SWaT: A Water Treatment Testbed for Research and Training on ICS Security

Aditya P. Mathur and Nils Ole Tippenhauer
Singapore University of Technology and Design
8 Somapah Road
Singapore 487372
{aditya_mathur, nils_tippenhauer}@sutd.edu.sg

***Abstract*—This paper presents the SWaT testbed, a modern industrial control system (ICS) for security research and training. SWaT is currently in use to (a) understand the impact of cyber and physical attacks on a water treatment system, (b) assess the effectiveness of attack detection algorithms, (c) assess the effectiveness of defense mechanisms when the system is under attack, and (d) understand the cascading effects of failures in one ICS on another dependent ICS. SWaT consists of a 6-stage water treatment process, each stage is autonomously controlled by a local PLC. The local fieldbus communications between sensors, actuators, and PLCs is realized through alternative wired and wireless channels. While the experience with the testbed indicates its value in conducting research in an active and realistic environment, it also points to design limitations that make it difficult for system identification and attack detection in some experiments.**

***Keywords*: Cyber Physical Systems, Industrial Control Systems, Cyber Attacks, Cyber Defense, Water Testbed**

I. INTRODUCTION

In recent years, cyber-security threats to Industrial Control Systems (ICS) have received increased attention from the industry and research communities. ICS are built from, and depend upon, the integration of computational algorithms and physical components. ICS have evolved as a natural consequence of increasingly interconnected physical processes. They interact with the physical world, i.e., by sensing and actuating physical processes, and also with users, e.g., via human-machine-interfaces (HMIs), engineering work stations, corporate work stations, smart phones, etc. ICS are found in diverse areas such as critical infrastructure systems (electricity, water, gas distribution, communication and transportation networks), industrial applications (such as process plants, automotive industries), and small scale systems (such as robotics, health care systems, and home automation).

The inclusion of networking within an ICS, and in some cases its connectivity to the Internet, introduces the threat of cyber attacks. Such attacks could come from inside the system perimeter such as by an employee, or through the network from an outside attacker. In either case, researchers have proposed algorithms for the prevention and detection of attacks. Mechanisms for defending an ICS against attacks have also been proposed. However, many of the published algorithms are assessed for their effectiveness using simulation or numerically [2], [4], [6], [15], [16]

978-1-5090-1161-2/16/$31.00 ©2016 IEEE

This paper describes an operational ICS, the **S**ecure **W**ater **T**reatment (SWaT) testbed. SWaT was designed and built to enable experimental research in the design of secure ICS, and is one of the core components of a larger research effort in iTrust, all focusing on the design of secure cyber-physical systems. In addition to the testbed, work is also underway on complementary tools such as simulation environments [1], attack, and defense modeling tools. The long term objective of SWaT, and other testbeds that would be operational alongside SWaT, is to transform the process of ICS design. The current state-of-the-art in ICS design focuses on functionality and safety. Testbeds, and the associated research projects under the iTrust research center, are aimed at bringing cyber security into the design stage of ICS. However, doing so requires extensive experimentation to validate software and hardware based methods and tools aimed at improving ICS resilience. Testbeds such as SWaT are essential for such validation. All testbeds under iTrust are available for collaborative projects that allow the broader ICS community to contribute to realizing the long-term goal of transforming ICS design process.

Contributions: (a) Design of a testbed for research in cyber security of Industrial Control Systems. (b) Use of a testbed in assessing the effectiveness of methods for cyber attacks and and defense against.

Organization: The remainder of this paper offers an introduction into the architecture of SWaT, its utility, and lessons learned. The overall physical and cyber architecture of SWaT is in Section II. A set of sample experiments conducted so far using SWaT, and the outcome, are in Section III. Similar testbeds and a brief comparison with SWaT are in Section IV. Strengths and shortcomings of SWaT are in Section V. The conclusion and future plans for the use of SWaT and collaboration with other researchers are provided in Section VI.

II. ARCHITECTURE OF SWAT

SWaT is an operational testbed for water treatment producing 5 US gallons/hr of filtered water. In a small footprint of approximately 90 square meters (Figure 1), the testbed represents a small-scale version of a large modern water

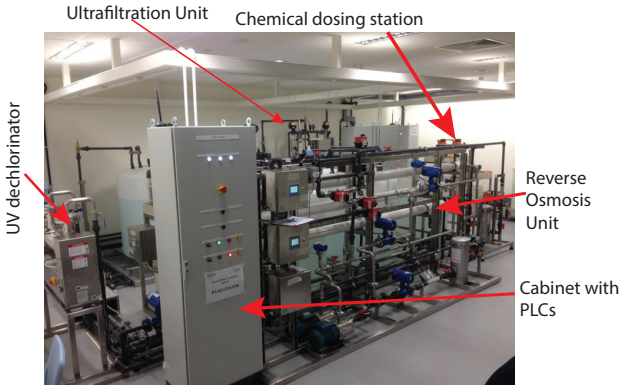

Fig. 1. A pictorial view of SWaT. The Reverse Osmosis unit is seen in the front, while the view on Ultrafiltration unit, tanks, and several other components is obstructed.

