Volatile Memory Extraction-Based Approach for Level 0-1 CPS Forensics

Rima Asmar Awad
Cyber and Data Analytics Division
National Security Sciences Directorate
Oak Ridge National Laboratory
Oak Ridge, TN, USA
awadrl@ornl.gov

Mike Rogers
Computer Science Department
Collage of Engineering
Tennessee Technological University
Cookeville, TN, USA
mrogers@tntech.edu

Juan Lopez Jr.
Cyber and Data Analytics Division
National Security Sciences Directorate
Oak Ridge National Laboratory
Oak Ridge,TN, USA
lopezj@ornl.gov

Kalyan Perumalla
Comp. Sci. and Math. Division
Computing and Comp. Sci. Directorate
Oak Ridge National Laboratory
Oak Ridge, TN, USA
perumallaks@ornl.gov

***Abstract*—Most security analyzers operate on system state that is far removed from end-point components in cyber-physical systems (CPS) identified as level 0-1 in the Purdue Architecture Reference Architecture (PERA) [1]. For example, many operate on system logs and other data dumps to disks. Tremendous value that can be gained in cyber security forensics if low level details such as dynamic changes to volatile memory can be extracted and provided to more sophisticated analysis tools. However, obtaining detailed and dynamic system state at the level of volatile memory is extremely challenging [2]. Here, we attempt to apply IT memory forensic mechanisms to CPS end-point devices and statistically evaluate them. Our focus is to extract volatile and dynamically changing internal information form CPS 0-1 level devices, and design preliminary schemes to exploit that extracted information. This new capability of generating a sequence of volatile memory snapshots for offline, detailed and sophisticated analysis opens a new class of cyber security schemes for CPS forensic analysis. As a case study for our ongoing research, we apply the proposed methodology to Modicon PLC using Modbus protocol. We extract the memory layout and subject the device to read operations at the most critical regions of memory. Similarly, write operations are initiated to carefully determine memory locations (for example, bytes that hold the firmware version number). This capability of generating a sequence of volatile memory snapshots for offline, detailed and sophisticated analysis opens a new class of cyber security schemes for CPS forensic analysis. Also, the ability to dynamically make controlled modifications to specific memory locations opens the potential for new mechanisms such as taint analysis and watermarking.**

***Index Terms*—CPS, Forensics, Cyber Attacks, Modbus, Watermarking, Memory.**

I. Overview

Cyber-Physical Systems (CPS) that integrate computation, networking, and complex physical processes, and play an important role to ensure the proper operation and functionality of critical national infrastructure. Researchers have made major breakthroughs and advances in computer science (e.g. programming languages, real-time computing techniques, visualization methods, compiler designs, distributed systems and embedded systems architectures), cybersecurity (e.g. information assurance, encryption, security frameworks, and digital forensics), computation and software development approaches which contribute to a wide range of innovative techniques that improved system reliability, efficiency, and fault tolerance. However, the rapid increase in digitization and subsequent integration into CPSs increased the attack surface and gave rise to new, additional security vulnerabilities. This results in potential losses of enormous economic value or destructive consequence in some cases. As a result, it is of significant importance to investigate the security issue of CPSs and ensure that such systems are operating in a safe manner even while incorporating more sophisticated operation.

Digital forensics has proven to be a powerful tool in data collection with a focus on data preservation as evidence in order to identify the root cause of a system failure or a cyber incident . However, most CPS forensic analysis techniques are limited to leveraging tools developed for traditional computing systems and therefore do not consider the differences between traditional computing systems and CPS in particular. Additionally, current forensic tools and frameworks focus on high level components of CPS, such as networked operations' management and control center operations, which do not adequately address data collection from lower level components close to the physical process identified as level 0-1

*This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes.
DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

in the Purdue Reference Architecture Model (PERA) provided in figure 1. While a focus on securing higher level CPS components is critical for security, it is also important to remember that network attacks are most likely not the first and only exploitation point when it comes to CPS. In [3], the authors present a survey on published forensics methodologies and frameworks applied to SCADA systems, which are CPS instances. In their paper, the authors demonstrate that a gap exists in the forensics of end-point devices in CPS, especially when it comes to live forensics of volatile memory and acquiring root cause evidence at run time. In an attempt to bridge this gap, we propose a framework for live memory forensics of CPS level 0-1 devices, that is based on adapting memory extracting techniques from the more mature IT space. As part of the ongoing research, we use the Modicon PLC as a use case, and apply candidate memory extraction techniques it.

