For higher frequencies, it requires a video recorder with slow-motion features. Nevertheless, better quality and more sophisticated features demand more storage capacity.

Figure 5 shows the storage analysis with two different video resolutions. These resolutions were used to record the same 942 bits of data represented in Figure 4. The X-axis represents the time taken to transmit the bits and the Y-axis represents the storage required in Megabytes. Table II shows in detail the amount of storage required for both video resolutions.

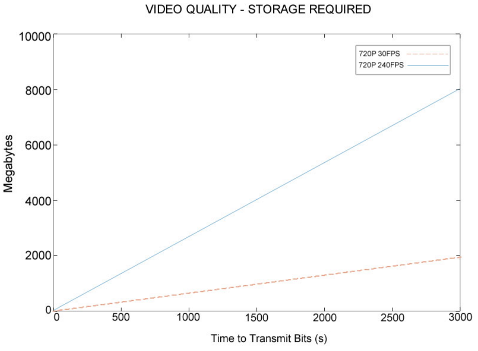


Figure 5. Storage Analysis.

TABLE II.

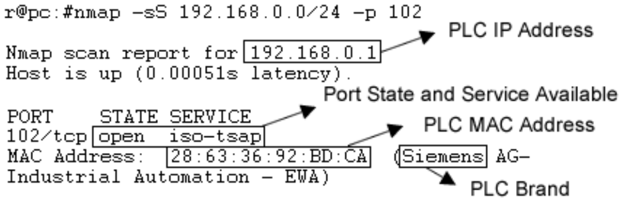
Time Taken to Exfiltrate 942 Bits of Data (s)	Storage in Mb Required at 720p 30fps	Storage in Mb Required at 720p 240fps
9.21	26.09	6.14
18.96	53.72	12.64
58.76	166.48	39.18
104.95	297.35	69.97
481	1362.82	320.68
951.92	2697.07	634.65
1422.7	4030.94	948.51
1893.47	5364.77	1262.38
2364.25	6698.63	1576.25
2835.02	8032.46	1890.11

V. RESULTS

This section describes the operation of our stealthy WaterLeakage malware in addition to the feasibility of this approach.

A. *Network Scan*

Nmap is a lightweight and powerful host discovery tool with a considerable number of features available to use. The malware only uses the TCP SYN Scan feature of nmap because it performs quickly and without raising alarms from security devices placed on the network such as firewalls. Figure 6 shows the command used to execute the network scan, the flag -sS means TCP SYN Scan [33]. The range of IP Addresses scanned are from 192.168.0.1 to 192.168.0.254. The flag -p at the end of the command indicates that the scan considers only the port 102.



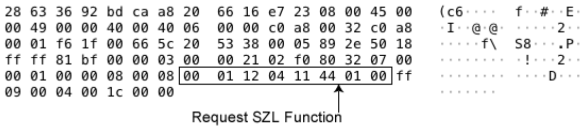
```
r@pc:#nmap -sS 192.168.0.0/24 -p 102

Nmap scan report for 192.168.0.1
Host is up (0.00051s latency).

PORT    STATE SERVICE
102/tcp open  iso-tsap
MAC Address:  28:63:36:92:BD:CA  (Siemens AG-
Industrial Automation - EWA)
```

Figure 6. Network Scan Result.

The result of the scan shows that the IP address 192.168.0.1 satisfy the requirements previously described. The report provided by the NMAP tool shows that the port 102 is open and also the type of service available. It means that we can use ISO-TSAP (International Standard Organization – Transport Service Access Point) to communicate with the PLC. This protocol was designed years ago with no security in mind with the intention to be open and reliable, however, it is not secure at all. Further, the report provides the MAC address of the device and it indicates that it is a Siemens device.

