Out of Control: Ransomware for Industrial Control Systems

David Formby*†, Srikar Durbha*, Raheem Beyah*†
*School of Electrical and Computer Engineering
Georgia Institute of Technology
djformby@gatech.edu, sdurbha6@gatech.edu, rbeyah@ece.gatech.edu
†Fortiphyd Logic
dformby@fortiphyd.com, rbeyah@fortiphyd.com

***Abstract*—Ransomware has recently emerged as the trending new business model for cybercrime with high-profile attacks on hospitals revealing how profitable the technique can be when used to hold operationally critical assets for ransom. Meanwhile, industrial control system (ICS) networks are still struggling to update their security practices due to the perceived absence of threats and rarity of real-world ICS attacks. Recent reports suggest that ICS networks may be the next domain that ransomware is targeting, but to date all attacks have simply used standard ransomware against personal computers with limited effect. In this work, we develop the first known version of ransomware that targets programmable logic controllers, discuss the economic implications of such an attack, and lay out a generic framework for ICS ransomware to aid in future study and defenses.**

I. Introduction

The industrial control systems (ICSs) that make modern day life possible by providing the power to turn our lights on, treating and distributing the water we drink from our faucets, and manufacturing all the material goods we take for granted have so far remained largely untouched by malware, even as data breaches and hacks of enterprise systems have become regularly recurring headlines in the news. The few high-profile ICS attacks that have occurred, namely Stuxnet and the Ukrainian power outages, have been targeted attacks to achieve military-like goals rather than financial gain. However as recent years have shown, the majority of malware authors and bad actors on the Internet are not state-sponsored shadowy arms of the military, but criminals motivated by financial gain who create an entire industry out of selling and distributing malware, botnets, and stolen credit card information.

ICS networks have so far largely avoided being targets of such cybercrime, but not because they are inherently more secure. In fact, according to the rampant vulnerabilities and insecure protocols detailed in a July 2016 report by the Kaspersky Security Intelligence group [4] they are still completely insecure and do not even seem to be improving. The only other explanation for this fragile peace is that cybercriminals have not yet figured out how to translate their operations into a profitable business model for this different environment. In the typical enterprise environment the "crown jewels" that are most important to the victim, and thus the target of any attacker, is the company's data, which explains the huge success of recent ransomware strains. However, in the ICS environment there may be some valuable intellectual property data in manufacturing facilities, but not so much in the power grid, water treatment and distribution, and natural gas utilities. In these areas, a company's "crown jewels" that they care most about are *not* any kind of data, but rather the *continued availability and safe operation of their facilities*. For example, the famous blackout of 2003 that affected the northeastern United States and was caused by a simple software bug, had an estimated economic cost of $7 to $10 Billion dollars [3]. Manufacturing facilities [16] can lose millions of dollars in lost product for every hour of system downtime, and it is difficult to even put a monetary value on the assurance that household faucets will be flowing with clean water, or any water at all.

With these economic aspects in mind, this paper explores why ICS networks are likely the next target for ransomware and we develop the first known example of a cross-vendor, ransomware worm for programmable logic controllers (PLCs), named *LogicLocker*. LogicLocker uses the native sockets API on a Schneider Modicon M241 to scan the network for known vulnerable targets, namely Allen Bradley MicroLogix 1400 PLCs and Schneider Modicon M221 PLCs, and infect them by bypassing their weak authentication mechanisms, locking legitimate users from easily recovering the PLC, and replacing the program with a logic bomb that begins to dangerously operate physical outputs threatening permanent damage and human harm if the ransom is not paid in time.

The main contributions of this work include:

- The first known example of ransomware to target PLCs in industrial control system networks
- The first proof of concept of a cross-vendor worm for PLCs
- A detailed comparison of the economics around traditional ransomware and ICS ransomware
- A survey of devices vulnerable to this kind of attack that are currently discoverable on Shodan
- Explanation of the generic framework for ICS ransomware to aid in future research and defenses

The remainder of this paper is organized as follows. Related work in the area of ransomware and ICS security is presented in Section II. Section III describes the exact threat model that we assume would be implementing this kind of ransomware attack, and Section IV explains the financial incentives of why ICS is the likely next target of ransomware.

A survey of the most popular controllers on Shodan is presented in Section V and the anatomies of LogicLocker and generic ICS ransomware are explained in Section VI. Finally, defenses against such attacks are suggested in Section VIII and conclusions are summarized in Section IX.

II. Related Work

Ransomware has evolved significantly over the years as malware authors have learned and adapted their methods of distribution, approach to extorting victims, and the technology that they use to hold assets for ransom. The first known example of ransomware was observed in 1989 when the AIDS Trojan was distributed through paper mail on floppy disks. Despite the novelty of the attack method, the campaign was largely unsuccessful for a variety of reasons including inefficient distribution, small pool of targets, inaccessibility of widespread strong encryption, and trouble with international payments.