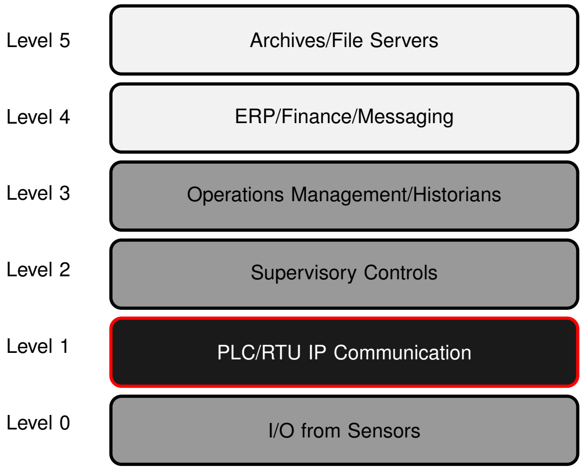

Fig. 1. The layers of industrial cyber-physical systems based on the Purdue Reference Architecture (PERA) Model

II. State of the Art

A. Layered Architecture of Cyber-Physical Systems

The Purdue Reference Architecture Model, provides a model for enterprise control, which end users, integrators and vendors can share in integrating applications at key layers in the enterprise. A description of each level is provided below: Level 0: (The physical process) Defines the actual physical processes. Level 1: (Intelligent devices) Sensing and manipulating the physical processes. Process sensors, analyzers, actuators and related instrumentation. Level 2: (Control systems) Supervising, monitoring and controlling the physical processes. Real-time controls and software; DCS, human-machine interface (HMI); supervisory and data acquisition (SCADA) software. Level 3: (Manufacturing operations systems) Managing production work flow to produce the desired products. Batch management; manufacturing execution/operations management systems (MES/MOMS); laboratory, maintenance and plant performance management systems; data historians and related middleware. Level 4: (Business logistics systems or Enterprise Resource Planning (ERP)) Managing the business-related activities of the manufacturing operation. ERP is the primary system; establishes the basic plant production schedule, material use, shipping and inventory levels.

B. Security Analyzers

Ujjwal et al. [4] propose an approach that is based on mutual authentication for both hardware and firmware. The authentication process begins with the hardware authenticating the firmware during the initial power-up sequence of the device by verifying the checksum. Additionally, the firmware also verifies the identity of the hardware, which cannot produce correct results unless it receives a unique hardware fingerprint that is known as a system ID.

SWATT [5] acts as an intermediary between the device and external actors during communication. A challenge-response protocol allows only validated devices to communicate. The technique is similar to secure bootstrapping; however, SWATT is unique because it does not require additional hardware to externally verify the system was in a secure state. One potential flaw with SWATT is the assumption that the device is in a trusted environment. The authors explain that because of the untrusted environment, any integrity checking functions stored on the unit can be modified by an attacker making them insecure.

Khan et al. [6]presents another verification method for embedded systems utilizing an external FPGA to store a hash and to check memory on device reset. If the computed hash matches the stored hash during the boot process, then the power up sequence continues and allows the device to run. If the hash values do not match, the device remains in reset mode. This method incurs additional overhead in the form of delay at boot time. However, it does not affect the systems functionality after verification occurs. This can be beneficial for field device security. System configuration changes will be required to implement boot time checking in a critical process environment because a no-start situation is unacceptable and may create a denial of service.

Wu et al. [7] emphasize the urgent need to develop forensic tools that are specifically designed for SCADA system. In an attempt to explore the problem, the authors propose two hypotheses and conduct experiments to prove their validity. For the first hypothesis, they state that the program code can be used as a forensic artifact that can reveal the attackers intention; their first experiment proved their concept and supported their first hypothesis. However, the second experiment for their second hypothesis, stating that PLC logger can be used as a forensic tool, failed most test cases. The authors conclude that PLC logger in its current state cant be used as a forensic tool for SCADA systems.

C. The Strengths and Weaknesses

The body of CPS's forensics literature sheds the light and attempts to find solutions for major vulnerabilities in CPS systems, such as persistent memory verification. However, devices at levels 0-1 do not have volatile memory support. Therefore, it is not possible to apply traditional algorithms, mechanisms, and analysis systems that primarily rely on artifacts such as system logs and file system data. This aspect makes it important to tap into volatile memory even while the devices are running (e.g. live run-time) while the CPS is in operational mode. To our knowledge, none of the previous works tackled the gap of code injection or attacks on volatile memory that goes undetected.