B. *Information Gathering*

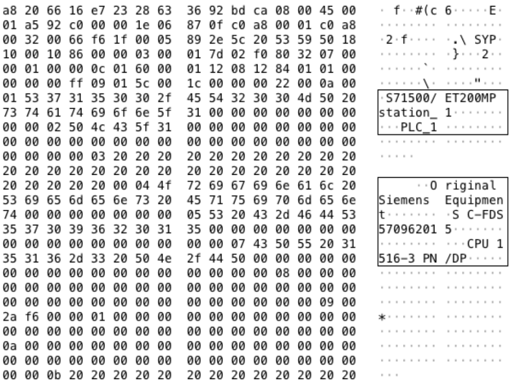
To obtain information from the PLC we craft an ISO-COTP (COTP uses TSAP) packet emulating a connection from an external client [37]. Siemens PLCs do not distinguish between authorized and unauthorized connections, for that reason, it is possible to inquire information pretending to be an authorized client. We use the SCAPY tool [38] to generate the packet. Further, we use the Wireshark tool to monitor the network activity between the Raspberry Pi and the PLC. Figure 7 shows the packet that request information from SZL (System Status List).



```
28 63 36 92 bd ca a8 20  66 16 e7 23 08 00 45 00   (c6····  f··#··E·
00 49 00 00 40 00 40 06  00 00 c0 a8 00 32 c0 a8   ·I··@·@·  ·····2··
00 01 f6 1f 00 66 5c 20  53 38 00 05 89 2e 50 18   ·····f\  S8···.P·
ff ff 81 bf 00 00 03 00  00 21 02 f0 80 32 07 00   ········  ·!···2··
00 01 00 00 08 00 08 00  01 12 04 11 44 01 00 ff   ········  ····D···
09 00 04 00 1c 00 00                               ·······
```

Figure 7. Request PLC Information

The PLC receives, process and send the response with the information available on the SZL. Figure 8 shows the information retrieved from the PLC. The detail of this information is shown in Table III.



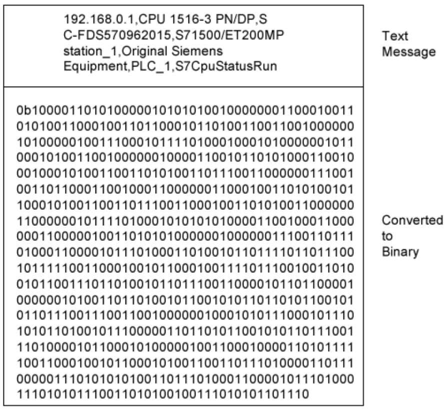
```
a8 20 66 16 e7 23 28 63  36 92 bd ca 08 00 45 00   · f··#(c 6·····E·
01 a5 92 c0 00 00 1e 06  87 0f c0 a8 00 01 c0 a8   ········ ········
00 32 00 66 f6 1f 00 05  89 2e 5c 20 53 59 50 18   ·2·f····  .\ SYP·
10 00 10 86 00 00 03 00  01 7d 02 f0 80 32 07 00   ········ ·}···2··
00 01 00 00 0c 01 60 00  01 12 08 12 84 01 01 00   ·······`· ········
00 00 00 ff 09 01 5c 00  1c 00 00 00 22 00 0a 00   ······\· ····"···
01 53 37 31 35 30 30 2f  45 54 32 30 30 4d 50 20   ·S71500/ ET200MP
73 74 61 74 69 6f 6e 5f  31 00 00 00 00 00 00 00   station_ 1·······
00 00 02 50 4c 43 5f 31  00 00 00 00 00 00 00 00   ···PLC_1 ········
00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00 00 00 00 03 20 20 20  20 20 20 20 20 20 20 20   ·····
20 20 20 20 20 20 20 20  20 20 20 20 20 20 20 20
20 20 20 20 20 00 04 4f  72 69 67 69 6e 61 6c 20        ··O riginal
53 69 65 6d 65 6e 73 20  45 71 75 69 70 6d 65 6e   Siemens  Equipmen
74 00 00 00 00 00 00 00  05 53 20 43 2d 46 44 53   t······· ·S C-FDS
35 37 30 39 36 32 30 31  35 00 00 00 00 00 00 00   57096201 5·······
00 00 00 00 00 00 00 00  00 00 07 43 50 55 20 31   ········ ···CPU 1
35 31 36 2d 33 20 50 4e  2f 44 50 00 00 00 00 00   516-3 PN /DP·····
00 00 00 00 00 00 00 00  00 00 00 00 08 00 00 00   ········ ········
00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00 00 00 00 00 00 00 00  00 00 00 00 00 00 09 00   ········ ········
2a f6 00 00 01 00 00 00  00 00 00 00 00 00 00 00   *······· ········
00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0a 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00 00 0b 20 20 20 20 20  20 20 20 20 20 20 20 20   ···
```

Figure 8. PLC Response

TABLE III.

Tag	Value
Module Type	CPU 1516-3 PN /UP
Serial Number	S C-FDS57096201 5
ASName	S71500/ ET200MP station_1
Vendor	Original Siemens Equipment
Module Name	PLC_1

Moreover, we also collected the CPU Status crafting and sending a new packet to the PLC. It should be considered that with the information collected so far, the attacker might be able to plan a more sophisticated and tailored attack. The PLC brand and model allow obtaining vulnerabilities and exploits that might be already available. This information is finally stored and converted to binary as shown in figure 9. The text message is a sequence of the collected information separated by a comma.