In the Symantec report on the evolution of ransomware [13], ransomware is categorized into four main types and key points in history are identified when malware authors pivot between them. The first epidemic of ransomware appeared in the form of fake applications promising to fix imaginary problems in the victim's computer for a small fee. Next, attackers increased the supposed threat to the victim by posing as fake antivirus programs promising to clean out all the infections it found with the free scan. As the average user became more tech-savvy and better at spotting these scams, attackers became even more aggressive and began locking users out of their computers usually under the pretense of some law enforcement agency forcing them to pay a fine for piracy. Again, average users began to learn how to detect these scams and legitimate security products were released to restore victim's computers without paying the ransom. To compensate for all the previous weaknesses in ransomware, attackers finally moved to cryptoransomware to have stronger control over the victim's valuable assets and gave up trying to deceive the users at all, instead opting to overtly demand payment or their data would be destroyed.

It is this cryptographic breed of ransomware that has been making the headlines recently by extorting large payouts ranging from $10,000 to restore a public school district's records to $17,000 to restore patient records at a hospital [15]. The FBI estimated ransomware cost the United States a total of $24 million in the year 2015 with the cost flying up to an estimated $209 million in just the first three months of 2016 [8]. With cybercriminals clearly catching on to the profitability of ransomware, they appear to be starting to look into ICS networks as their next potential victims. In a report released by Fortinet [12] this year, evidence shows that attackers seem to be retooling their standard cryptoransomware and targeting them at manufacturing facilities in particular. Another report from Booz Allen Hamilton [5] takes this concept one step further by speculating that ransomware will not just shift to the manufacturing sector, but also shift to attacking the PLCs themselves rather than the personal computers. As further evidence for the truth of these predictions, the world is seeing high profile ransomware attacks creep ever more closer to control systems. In November of 2016, the ticketing machines at San Francisco's Muni transit system were infected with ransomware, taking several days to recover and allowing passengers to ride free during the busy Thanksgiving holiday weekend [9]. Then, in late January of 2017, it was reported that ransomware infected a luxury Austrian hotel, preventing new room key cards from being programmed for guests and essentially locking them out of their rooms. With each hotel room costing upwards of several hundred US dollars per night, the victim decided it was worth paying the attacker roughly $1,600 to restore the systems and continue normal business operation quickly [7].

Although there haven't yet been any known ransomware attacks on PLCs, there have been other high-profile attacks on control systems. The first known example of malware to target PLCs was Stuxnet, which was discovered in 2010 to have reprogrammed the PLCs controlling Iranian centrifuges and destroyed them by tampering with the rotation speeds [11]. Other proof of concept attacks have been demonstrated at Black Hat conferences that turned PLCs into port scanners [10] with self-propagating worms [14].

III. Threat Model

The threat model that this paper suggests for ransomware is that of organized cybercrime rather than state sponsored attackers or unskilled script kiddies. The distinction is important to make in terms of likelihood of attack, motivations and goals of the attack, and the skill level of the attackers. As a general rule in network security, the sophistication of attackers is inversely proportional to the frequency of attacks, where unskilled attackers simply reusing popular exploit code make up most attacks followed by professional cybercriminals and then the rare state sponsored attacks. In this paper, the attackers are assumed to be cybercriminals with enough skill to compromise a PLC at the application layer, but either lack the skills, resources, or motivation to develop exploits at the firmware level for each model PLC they want to attack. By keeping the attack at the application layer, they are able to target a wider range of victim devices and thus increase their expected profits.

The high level goal of the cybercriminal attacker is to cause the victim enough lost revenue from system downtime and threaten enough damage to personnel and expensive equipment as to make paying the ransom more attractive than other means of restoring the facility to operation. He achieves this goal by stealing the original PLC program, locking down the PLC using the methods described in Section VI-C, encrypting the original program, and adding a logic bomb in the PLC code that will start dangerously operating outputs if the ransom is not paid in time. Furthermore, this last goal will take two forms depending on the knowledge level of the attacker. In the first case, an attacker has little to no knowledge of the underlying physical process behind the control system, and will erratically operate all outputs in hopes of causing physical damage. In the second case, a more knowledgeable attacker who has performed reconnaissance on the physical process will know exactly which outputs to operate and can intelligently try to move the system into a more vulnerable state before triggering the logic bomb.

IV. Economics of Ransomware

One of the main reasons why ransomware attacks have been so successful against hospitals, is the absolute need for patient's data to continue operating the hospital and providing patients with care. Industrial control systems suffer from similar needs by absolutely needing *control* of their PLCs to continue operation of their systems providing power to people's homes or manufacturing their product.