III. Threat Model and Challenges

Digital forensics is primarily concerned with the acquisition and analysis of digital evidence on computer devices. Digital forensics has been used for electronic crime investigations. However, it has been adopted recently for the purposes of security incident response. The main goal of digital forensics has been focused on the acquisition and analysis of persistent data found on hard dive disks and storage media. Unfortunately, modern cyber attacks had become more sophisticated and leaves little to no persistent traces of their activities. Consequently, persistent data investigations face difficult challenges in figuring out the complete picture of the cyber attack. As a result, there has been a research focus shift toward the recovery of transient system-state information stored in volatile memory, commonly referred to as memory forensics.

A. Volatile Memory Acquisition History for Traditional Systems

Digital forensics deals with the examination of computer systems in a forensically sound manner with the aim of identifying, preserving, recovering, analyzing and presenting facts that leads to the root cause of a system failure. In a traditional digital forensic approach, the investigator takes the targeted machine offline, and makes a copy (image) of the disk. The analyst then examines the image offline and in a controlled environment in search for digital evidence. While traditional forensics can be of value, this approach has several drawbacks. In many cases the size of the disk image is large and analyzing it can be a very tedious task. Additionally, taking the system offline for enough time to image the disk may not be acceptable as it leads to system downtime and potential monetary losses. Finally, detailed information about what is happening on a running system is lost when the plug is pulled. Live forensics, an approach that seeks to take a snapshot of the state of the system, and can capture both the volatile and static information about the file system is desired.

B. CPS Memory forensics

Historically, CPS systems were designed to be deployed in isolation from network, security threats were not taken in consideration. To re-mediate the vulnerable CPS systems, forensics practitioners attempted to adapt techniques and tools designed for IT systems to enhance the security of CPS. However, the focus was on the high level components of the systems, leaving 0-1 level devices with no support. Just as with traditional systems, CPS incidents may leave no traces in persistent memory and the acquisition of volatile memory becomes a must to to provide insight into the state of the system. Figure 2 presents the main components of a typical CPS systems, and illustrate that malicious activities can go undetected if volatile memory is not continuously verified.In a scenario where an attacker exploits a system vulnerability, and is able to execute code on the target device, the attacker can execute the malicious code in the volatile memory but leaves no traces in persistent storage. In such case, taking the device offline and examining the persistent storage will not lead to finding any digital evidence.

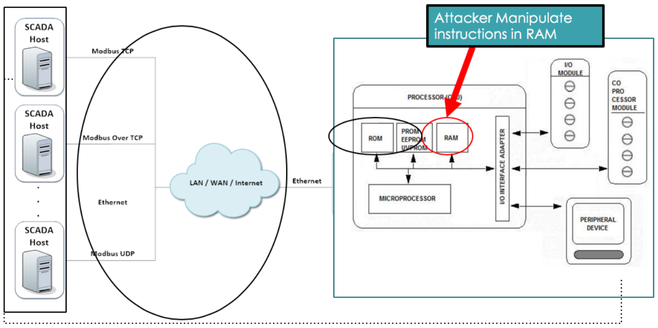

Fig. 2. Threat Model

C. Cited Cyber-Attacks Targeting End-Point Device's Memory

Kalle et al. [8] assume a realistic attack scenario where engineering software in the control center is not accessible for the attack. The authors exploit a critical (zero-day) vulnerability in the password authentication mechanism of a Modicon M221 PLC to acquire a logic program. Once they bypass security measures and extract the logic program, they decompile it, inject malicious code, then transfer the infected binary back into the PLC. The authors also claim success in keeping their attack stealthy by implementing a virtual PLC that intercepts the engineering station requests and responds by sending back non-malicious logic.

Beresford [9] exploits the Seimens S7 PLC by taking advantage of multiple protocol vulnerabilities. Beresford uses multiple methods to perform remote code execution attacks and demonstrates that it is possible for an attacker to read, write and modify configurations in memory. The authors also explain that it does not take much effort to cause damage to a PLC; a simple replacement of one TAG with another could cause a process to malfunction.