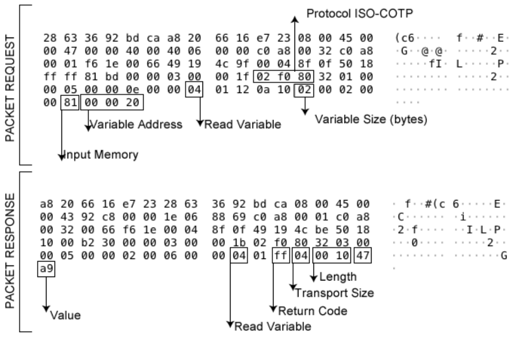


```
192.168.0.1,CPU 1516-3 PN/DP,S
C-FDS570962015,S71500/ET200MP
station_1,Original Siemens
Equipment,PLC_1,S7CpuStatusRun
```

```
0b100001101010000001010101001000000001100010011
010100110001001101100010110100110011001000000
1010000001001110001011110100010001010000001011
0001010011001000000010000110010110101000110010
001000101001100110101001101110011000000111001
00110110001100100011000000110001001101010100101
10001010011001101110011000100110101001100000000
110000001011110100010101010100001100100011000
00011000000100110101010000000100000011100110111
010001100001011101000110100101101111011011100
101111100110001001011000100111101110010011010
010110011101101001011011100110000101101100001
000000101001101101001011001010110110101100101
0110111001110011001000000010001010111000101110
101011010010111000001101101011001010110111001
1101000010110001010000001001100010000110101111
100110001001011000101001100110111010000110111
000001110101010100110111010001100001011101000
111010101110011010100100111010101101110
```

Figure 9. First Message.

Moreover, as Siemens PLCs addresses the Input and Output memory in fixed spaces, it is possible to access these spaces. Figure 9 shows the network packet crafted and the response sent to the PLC. This packet reads the Input space of the memory of the PLC addressed to the ultrasonic sensor (IW4.0). As a result, we obtain the value available at that moment in this space of memory.

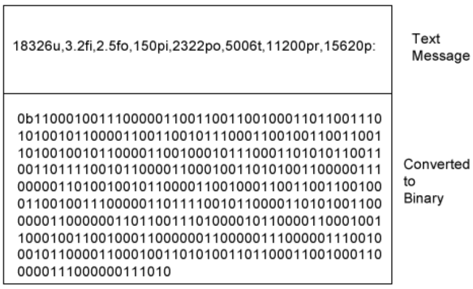


```
28 63 36 92 bd ca a8 20  66 16 e7 23 08 00 45 00   (c6····  f··#··E·
00 47 00 00 40 00 40 06  00 00 c0 a8 00 32 c0 a8   ·G··@·@·  ·····2··
00 01 f6 1e 00 66 49 19  4c 9f 00 04 8f 0f 50 18   ·····fI·  L·····P·
ff ff 81 bd 00 00 03 00  00 1f 02 f0 80 32 01 00   ········  ·····2··
00 05 00 00 0e 00 00 04  01 12 0a 10 02 00 02 00   ········  ········
00 81 00 00 20                                      ····
```

```
a8 20 66 16 e7 23 28 63  36 92 bd ca 08 00 45 00   · f··#(c 6·····E·
00 43 92 c8 00 00 1e 06  88 69 c0 a8 00 01 c0 a8   ·C······  ·i······
00 32 00 66 f6 1e 00 04  8f 0f 49 19 4c be 50 18   ·2·f····  ··I·L·P·
10 00 b2 30 00 00 03 00  00 1b 02 f0 80 32 03 00   ···0····  ·····2··
00 05 00 00 02 00 06 00  00 04 01 ff 04 00 10 47   ········  ·······G
a9                                                  ·
```

Figure 10. Packet request and response to Input Memory IW4.0

Furthermore, we crafted additional packets to read the entire space of the PLC memory addressed to Input and Outputs devices. This information could be useful for an attacker who wants to learn about the control process operation. The PLC used in this paper has six sensors and two actuators connected to the Input and Outputs memory. We collected the information from the Input and Output memory of the PLC as shown in Figure 11. The description of the characters next to the integer values is described in Table IV.



```
18326u,3.2fi,2.5fo,150pi,2322po,5006t,11200pr,15620p:
```

```
0b11000100111000001100110011001000110110011 10
1010010110000110011001011100011001001100 11001
101001001011000011001000101110001101010110011
0011011110010110000110001001101010011000001 11
000001101001001011000011001000110011001100100
011001001110000011011110010110000110101001100
000011000000110110011101000010110000110001001
1000100110010001100000011000001110000011100 10
001011000011000100110101001101100011001000110
0000111000000111010
```

Figure 11. Second Message.