A. Traditional vs ICS Ransomware

Traditional ransomware attackers have demonstrated careful consideration of the demanded payment depending on how valuable the data is they have stolen and the population of victims they are targeting, carefully balancing Equation IV-A to ensure they remain profitable. On one end of the spectrum, the typical Internet user who gets his photos and personal documents encrypted will be asked for a payment in the hundreds of dollars. Considering the cost or effort required for attackers to launch the attack, it is relatively inexpensive to send out mass phishing emails that trick a number of victims into downloading the ransomware payload. The importance of the data is not usually life threatening, so only about half of the victims end up paying the ransom [6]. Even so, attacks like this are known to be profitable for attackers. So when attacking a smaller pool of more specific targets, the campaign will only be successful if the value of whatever the attacker is holding hostage makes up for the smaller population size.

$$Profit = Population * Value - Cost \tag{1}$$

On the other end of the spectrum are companies whose data is imperative to continue business operations as normal. There are fewer targets for the attacker to compromise, but the stakes are higher and to balance out the tradeoff, he can ask for ransoms in the thousands of dollars, still resulting in profit. Depending on the target, the cost to the attacker to compromise the victim ranges from high, for business with strong security practices, or low for unprepared business. By targeting networks with traditionally weak security practices, such as hospitals, schools, and ICS networks, the attacker keeps his costs low and profit margins high.

Industrial control systems represent a relatively small pool of targets, so whatever assets an attacker holds for ransom must be valuable enough to still balance out the tradeoff equation in his favor. ICS networks usually have little valuable data, but instead place the highest value on downtime, equipment health, and safety to personnel. Therefore, ransomware authors can threaten all three to raise the value side of the tradeoff equation to make ICS ransomware profitable.

Downtime. Depending on the victim's industry, downtime can have minor or catastrophic effects on profits. As stated before, car manufacturers were estimated to suffer millions of dollars in lost revenue for every *hour* of downtime and large scale blackouts can climb up into several billion dollars in costs. A successful ransomware attack on an ICS network will threaten the victim with an unacceptable duration of downtime by halting operation, and hampering restoration efforts by stealing the program on the PLC and locking users out. Furthermore, if certain processes like perishable food manufacturing are interrupted by the ransomware, the victim facility suffers extra downtime due to having to flush out the system, resanitize the equipment, and reboot the entire process after recovering the PLCs.

It is important to note here that many facilities, especially smaller businesses, do not program their own PLCs and instead pay third party Original Equipment Manufacturers (OEMs) or system integrators to do the programming for them. Therefore, once a PLC is infected, restoration is not a simple matter of reflashing the firmware from scratch and reloading the old program. In some cases the programs written by the third parties are proprietary software that they want kept secret, meaning the victim facility will not have a backup copy of the program. The victim will then have to reach out to their OEM or integrator and schedule maintenance to fix the PLC(s), likely adding several days of downtime in the process. Furthermore, the possibility of a restoration effort also assumes that the parties who did the programming are still in business and keep accurate backups of all programs. Finally, the more PLCs the attacker can compromise, the longer he can extend the recovery process causing more downtime and increasing the attractiveness of his offer for immediate restoration.

Equipment Health. One of the unique characteristics of ICS networks is that attackers can also interact with the physical world of the network, for example speeding up centrifuges or opening valves to spill hazardous chemicals. Damaged equipment in ICS facilities can not only be expensive to find replacements or make repairs, the downtime required to get the system back online adds even more costs due to lost production. A successful ransomware attack will include logic bombs in the PLC code that begin operating the connected machinery with the goal of causing physical damage to the equipment. The effectiveness of such a logic bomb can be increased by performing network reconnaissance to try to understand the underlying physical process, moving the system into a vulnerable state, and intelligently operating the outputs to cause the most damage.

Human Safety. One of the lessons attackers are learning from the hospital ransomware attacks is that when human beings are on the line, companies are willing to pay the higher price to ensure their safety and health. Therefore the logic bomb that threatens equipment health also adds even more value to the attack by threatening the safety of any nearby personnel.

Table I summarizes and compares the analogous aspects between the recently popular strains of cryptoransomware and the proposed ICS ransomware. Again, we want to highlight the fact that current cryptoransomware has partly been profitable because most victims do not regularly backup their data, and thus can't simply wipe their computer and restore the data. Similarly, it is not clear how well PLC programs are backed up, and the restoration procedure if a backup is available likely involves more costly downtime and possible equipment repairs.

V. Vulnerability of Targets

Besides the critical nature of the data held for ransom, the other reason why hospitals have been such a prime target for ransomware is the traditionally weak security posture. Again,

TABLE I: Comparison of Traditional and ICS Ransomware

	Traditional	ICS
Target	Data	Safety, Operation
Method	Encryption	Logic Bomb, Downtime
Restoration	Data backup	Program backup, repairs

TABLE II: Examples of Known Devices Susceptible to Password Locking Ransomware

PLC Model	Shodan Count	Tested
MicroLogix 1400	1429	In lab
Schneider Modicon M221	250	In lab
Siemens S7-1200	167	Other [14]

ICS networks both at the vendor level and facility level suffer from a similar flawed philosophy of not treating malware as a realistic threat, and thus their networks are wide open for compromise.