Yoo et al [10] argue that attackers target the control logic of a PLC over the network to manipulate the behavior of a physical process. To develop defensive solutions for control logic injection attacks, it is required to explore new attack vectors that target the control logic of a PLC to sabotage a physical process. The authors take advantage of the fact that PLCs do not enforce data execution prevention (DEP). For their attack scenario, they proposes two control logic injection attacks: 1) Fragmentation and Noise Padding: The attacker transfers a (malicious) control logic as fragments with packets padded with noise to avoid detection. The malicious logic is transfered to an arbitrary address of a target PLC which is not in the address range for the code block, and thus, are not blocked by the signatures. 2) Data Execution: the attacker targets the pointer in the configuration block, which indicates the address of the code block and are used by the PLC to start executing the control logic. The attacker modifies the pointer to the base address of the malicious control logic in the data block, which in turn, redirects the PLCs system control flow to the malicious logic and forces the PLC to start executing it.

D. Diversity of Memory layout of End-Point Devices: The Challange

Cited cases of ingress and egress for MicroLogix live data acquisition is needed to acquire volatile data for subsequent offline analysis. The procedure must be shown to avoid or minimize endpoint device service interruption. One of the main challenges that a CPS forensic analyst encounters is the customized architecture and kernels running on the system components. For example, the Modicon PLC and the Allen Bradly MicroLogix PLC both operate as a distinct architecture and both run proprietary firmware [11]. Eventually, a tool developed specifically for one PLC will not be typically no be compatible with the other. Not only do devices from different manufacturers have diverse architectures and memory layout, preliminary testing indicates this problem exists within the same manufacturer.

IV. Volatile Memory Acquisition Approaches: IT Systems & CPS Systems

With the increased number and sophistication of cyber-attacks, digital forensics becomes an increasingly valuable source of digital evidence for both IT and OT space. However, solely focusing on traditional forensic sources (persistent storage), can lead to incomplete picture of an incident. In addition, sophisticated modern attacks may operate solely in memory and do not leave any persistent traces [12]. To address the shortcoming of traditional forensics, forensic practitioners started giving more attention to volatile memory forensics. Although very little work has been done to acquire volatile memory of CPS 0-1 devices, there are many techniques and methodologies that were proposed to extract RAM of a computer. The purpose of this section is to highlight methodologies developed for IT systems and identify those that can possibly be applied to CPS 0-1 level devices. As for an ongoing work, we plan apply the identified candidate IT methodologies to selected CPS devices, and statistically compare the results to define how effective they can be if adapted as a memory extraction capability for CPS 0-1 devices.

A. Volatile Memory Acquisition for IT Systems

To measure the soundness and quality of acquired memory image, the digital forensic literature introduces three criteria [12] [13]: Correctness, atomicity and integrity. However, authors in [14] state that the most difficult component of memory forensics is typically the acquisition of a memory image in both an atomic and available manner, where availability refers to that the approach must work on arbitrary computers and/or devices.

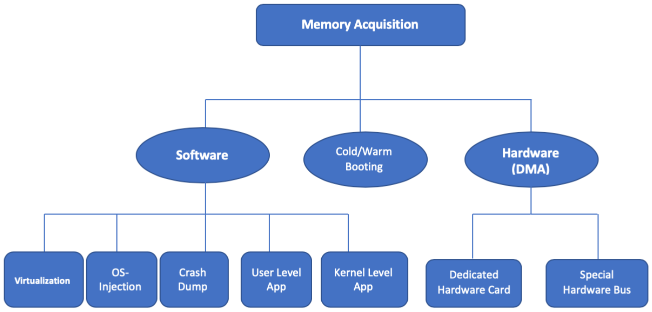

Fig. 3. Memory Acquisition Approaches for IT Systems

Technique	Atomicity	Availability	Comments
Dedicated Hardware	High	Low	Atomicity may be compromised with hardware tampering.
Hardware Bus	Moderate	High	Atomicity affected by random crashes in some cases.
Virtualisation	High	High	In environments that utilise virtualisation.
Crash Dumps	Low	Low	Not all memory is dumped, can overwrite system page file and requires registry modifications to work.
User-mode Applications	Low	High	Easily subverted, will modify memory when capturing it and will not have access to entire memory range.
Kernel-mode Applications	Low	Moderate	Easily subverted and will modify memory when capturing it.
Operating System Injection	High	Low	Reliance on hardware platforms and very slow.
Cold Booting	High	High	Requires off the shelf items and Syslinux

Fig. 4. Evaluation of software and hardware approaches [14]

Techniques for capturing volatile data from IT systems have conventionally been divided into hardware and software based techniques. While software approaches depend on functions provided by the operating system, hardware-based approaches directly access a computers memory during the imaging process. Figure 3 is a summary of existing software and hardware memory acquisition approaches used in the IT space.