treatment plant found in large cities. The overall testbed design was coordinated with Singapore's Public Utility Board, the nation-wide water utility company, and constructed by a third party vendor. That collaboration ensured that the overall physical process and control system closely resemble real systems in the field, so that the results can be applied to real systems as well. We use SWaT to investigate cyber-attacks and respective systems responses, and to conduct experiments with novel countermeasure designs (e.g., physics-based). As shown in Figure 2, SWaT consists of six stages labeled P1 through P6. Each stage is controlled by its own set of dual PLCs, one serving as a primary and the other as a backup in case of any failure of the primary. Overall, the testbed leverages a distributed control approach in normal operations, where each process stage is individually controlled by the local PLCs. For some of the process stages, the local control requires state information from other process, to obtain this information, the PLCs are networked and in constant communication. In addition to this automated distributed control mode, the operator can also manually control all actuators of the testbed through the HMI, and the SCADA system.

Communications: We provide a general overview of the communication structure in Figure 3. Within each process stage, the main PLC obtains data from local sensors and controls actuators such as pumps and valves. For example, turning the pumps ON, or opening a valve, causes water to flow either into or out of a tank. In addition to the actuators, sensors such as level sensors in each tank enable the PLCs to monitor the status of the system, and to decide when to turn a pump ON or OFF. Several other sensors are available to check the physical and chemical properties of water flowing through the six stages. The local communications between a PLC and its direct sensors and actuators is using an Ethernet-based ring topology, using Allan-Bradley's Device Level Ring (DLR) protocol. The ring ensures that loss of a single link can be tolerated, and no data or control functionality is impacted.

Between the different process stages, PLCs communicate with each other through a separate network, which we call L1 network. That network is based on a conventional Ethernet star topology, with an industrial switch connecting all 6 process stages, the HMI, SCADA system and historian.

All network communication by PLCs, sensors and actuators in SWaT is using the industrial EtherNet/IP (ENIP) and Common Industrial Protocol (CIP) stack [8]. In ENIP, sensor values or actuator settings are mapped to *tags*. Each tag can be addressed either via a string descriptor defined by the system designer (e.g., MV101 for motorized valve 1 in process 1), or a more direct mapping to bank number and pin number or similar (directly referring to digital/analog pins of a unit's IO panel). Communications among sensors, actuators, and PLCs can be via either wired Ethernet or Wi-Fi links; manual switches allow to change the configuration between the wired and wireless communication.

Stages in SWaT: Stage P1 controls the inflow of water to be treated by opening or closing a valve (not shown) that connects the inlet pipe to the raw water tank. Water from the raw water tank is pumped via a chemical dosing (stage P2,chlorination) station to another UF (Ultra Filtration) Feed water tank in the stage P3. In stage P3, a UF feed pump sends the water via UF membrane to RO (Reverse Osmosis) feed water tank in stage P4. Here an RO feed pump sends the water through an ultraviolet de-chlorination unit controlled by a PLC in stage P4. This step is necessary to remove any free chlorine from the water prior to passing it through the reverse osmosis unit in stage P5. Sodium bisulphate (NaHSO3) can be added in stage P4 to control the ORP (Oxidation Reduction Potential).

In stage P5, the de-chlorinated water is passed through a 3-stage RO filtration unit. The filtered water from the RO unit is stored in the permeate tank and the reject in the UF backwash tank. Stage P6 controls the cleaning of the membranes in the UF unit by turning on or off the UF backwash pump. The backwash cycle is initiated automatically once every 30 minutes and takes less than a minute to complete. Differential pressure sensors in stage P3 measure the pressure drop across the UF unit. A backwash cycle is also initiated if the pressure drop exceeds 0.4 bar, indicating that the membranes need immediate cleaning. A differential pressure meter installed in stage P3 is used by PLC-3 to obtain the pressure drop.