A. *Survey of Shodan Devices*

Kaspersky Security Intelligence performed a July 2016 survey of devices discoverable on Shodan and focused on the surprising number of known vulnerabilities they found [4]. To complement this study, we performed our own brief survey of devices we currently know would be susceptible to the type of ransomware attack described here. Two devices from popular vendors were tested in our lab, and a third was proven susceptible in a presentation at Black Hat [14]. Table II summarizes the findings to show how large the current attack surface is for any attacker wishing to begin a ransomware campaign on Internet-facing PLCs. This only represents a small portion of the total potential attack surface since there are tens of thousands more PLCs attackers can target after first compromising devices on victim corporate networks. Note that the three devices presented here are from three of the most popular vendors of PLCs. Documentation on how vendors actually implement their password authentication is difficult to find, but the survey here suggests this is a common poor practice across all vendors.

B. *Experimental Setup*

The PLCs, illustrated in Figure 1, that were used for this proof of concept version of LogicLocker included a Schneider Modicon M221, an Allen Bradley MicroLogix 1400, and a Schneider Modicon M241. Allen Bradley and Modicon represent some of the most popular PLC brands in the world, but we want to stress however that it has been public knowledge for years that most PLCs do not properly authenticate programming log-ins. Similar attacks could just as easily been constructed for other major vendors, for example Siemens, as illustrated in a Black Hat presentation [14].

Fig. 1: PLCs used in the proof of concept testbed

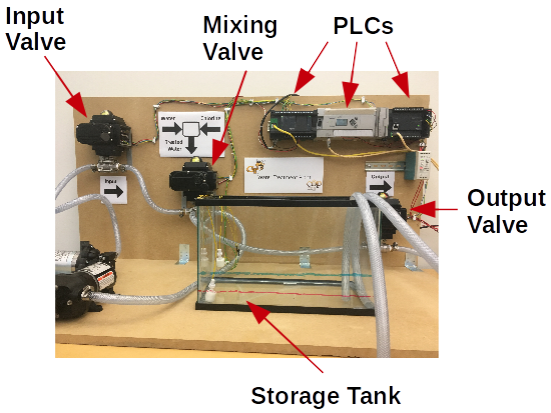

Fig. 2: Water Treatment Testbed

As an example, some of the security features that the MicroLogix 1400 PLC offers are password protection (from the programming software) and an OEM Lock mechanism that requires the user (from the programming software) to already have a PLC program with a matching 16-bit checksum before logging in and reading it, to protect proprietary PLC programs from curious users. Other noteworthy features include an email client to send notifications to operators and a generic socket interface to communicate using non-standard ICS protocols. Finally, to address the cybercriminal economic aspects of this choice, there are currently over 1,400 of this model device discoverable on Shodan. If an attacker were to use LogicLocker to compromise all 1,400 devices and ask for a ransom on the same order of magnitude as the school and hospital ransoms around $15,000, the attacker would be earning up to $21 million from this attack alone. With minor modifications he can likely reuse the same code to target other models in the same family of PLCs as well to increase his earnings. Again, note that this count is only considering the MicroLogix 1400 PLCs directly connected to the Internet. A single large manufacturing facility can have hundreds of PLCs on their control network which the attacker can vertically move to through first compromising the corporate network, adding up to tens of thousands of total potential targets.

To illustrate how a ransomware attack might happen, a small scale testbed, labeled in Figure 2, was built to mimic the disinfection stage and storage stage of a city water treatment facility. In the disinfection stage, precise ratios of input water is mixed with chlorine. In the storage stage, the mock up facility keeps a minimum amount of reserved water to ensure demand can always be met.

VI. Anatomy of Attack

The basic steps in the proposed ICS ransomware attacks include the initial infection, optional step of movement, locking and encrypting, and finally negotiation for the ransom. The timeline of the entire process is illustrated in Figure 3, with Tables III through VII listing the options that the attacker has at his disposal at each step.

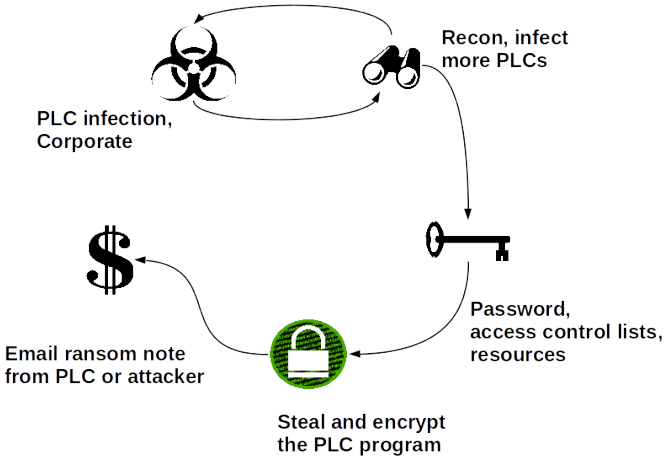

Fig. 3: General Flow of ICS Ransomware Attack

TABLE III: Approaches to Infection

	Internet-facing PLCs	Corporate Network
Advantages	Easier targets	More targets
Disadvantages	Fewer targets	More effort