TABLE IV.

Tag	Value
u	Ultrasonic Sensor
fi	Flow In
fo	Flow Out
pi	Pressure In
po	Pressure Out
t	Temperature
pr	Proportional Valve
p	Pump
:	End of Line

C. *Data Exfiltration or Sender*

In the last step, the sensitive information is exfiltrated through a visual channel. It should be noted that in order to succeed in this task the attacker needs to know the internal wiring of the lamps in advance. In this paper, we possess this information. Turning on and off the lamps involves writing 0 or 1to the Output memory of the PLC. To achieve this, we craft and send a packet over the network with that request. Figure 12 shows the Lamps attached to the Festo Rig, they turn on and off depending on the space of memory addressed. The binary message to exfiltrate through the lamps is shown in Figure 9 and it contains 942 values. The lamp Q1 (amber light on the left) represents 0 and the lamp Q2 (amber light on the right) represents 1. To exfiltrate the message we consider three different scenarios.

- The sensitive information is exfiltrated as fast as possible.
- The sensitive information is exfiltrated adding 0.02 seconds of delay between the packets.
- The sensitive information is exfiltrated adding 0.5 seconds of delay between the packets.

D. *Data Reception and Processing*

To receive the message, we place a video recorder camera one meter away the lamps. We record the message using two video resolutions as it is shown in Table V. It is impossible to recover the message when the video camera is recording in normal resolution (720p 30fps) and a delay from 0 to 0.4 seconds is added between the transmission of the packets. Slow motion resolution, for instance 720p 240fps, show better performance for that scenario.

TABLE V.

Message (bits)	Delay Between Data Packets (s)	Time Taken to transmit the packets (s)	Resolution Required	Storage required (Mb)
942	0	9.21	720p 240fps	26.1
	0.02	29.62	720p 240fps	89.92
	0.5	481	720p 30fps	320.67

There are a considerable number of studies for video processing analysis, however in this paper, we apply the background subtractor method [39], [40] which focus in detecting the difference between the current frame and the reference frame extracted from the video recording. An example of this process is shown in Figure 13 where the reference frame is extracted at the beginning of the video. This frame has no lights turned on. The current frame, referenced in the Figure mentioned above as Frame, varies along with the video sequence. The result of the subtraction of both images is transformed to RGB format because it allows to analyse the pixels on this image in detail and to detect whether the lamp turned on is Q1 or Q2. Consequently, we can convert the stream of binary sent through the lamps. Finally, the exfiltrated information is converted from binary to text using the same python module called binascii.

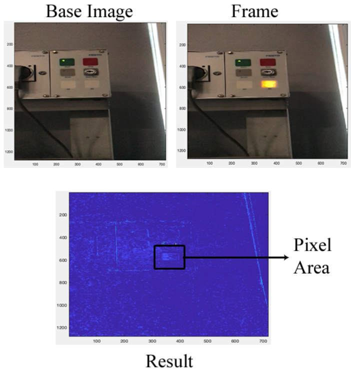


Figure 13. Image Processing, background subtractor.

VI. Conclusion

In this paper, we presented our stealthy WaterLeakage malware capable of collecting and exfiltrating sensitive information from a control process using a covert channel attack. Hackers usually exfiltrate sensitive data in a discrete manner, for that reason, we presented different situations, where the frequency of the light used to exfiltrate the binary message. This depends on the scenario. In some situations, the message might be required to be sent as fast as possible and in others, slower transmission is more suitable. Moreover, it should also be considered that storage capacity in the receiver might be a key point when planning the attack. Further, we highlight that persons with high knowledge and technical skills pose a considerable risk to the company when they are colluding with the attacker.

Additionally, the Input and Output memory of Siemens PLCs are fixed spaces that could be overwritten through the network. It represents a high threat to the control process because the values provided by the sensors connected to the PLC inputs might be manipulated. In addition, the attacker might also compromise the actuators when the Output memory of the PLC is attacked. It should also be considered that PLC models older than the Siemens S7-1200 use fixed spaces for the working memory, as a result, the entire memory could be corrupted and overwritten with values injected by the attacker.