In [14], authors evaluate the approaches presented in 3 in terms of availability and atomicity. According to their evaluation, the authors conclude that hardware based approaches are highly atomic, but they are low in terms of availability as they require the preparation of a system prior to its investigation. On the other hand, software-based approaches are more available, but less atomic since the software needs to run on the target system's memory.

B. Volatile Memory Acquisition for 0-1 CPS Forensics

Very limited forensics tools have been developed for CPS in general and even less for 0-1 level devices.

1) Acquisition via Function Codes of CPS Communication Protocols: Due to the nature of CPS low level devices, CPS communication protocols have function codes that allows reading specific data types form memory space, and in some cases the logic program from live devices. Authors in [15] analyzed two major vendors PLCS to evaluate the possibility of using communication protocols function code as a forensic capability. The main focus of authors was on evaluating the effectiveness of the capability in acquiring RAM and NVRAM. As a conclusion of their experiment, they find that it was possible to acquire the RAM of one of the PLC's, Whereas the second PLC does not offer any support. Similarly, one of the PLC's offer little support for extracting NVRAM, while the second one had none.

2) Acquisition via JTAG Port: JTAG (Joint Test Action Group), commonly referred to as boundary-scan and defined by the Institute of Electrical and Electronic Engineers (IEEE) 1149.1, was developed as an integrated method for testing and debugging interconnects and components on PCB board [16]. Beyond PCB boards debugging, JTAG has been adapted for digital forensics purposes and has been used as a method to extract a full physical image from devices that cannot be acquired with normal tools. This is accomplished by connecting the Test Access Port (TAP) on the PCB board to a JTAG emulator to access raw data stored in the connected device. By using the TAPs, communication can occur via the boundary-scan path, which interfaces with the Boundary Scan Registers (BSR) that interface with components on the PCB. These components can be programmed or read without being removed independently, for reading or programming purposes [17].

JTAG memory acquisition is an invasive but non-destructive forensic technique that allows the acquisition of full memory space. However, using JTAG is available only for a limited number of devices with TAP ports. In addition, it requires a high skill level to disassemble the targeted device and identify the TAP pins, and according to [17], the acquisition speed is very slow.

C. Discussion

Digital forensics for CPS systems is not as mature as it is in IT systems. The fact that CPS systems were not initially designed with security in mind, and the nature of such systems, makes it challenging to deploy existing IT forensic techniques in CPS systems. Whereas IT systems are usually designed for general purposes, such as storing, moving, and protecting data, CPS systems are purpose built for and are usually used for manufacturing, supervising and controlling utilities. In addition, CPS end-point devices have to follow a specific business logic, and the breakage to business logic is likely to cause serious accidents. Different from IT systems, CPS does not allow for service interruption window. While a false positive or even loss of service can result in business delays in traditional IT, the impact on CPS systems can be very costly and perhaps disastrous and life threatening in some cases.

Adaption of IT digital forensic techniques for CPS systems can be challenging. However, a careful study of both systems' architecture and a thorough understanding of how they function may allow selected IT forensics techniques to be deployed in CPS systems while addressing the specific requirements and challenges. After thoroughly studying the nature of CPS end-point devices, we have concluded that it is possible to evaluate almost all IT hardware/software memory acquisition techniques on CPS devices. We exclude cold/warm booting techniques as they are very invasive and may require the destruction of the targeted device's PCB, and, hence, yields unusable.

V. Experimental Framework Development

To address the gap in the realm of CPS forensics, we propose a methodology for forensics on level 0-1 devices of the CPSs. We mainly attempt to evaluate and compare the effectiveness of IT forensic techniques on selected CPS end-point devices. We mainly target the volatile memory of the selected devices and intend for acquisition and verification at run-time without interrupting the processes controlled by the device. We intend to accomplish the verification step by continually acquiring the memory dump of volatile portions of the memory and monitoring the changes as the device is running. This should allow us to extract essential data from acquired memory dumps in order to triage the system in case of system failure or cyber attack and, eventually, identify the root cause of the failure. We overcome the diversity of end-point devices by developing a general methodology that consists of a subset of device specific tools for memory acquisition.