A. Initial Infection

As illustrated in Section V, there are currently thousands of PLCs directly connected to the Internet and easily discoverable on Shodan. In the simplest case, an attacker can directly target one of these Internet-facing PLCs. In a more complex case, an attacker can use standard malware and tactics to first infect a workstation on the victim's corporate network and then use that machine as a stepping stone into the control network if there is not proper segmentation. While this may require more effort, this mirrors the standard attack method in IT networks of compromising weak devices and pivoting inside the network.

For compromising the PLC itself, it is widely known that many PLCs do not provide strong authentication of new programs being loaded onto them or in the best case only disable remote programming from the network. In this case, the attacker would have to invest more time and effort into finding a vulnerability to exploit that would provide him with programming access. However, given the always-on nature of ICS devices, and the Shodan survey results, devices appear to go long periods of time with known vulnerabilities. Furthermore, the number of vulnerabilities being reported to ICS-CERT is not slowing down suggesting that there are still plenty of vulnerabilities to discover. Cybercriminals wanting to create an industry out of ICS ransomware could feasibly begin finding vulnerabilities on their own to exploit for their ransomware attacks. The different approaches to infecting PLCs with the ICS ransomware are summarized in Table III.

B. Horizontal and Vertical Movement

An attacker can increase his expected profits by instead of just infecting one PLC, moving throughout the victim's network either horizontally, vertically, or both in terms of the Perdue reference model, Figure 4 [17]. Horizontally, the attacker can maximize his profits by infecting as many PLCs in the victim's facility as he can, spreading throughout Level

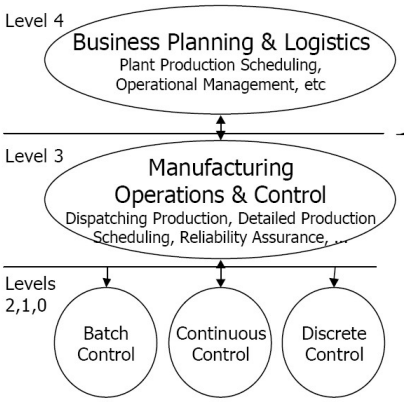

Fig. 4: Purdue Enterprise Reference Architecture

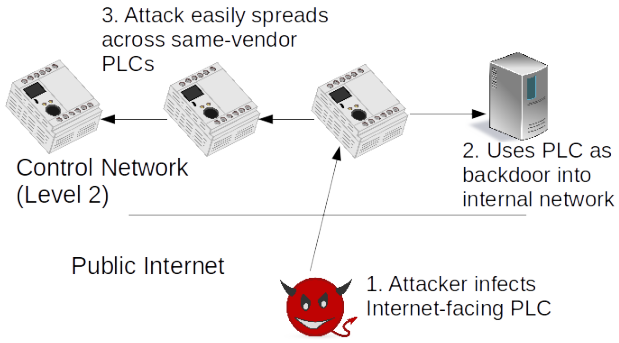

Fig. 5: Attacker using PLC for horizontal movement within the internal network

1 of the reference model, illustrated in Figure 5. The more PLCs he can compromise, the stronger a grip he has on the "crown jewels" of the victim's operations and higher ransom he can ask for. Whereas if he infects just one PLC, he risks the chance that the facility can still run in some limited capacity with manual labor or they have a backup PLC they can swap out quickly. In fact, such a horizontal attack would be the most profitable and easily accomplished due to the high probability of the existence of multiple PLCs of the same model (as in Stuxnet) and other PLCs from the same vendor with shared vulnerabilities. To even further strengthen his hold on the victim, he can try moving vertically in the network and attacking the human machine interfaces (HMIs) or engineering workstations with standard malware in the hopes of strengthening the persistence of the ransomware or of stealing the backup copy of the PLC programs.

Vertical movement can be achieved starting at the PLC or the corporate network. If the initial infection was through an Internet-facing PLC and the PLC supports generic sockets like the MicroLogix 1400, he can use the PLC as a backdoor into the rest of the network. If the initial infection was through the corporate network like most real infections are, he can first perform reconnaissance, steal any valuable data, and wait to shut down operations until he can compromise multiple PLCs at a time, as illustrated in Figure 6.

Table IV summarizes the advantages and disadvantages of the two approaches to movement throughout a victim network.