Acknowledgment

This research is supported by the School of Computing and the School of Engineering and the Built Environment of Edinburgh Napier University.

References

[1] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, and A. Hahn, “Guide to Industrial Control Systems (ICS) Security,” 2015.

[2] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly, “Identifying, understanding, and analyzing critical infrastructure interdependencies,” *IEEE Control Syst. Mag.*, vol. 21, no. 6, pp. 11–25, Dec. 2001.

[3] R. S. H. Piggin, “Development of industrial cyber security standards: IEC 62443 for SCADA and Industrial Control System security,” in *IET Conference on Control and Automation 2013: Uniting Problems and Solutions*, 2013, pp. 1–6.

[4] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, “A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications,” *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.

[5] Kaspersky, “What happened to the Internet: attack on Cisco switches,” 2018. [Online]. Available: https://www.kaspersky.com/blog/cisco-apocalypse/21966/. [Accessed: 29-Nov-2018].

[6] H. Abdo, M. Kaouk, J. Flaus, and F. Masse, “A safety / security risk analysis approach of Industrial Control Systems : A cyber bowtie – combining new version of attack tree with bowtie,” *Comput. Secur.*, vol. 72, pp. 175–195, 2018.

[7] IIoT World, “The impact of WannaCry on industrial control systems (ICS),” 2017. [Online]. Available: https://iiot-world.com/cybersecurity/the-impact-of-wannacry-on-industrial-control-systems-ics/. [Accessed: 25-Nov-2018].

[8] R. Langner, “Stuxnet: Dissecting a Cyberwarfare Weapon,” *IEEE Secur. Priv.*, vol. 9, no: 3, pp. 49–51, May 2011.

[9] T. Chen and S. Abu-Nimeh, “Lessons from Stuxnet,” *Computer (Long. Beach. Calif).*, vol. 44, pp. 91–93, 2011.

[10] J. Schlechtendahl, M. Keinert, F. Kretschmer, A. Lechler, and A. Verl, “Making existing production systems Industry 4.0-ready,” *Prod. Eng. Res. Dev.*, vol. 9, no. 1, pp. 143–148, 2015.

[11] F. Khorrami, P. Krishnamurthy, and R. Karri, “Cybersecurity for Control Systems: A Process-Aware Perspective,” *IEEE Des. Test*, vol. 33, no. 5, pp. 75–83, Oct. 2016.

[12] D. Palmer, “GreyEnergy: New malware campaign targets critical infrastructure companies,” 2018. .

[13] B. Genge, P. Haller, and I. Kiss, “Cyber-security-aware network design of industrial control systems,” *IEEE Syst. J.*, vol. 11, no. 3, pp. 1373–1384, 2017.

[14] M. Guri, G. Kedma, A. Kachlon, and Y. Elovici, “AirHopper: Bridging the air-gap between isolated networks and mobile phones using radio frequencies,” in *2014 9th International Conference on Malicious and Unwanted Software: The Americas (MALWARE)*, 2014, pp. 58–67.

[15] M. Guri, M. Monitz, and Y. Elovici, “Bridging the Air Gap between Isolated Networks and Mobile Phones in a Practical Cyber-Attack,” *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 4, pp. 1–25, 2017.

[16] M. Guri, B. Zadov, A. Daidakulov, and Y. Elovici, “ODINI : Escaping Sensitive Data from Faraday-Caged , Air-Gapped Computers via Magnetic Fields,” pp. 1–18, 2018.

[17] M. Guri, B. Zadov, A. Eran, and Y. Elovici, “LED-it-GO Leaking (a lot of) Data from Air-Gapped Computers via the (small) Hard Drive LED,” *Int. Conf. Detect. Intrusions Malware, Vulnerability Assess.*, pp. 161–184, 2017.

[18] M. Guri, “BitWhisper : Covert Signaling Channel be- tween Air - Gapped Computers using Thermal Manipulations.”

[19] M. Guri, Y. Solewicz, A. Daidakulov, and Y. Elovici, “Fansmitter : Acoustic Data Exfiltration from (Speakerless) Air-Gapped Computers,” 2016.

[20] E. Ronen and A. Shamir, “Extended functionality attacks on IoT devices: The case of smart lights,” *Proc. - 2016 IEEE Eur. Symp. Secur. Privacy, EURO S P 2016*, pp. 3–12, 2016.