A. Strengths and Benefits: Addressing the Weakness

Most CPS security analyzers focus on level 2-3 components of the system. There are a few works that target level 0-1 components to either (i) attempt to verify non-volatile storage such as flash memory and PROM, or (ii) use attestation that typically requires the availability of dedicated hardware [18]. However, the nature of embedded devices and the fact that they are constrained in terms of resources, the use of attestation to verify memory is challenging and does not exist without limitations. Our proposed methodology overcomes the challenge introduced by attestation, as it does not rely on hardware. The fact that we collect memory images from end point devices and transfer them back to the host to be analyzed offline, makes the proposed methodology lightweight, hence, it does not exhaust the resources of the end-point device.

B. Use case example

As a use case for this paper, we evaluate the possibility of acquiring volatile memory from PLC by using the communication's protocol function codes. For this purpose, we use the M221 Modicon PLC manufactured by Schneider electric. We initiate communication with the device via the Modbus communication protocol, and we take advantage of the protocol's function codes, such as codes used for read and write requests. We first subject the device to a write request, to initialize the list of essential memory address. We accomplish this by analyzing the packet format for the write request. Later, we issue a read request that goes over the list of essential memory address and returns their content. 5 is the process flow diagram that illustrates the approach. For the Modicon

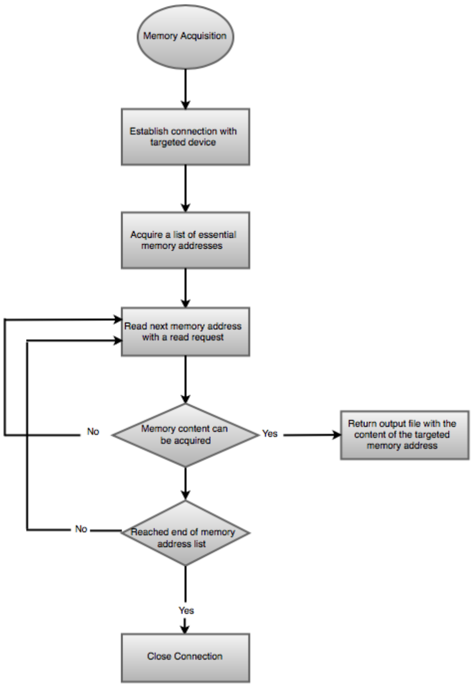
Memory Acquisition
Establish connection with targeted device
Acquire a list of essential memory addresses
Read next memory address with a read request
No
Memory content can be acquired
Yes
Return output file with the content of the targeted memory address
No
Reached end of memory address list
Yes
Close Connection

Fig. 5. Process Flow Diagram

PLC, we were able to acquire the entire memory space of the device. We also, attempt to evaluate memory extraction technique via JTAG describe in 4 on the same device. For this purpose, we disassemble the device in search for TAP port, and we use JTAGulator device [19] to identify the TAP pinouts that are kept by manufacturers on PCB board for future debugging and improvement. Despite the fact the we were able to find the JTAG pins, unfortunately, they were disabled by manufacturer and we were not able to use them to acquire the memory.

VI. Ongoing and Future Work

Ongoing work includes experimental testing with volatile memory capture techniques adapted for IT space that do not introduce service interruptions or degrade services while the device is functioning in full operational mode. Various manufacturer devices will be introduced to verify that the technique is adaptable for various CPS protocols and is agnostic to memory layout and system architectures. Statistical analysis will be incorporated to detect and confirm departures from normality with regard to system performance.

VII. Conclusion

CPS systems are the underpinning technologies that ensure the proper operation and functionality of critical national infrastructures. With the rise of attacks against critical infrastructure and CPS systems, it becomes necessary to leverage digital forensics in similar increasingly complex ways. Unlike traditional IT systems, forensics capabilities for CPS systems are still in early stages of development. This is mainly to their specialized and critical nature, in addition to the fact that they were not designed with security in mind, and and the prevalence of proprietary and poorly documented protocols. Nevertheless, over the recent years forensics practitioners and security professionals have adapted various forensics methods and tools from IT space to the CPS world. Unfortunately, CPS 0-1 level devices were out of their scope, which left the systems highly vulnerable as end-point devices present an appealing attack vector to intruders. To address this gap, we attempt to apply memory forensics techniques developed for IT systems on CPS systems. After careful statistical analysis of the results, we propose a general purpose framework that consists of a subset of device specific tools for live memory forensics of CPS level 0-1 devices.