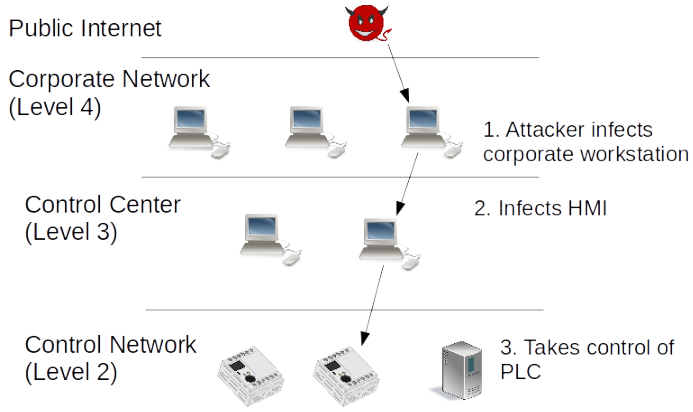

Fig. 6: Attacker using vertical movement to compromise PLC from the corporate network

TABLE IV: Approaches to Movement

	Horizontal	Vertical
Advantages	Less security	Performing recon
	More control	Standard malware
Disadvantages	Less recon	More time

To maximize effectiveness the attacker can also choose to use both methods.

C. Locking

Since most of the success of an ICS ransomware attack relies on the attacker holding the operation of the facility hostage, he must ensure that access to the PLC is locked down to prevent quick restoration. Depending on the features available on the PLC, he may take a combination of several approaches.

The simplest approach would be to change the password on the PLC to a strong random password of the attacker's choosing. As stated before, the password authentication many PLCs offer is really only checked in the programming environment, not the PLC. Therefore, the mechanisms designed to protect the victim's PLC ironically prevent him from recovering it with the legitimate software while doing nothing to stop the attacker. The victim can of course write his own custom software to regain access to the PLC, but with the likely disapproval of the legitimate software vendors. Following this same ironic logic, the attacker can then attempt to use all other security features of the PLC against its rightful owner to hamper restoration as much as possible. Other security features could include access control lists for IP addresses and enabling the OEM lock for protecting proprietary PLC programs.

If the PLC does happen to offer password protection that is checked on the PLC side, the slow embedded nature of the PLC means brute forcing even weak passwords over the network can take an infeasible amount of time. For example, the MicroLogix 1400 used in this research has a round trip time (RTT) of approximately 1ms over an Ethernet cable less than 1m long, meaning the majority of that RTT is processing the ICMP echo request. Assuming a best case scenario that a password check is as fast as an echo reply, the time it would take to brute force recover a 6-digit alphanumeric password, which would normally be considered a trivially weak password, would be roughly 657 days.

TABLE V: Approaches to Locking

Approach	Strength	Weakness
Password	Easy	Usually bypassed
OEM Lock	Easy	Usually bypassed
Using up resources	Stronger lock	Harder to implement
Changing IP/ports	Easiest	Easy to find

Other means of preventing recovery that are not as strong could include reading status registers in the PLC, e.g. current active TCP connections, to monitor for recovery attempts and deciding to harm the attached equipment if detected. Alternatively, the attacker can try locking down the PLC by taking advantage of the limited resource environment. Many PLCs have a maximum number of active TCP connections they can handle, so an attacker can either remotely use up all active connections or potentially use a generic socket interface to create several TCP connections to the localhost address. Finally, to add more confusion and downtime to the attack, the ransomware can change the IP address of the PLC and the port numbers it is listening on.

D. Encryption

Even if the victim regains programming access to the PLC by successfully bypassing the locking mechanisms in the previous section or reflashing the PLC, the attacker has already removed the original program from the PLC. If the attacker wanted to follow the traditional steps of cryptoransomware, he would leave a copy of the encrypted program on the PLC. However, whatever program he writes to the PLC must still be executable without crashing or else restoration after payment is not feasible. Therefore, the attacker can take one of three approaches.

The simplest approach would be to use standard encryption schemes on the attacker's machine over the raw binary of the program and email it to the victim along with the ransom note. Then, when the victim pays, send them a tool to decrypt the binary and load the program onto the PLC using the same technique the attacker used to load his own program. While this is the easiest approach and assumes the attacker has set up a command and control connection with the PLC, the victim may have a hard time trusting that the attacker will restore the PLC's program. So to be more persuasive, the attacker can take a second approach and be creative with how he leaves the program on the PLC in an encrypted form. The easiest way of accomplishing this approach would again be to encrypt the raw binary of the program using standard encryption schemes, but store it in the data section of the PLC's memory as raw strings. However, depending on the capabilities of the PLC, it may not have enough data memory to store the entire PLC program and the attacker's logic bomb program. So the attacker may take a third and more complex approach of encrypting the program while still ensuring that it is executable.