[21] Z. Zhou, W. Zhang, S. Li, and N. Yu, "Potential risk of IoT device supporting IR remote control," *Comput. Networks*, 2018.

[22] M. Guri, B. Zadov, D. Bykhovsky, and Y. Elovici, "PowerHammer: Exfiltrating Data from Air-Gapped Computers through Power Lines," 2018.

[23] M. Guri, B. Zadov, A. Daidakulov, and Y. Elovici, "xLED: Covert Data Exfiltration from Air-Gapped Networks via Router LEDs," 2017.

[24] S. Chandra, Z. Lin, A. Kundu, and L. Khan, "Towards a Systematic Study of the Covert Channel Attacks in Smartphones," *Int. Conf. Secur. Priv. Commun. Syst.*, pp. 427–435, 2014.

[25] R. Schlegel, K. Zhang, X. Zhou, M. Intwala, A. Kapadia, and X. Wang, "Soundcomber : A Stealthy and Context-Aware Sound Trojan for Smartphones," *NDSS*, vol. 11, pp. 17–33, 2011.

[26] G. Liang, S. R. Weller, J. Zhao, F. Luo, and Z. Y. Dong, "The 2015 Ukraine Blackout: Implications for False Data Injection Attacks," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3317–3318, 2017.

[27] R. P. Foundation, "Raspberry Pi 3 Model B." [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/. [Accessed: 08-Nov-2018].

[28] T. Nash, "Backdoors and Holes in Network Perimeters," vol. 1, no. August, 2005.

[29] A. Ginter, "The Top 20 Cyber Attacks Against Industrial Control Systems," pp. 1–4, 2017.

[30] A. Robles-Durazno, N. Moradpoor, J. Mcwhinnie, and G. Russell, "A supervised energy monitoring-based machine learning approach for anomaly detection in a clean water supply system," in *In Proceedings of the IEEE International Conference on Cyber Security and Protection of Digital Services (Cyber Security 2018)*, 2018, pp. 1–8.

[31] FESTO, "MPS PA Compact Workstation with level, flow rate, pressure and temperature controlled systems." [Online]. Available: https://www.festo-didactic.co.uk/gb-en/learning-systems/process-automation/compact-workstation/mps-pa-compact-workstation-with-level,flow-rate,pressure-and-temperature-controlled-systems.htm?fbid=Z2IuZW4uNTUwLjE3LjE4Ljg4Mi40Mzc2. [Accessed: 07-Jul-2018].

[32] Siemens, "Our fastest controller for automation." [Online]. Available: https://www.siemens.com/global/en/home/products/automation/systems/industrial/plc/simatic-s7-1500.html. [Accessed: 30-Jul-2018].

[33] NMAP, "Nmap: the network Mapper - Free Security Scanner." [Online]. Available: https://nmap.org. [Accessed: 15-Nov-2018].

[34] P. S. Foundation, "18.14. binascii — Convert between binary and ASCII." [Online]. Available: https://docs.python.org/2/library/binascii.html. [Accessed: 13-Nov-2018].

[35] Y. He, M. Rea, A. Bierman, and J. Bullough, "Evaluating Light Source Efficacy under Mesopic Conditions Using Reaction Times," *J. Illum. Eng. Soc.*, vol. 26, no. 1, pp. 125–138, 1997.

[36] N. R. Council, *Virtual Reality: Scientific and Technological Challenges*. Washington, DC: The National Academies Press, 1995.

[37] A. Robles-durazno, N. Moradpoor, J. Mcwhinnie, G. Russell, and I. Maneru-Marin, "Implementation and Detection of Novel Attacks to the PLC Memory of a Clean Water Supply System," in *In CITT 2018*.

[38] Y. Lopes, D. C. Muchaluat-Saade, N. C. Fernandes, and M. Z. Fortes, "Geese: A traffic generator for performance and security evaluation of IEC 61850 networks," *IEEE Int. Symp. Ind. Electron.*, vol. 2015–Septe, pp. 687–692, 2015.

[39] B. Ashwini, B. N. Yuvaraju, A. Y. Pai, and B. A. Baliga, "Real Time Detection and Classification of Vehicles and Pedestrians Using Haar Cascade Classifier with Background Subtraction," in *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, 2017, pp. 1–5.

[40] W. Zhang, L. Xu, Z. Li, Q. Lu, and Y. Liu, "A Deep-Intelligence Framework for Online Video Processing," *IEEE Softw.*, vol. 33, no. 2, pp. 44–51, Mar. 2016.