References

[1] T.J.Williams, P.Bernus, J.Brosvic, D.Chen, G.Doumeingts, L.Nemes, J.L.Nevins, B.Vallespir, J.Vlietstra, and D.Zoetekouw, "The purdue enterprise reference architecture," *Computers in Industry*, vol. 24, no. 2, pp. 141–158, 1994.

[2] I. Ahmed, S. Obermeier, M. Naedele, and G. G. Richard III, "Scada systems: Challenges for forensic investigators," *Computer*, vol. 45, no. 12, pp. 111–139, 2012.

[3] R. A. Awad, S. Beztchi, J. M. Smith, B. Lyles, and S. Prowell, "Tools, techniques, and methodologies: A survey of digital forensics for scada systems," in *Proceedings of the 4th Annual Industrial Control System Security Workshop*. ACM, 2018, pp. 1–8.

[4] U. Guin, S. Bhunia, D. Forte, and M. M. Tehranipoor, "Sma: A system-level mutual authentication for protecting electronic hardware and firmware," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 3, pp. 265–278, 2017.

[5] A. Seshadri, A. Perrig, L. Van Doorn, and P. Khosla, "Swatt: Software-based attestation for embedded devices," in *IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004*. IEEE, 2004, pp. 272–282.

[6] A. Khan, G. Ganesh, S. Dhodapkar, B. Biswas, R. Patil *et al.*, "A cryptographic primitive based authentication scheme for run-time software of embedded systems," in *2010 2nd International Conference on Reliability, Safety and Hazard-Risk-Based Technologies and Physics-of-Failure Methods (ICRESH)*. IEEE, 2010, pp. 500–504.

[7] T. Wu and J. R. Nurse, "Exploring the use of plc debugging tools for digital forensic investigations on scada systems," *Journal of Digital Forensics, Security and Law*, vol. 10, no. 4, p. 7, 2015.

[8] S. Kalle, N. Ameen, H. Yoo, and I. Ahmed, "Clik on plcs! attacking control logic with decompilation and virtual plc," in *Binary Analysis Research (BAR) Workshop, Network and Distributed System Security Symposium (NDSS)*, 2019.

[9] D. Beresford, "Exploiting siemens simatic s7 plcs," *Black Hat USA*, vol. 16, no. 2, pp. 723–733, 2011.

[10] H. Yoo and I. Ahmed, "Control logic injection attacks on industrial control systems," in *IFIP International Conference on ICT Systems Security and Privacy Protection*. Springer, 2019, pp. 33–48.

[11] M. D. Schwartz, J. Mulder, J. Trent, and W. D. Atkins, "Control system devices: Architectures and supply channels overview," *Sandia Report SAND2010-5183, Sandia National Laboratories, Albuquerque, New Mexico*, vol. 102, p. 103, 2010.

[12] S. Vömel, "Forensic acquisition and analysis of volatile data in memory," 2013.

[13] M. Gruhn and F. C. Freiling, "Evaluating atomicity, and integrity of correct memory acquisition methods," *Digital Investigation*, vol. 16, pp. S1–S10, 2016.

[14] G. Osbourne, "Memory forensics: Review of acquisition and analysis techniques," DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION EDINBURGH (AUSTRALIA) CYBER AND , Tech. Rep., 2013.

[15] I. Ahmed, S. Obermeier, S. Sudhakaran, and V. Roussev, "Programmable logic controller forensics," *IEEE Security & Privacy*, vol. 15, no. 6, pp. 18–24, 2017.

[16] M. Breeuwsma, "Forensic imaging of embedded systems using jtag (boundary-scan)," *digital investigation*, vol. 3, no. 1, pp. 32–42, 2006.

[17] O. Afonin and V. Katalov, *Mobile Forensics–Advanced Investigative Strategies*. Packt Publishing Ltd, 2016.

[18] A. Abbasi and M. Hashemi, "Ghost in the plc designing an undetectable programmable logic controller rootkit via pin control attack," *Black Hat Europe*, pp. 1–35, 2016.

[19] J. Grand, "Jtagulator: Assisted discovery of on-chip debug interfaces," in *21St defcon conference, Las Vegas*, 2013.