This third approach would involve using a secret key to randomly change the content and control flow of the program to result in dangerously unpredictable operation and infeasible

TABLE VI: Approaches to Encryption

Approach	Strength	Weakness
AES, email	Easiest	Assumes C&C
AES, on PLC	More persuasive	Not always possible
Encoding, on PLC	More persuasive	More complex

recovery of the original program. First, to change the content of the program the objective of the attacker will be to randomly change variables critical to the correct operation of the control system, such as timer and counter configurations. This could easily be accomplished by encrypting the original values and truncating them to the necessary length, for example a 16-bit counter. Second, the attacker can replace instructions with their syntactic equivalents (e.g. ANDs with ORs, additions with subtractions) to ensure the program is still executable. While this may not be the most secure or efficient method, one easily understood way to achieve this would be to first separate all instruction bytecodes in the machine language of the PLC into groups that can be syntactically substituted for each other and place each group into a circular queue. Then, the attacker chooses a strong random seed value and uses a pseudorandom number generator to generate a random queue rotation for every instruction in the program and replace each instruction with whatever equivalent instruction is at the head of the circular queue after the random number of rotations. This achieves the act of syntactically equivalent "encryption". Note that the encryption used to hold the program for ransom does *not* even have to be considered secure under industry standards, it merely has to slow down recovery enough to make paying the ransom more attractive than a recovery attempt. To "decrypt" such a program to the original, the attacker simply regenerates the random number sequence and rotates each instruction group queue in the opposite direction for the random number of rotations. To make recovery even harder to achieve, the attacker can employ similar reversible techniques to add arbitrary code to the PLC and shuffle the order of the instructions.

E. Negotiation

Notifying the victim of compromise using the limited resources on the PLC is also nontrivial. In the simplest case, an attacker can send an email separately notifying the victim of the compromise and demanding payment. However, to make a stronger show of force, the attacker may also leverage the PLC to deliver the ransom note. Some PLCs, as in the MicroLogix model used for LogicLocker, have embedded email clients that are normally used to automatically send alerts to operators. The attacker can reprogram the PLC to instead send the ransom note to the victim directly from the victim's own PLC. Other means of notifying the victim could include changing the PLC's web interface.

From the arguments laid out in Section IV, a successful ransomware attack should not only lock out users from accessing the PLC, but also threaten damage to expensive machinery and personnel. The attacker factors this into negotiating for a ransom by explaining to the victim that if payment is not received before a certain deadline, the program will be deleted and the logic bomb in the PLC will begin destroying equipment. To make this logic bomb even more

TABLE VII: Approaches to Negotiation

Approach	Strength	Weakness
Attacker Email	Easiest	Not as persuasive
PLC Web Page	More persuasive	Not always possible
Email from PLC	More persuasive	Not always possible

TABLE VIII: Anatomy of LogicLocker

Stage	Action
Initial Infection	Direct,Bypass password
Movement	Worm
Locking	Change password, OEM lock
Encryption	Remote
Negotiation	Email from attacker

effective, the attacker can perform reconnaissance to gain a basic understanding of the physical devices behind the control system so he can know the best way to cause harm. This can be achieved by studying the user customizable web interface available on most PLCs or by stealing data from the engineering workstations on the network. Once the victim pays the ransom, the attacker either remotely reprograms the PLC to the original functionality or sends the victim a tool to do so.

VII. LogicLocker

The proof of concept attack developed here for the testbed illustrated in Figure 2 takes the simpler approaches to the steps in the ransomware cycle. First, it is assumed that an attacker has either brute forced a weak password on an Internet-facing Modicon M241 or stolen legitimate credentials, and has loaded it with LogicLocker. LogicLocker then scans the internal network for vulnerable PLCs to infect further. The primary locking aspects of LogicLocker are achieved when the vulnerable PLCs, Modicon M221 and MicroLogix 1400, are reprogrammed with new passwords, locking legitimate users out of the official programming software. For the encryption stage, the attacker manually encrypts the stolen program on his own machine using standard encryption and a key generated for this victim. In the negotiation stage the attacker using LogicLocker sends an email from his own computer to the victim notifying them of the compromise. If the ransom is paid by the ultimatum, the attacker gives the victim a program that will reload the original programs, but if it is not paid he threatens to dump harmful amounts of chlorine into the water supply. To maximize chances of success, before notifying the victim of compromise LogicLocker first allows the level of the water in the storage tank get low while sending false level readings to the operators. Therefore, given the choice between paying and attempting a recovery, the victim also has to consider the effects of waiting too long and running completely out of clean water. Future versions of LogicLocker will use the PLC's own email client to send this ransom note. Finally, once the victim pays, the attacker sends the victim a tool that decrypts the original PLC program and reloads it on to the victim PLC. Table VIII summarizes the pieces of LogicLocker, describing each of the general steps in an ICS ransomware attack. Video demonstrations explaining the setup [2] and the attack [1] can also be found online.

VIII. Defenses

Most ICS equipment vendors refuse to provide common sense security on all their devices, and place the burden of security on the end user by suggesting they rely on the fallacy of air-gap security or additional equipment. However, following standard best practices can substantially reduce the risk of falling victim to the cybercriminal threat model described here. Since true air gaps rarely exist, defense-in-depth strategies should always be implemented.

Endpoint Security. Defense-in-depth strategies at the endpoint level would include changing all default passwords, disabling all protocols that are not critical for operation, using access control lists where possible, disabling remote programming, keeping device firmware up to date, and backing up all program files. When purchasing new equipment, carefully consider the security features of the product, keeping in mind that the password protection that many PLCs advertise really only authenticate users of the programming environment, not attackers with their own code.

Network Security. At the network level, the architecture should be segmented and both the IT network and the control network should be monitored for suspicious activity and protocol whitelisting should be implemented in firewalls. Control network topologies are usually static and devices are rarely reprogrammed without advanced planning to minimize system downtime. Using this insight, anomalies can be detected when different IP addresses begin communicating or reprogramming events are observed that do not match with planned maintenance. Furthermore, automated backups of PLC programs can help expedite recovery of the victim facility without having to pay the attacker. Future work inspired by this research will investigate remote software attestation techniques to determine malicious changes to PLC programs.

Policy. At the end user level, all employees should be trained to identify phishing emails and prohibited from using their own personal USB drives to reduce the risk of initial infection. Furthermore, facilities should have an incident response plan in place to put into action when a compromise does happen, and practice it in a safe environment to be prepared. This response plan could involve keeping backups of critical programs on the premise and having personnel trained in how to reflash and restore PLC programs quickly. If intellectual property issues prevent the the facilities from having a copy of the program, develop relationships with OEMs to provide emergency response services to restore the PLCs.

IX. Conclusion

ICS networks have so far remained largely unscathed by malware not because they are more secure than traditional networks, but because cybercriminals have yet to figure out a profitable business model to make such attacks worth their time. Recent attacks on hospitals have demonstrated how profitable ransomware can be when used to hold operationally critical assets hostage with the threat of human harm, and reports suggest attackers are beginning to shift their focus on ICS networks. In the hopes of preparing ICS operators for the likely coming wave of attacks, this research developed the first known ransomware to target PLCs in order to study the difficulties in attacking and defending ICS networks from these attacks. Ironically, we found that the weak security mechanisms that many of the most popular vendors provide on their PLCs actually do more harm to legitimate users in the context of ransomware than they do to protect against attacks. In future work, we will continue to investigate more sophisticated means of locking PLCs in order to suggest further defenses, as well as developing techniques to remotely detect when PLC programs have been changed.

References

[1] Plc ransomware worm demo. https://youtu.be/t4u3nJDXwes.

[2] Water treatment testbed. https://youtu.be/KTKRjvTgTQI.

[3] The economic impacts of the august 2003 blackout. Technical report, Electricity Consumers Resource Council, 02 2004.

[4] The evolution of ransomware. Technical report, Kaspersky Security Intelligence, 07 2016.

[5] Industrial cybersecurity threat briefing. Technical report, Booz Allen Hamilton, 06 2016.

[6] D. Bisson. Half of american ransomware victims have paid the ransom, reveals study. http://www.tripwire.com/state-of-security/latest-\\security-news/half-of-american-ransomware-victims-have-\\paid-the-ransom-reveals-study/.

[7] M. Burgess. Could hackers really take over a hotel? wired explains. http://www.wired.co.uk/article/austria-hotel-ransomware-true-doors-lock-hackers.

[8] D. Fitzpatrick and D. Griffin. Cyber-extortion losses skyrocket, says fbi. http://money.cnn.com/2016/04/15/technology/ransomware-cyber-security/.

[9] T. Fox-Brewster. Ransomware crooks demand $70,000 after hacking san francisco transport system – updated. https://www.forbes.com/sites/thomasbrewster/2016/11/28/san-francisco-muni-hacked-ransomware/\#53cc57247061.

[10] J. Klick, S. Lau, D. Marzin, J.-O. Malchow, and V. Roth. Internet-facing plcs - a new back orifice, 2015.

[11] R. Langner. To kill a centrifuge. Technical report, The Langner Group, 11 2013.

[12] B. McGee. Move over healthcare, ransomware has manufacturing in its sights. https://blog.fortinet.com/2016/06/06/move-over-\\healthcare-ransomware-has-manufacturing-in-its-sights.

[13] K. Savage, P. Coogan, and H. Lau. The evolution of ransomware. Technical report, Symantec, 08 2015.

[14] R. Spenneberg, M. Brggemann, and H. Schwartke. Plc-blaster: A worm living solely in the plc, 2016.

[15] H. Taylor. Ransomware: Lucrative, fast growing, hard to stop. http://www.cnbc.com/2016/04/11/ransomware-lucrative-fast-growing-hard-to-stop.html.

[16] E. Vadala. Downtime costs auto industry $22k/minute - survey. http://news.thomasnet.com/companystory/downtime-costs-auto-industry-22k-minute-survey-481017.

[17] T. J. Williams. The purdue enterprise reference architecture: a technical guide for cim planning and implementation. 1992.