Each PLC has memory locations, known as *tags*, to save sensor data. There is one tag for each sensor connected to a PLC. These tags are accessible across the entire SWaT. Thus, for example, the level of tank T301 in stage P3, can be obtained by PLC 1 to decide whether to start pump P101. Tag values are also sent to SCADA on demand. All sensor data, at every time instant, can be sent to the historian and saved for future analysis. Note that the historian resides in a separate computer connected via the Level 1 network to the PLCs and SCADA system. The availability of tag values at different points in SWaT was found useful in implementing attack detection algorithms. For example, PLC 3 can look into tags in PLC 1 to check if a process invariant—a physical condition—that uses sensors connected to both PLC 1 and

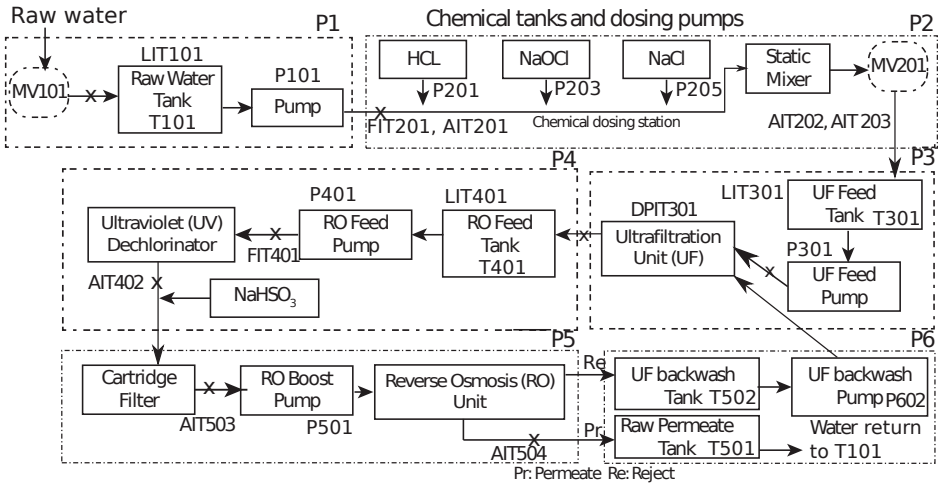

Fig. 2. Physical water treatment process in SWaT and attack points used in the case study. P1 though P6 indicate the six stages in the treatment process. Solid arrows indicate flow of water or chemicals in the dosing station. Dashed arrows indicate potential cyber attack points. LIT: Level Indicator and Transmitter; Pxxx: Pump; AITxxx: Property indicator and Transmitter; DPIT: Differential Pressure Indicator and Transmitter.

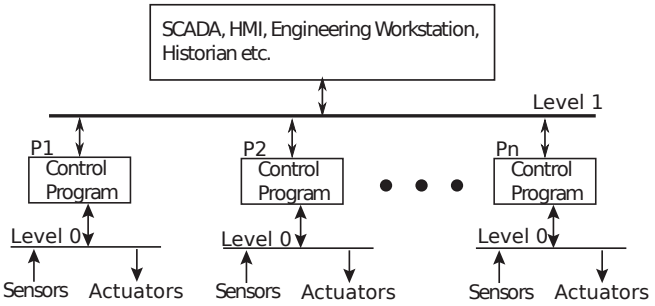

Fig. 3. Architecture of the control portion of a CPS. P1, P2,. . . ,Pn denote PLCs. Each PLC communicates with its sensors and actuators through a local network at Level 0. PLCs communicate among themselves via another network at Level 1. Communication with SCADA and other computers is via a Level 3 network not shown here. Note that the actuators, e.g., a pump, also have sensors to indicate their condition.

PLC 3, is true.

III. Experiments with SWaT

SWaT is serving as a fertile playground for faculty and researchers at SUTD, external academic collaborators, and commercial organizations. We believe that academic security research on ICS suffers from the general problem that it is difficult to access live ICS, in particular any active (security-related) tests in such systems are typically impossible. As a result, realistic examples of detailed ICS are rare, and theoretical assumptions made by researchers can be misleading. To mitigate that problem, we started a range of experiments on SWaT, which are supposed to give other researchers a better insight into the setup of an ICS such as SWaT. We hope that such information facilitates research activities, in particular assessment of academic and commercial prototypes, and experimental techniques. As SWaT is available for collaborative projects with our international partners, we expect that the outlined experiments are just the first steps, and more detailed results will follow soon.

A. Attacker Model

Initial system reconnaissance and first attacks on the system were launched soon after SWaT became operational. In particular, we focused on several attacker settings: a) an attacker A who has access to the local plant communication network, b) an attacker B who is in physical proximity, but not directly on site, and c) an attacker C who is on site and has physical access to the devices. In all cases, the goal of the attacker is to manipulate the normal operations of the plant. In an exemplary attack, the goal was to overflow the raw water tank in SWaT. It is argued that that this goal, while not threatening high damage, is representative for an attack that requires attackers to fully control sensor and actuators in an ICS. At the same time, it is safe enough to achieve in the lab without damaging equipment. For the first attack scenarios in Section III-B, we assume that the attacker has general knowledge about the technology used, but does not know details of the system under attack yet. In that setting the attacker tries to obtain more information on the system under attack, and find additional attack vectors to manipulate the system. In particular, the attacker has tools to interact with the industrial devices using the specific industrial protocols, such as EtherNet/IP and CIP [8].

B. System Reconnaissance

The experimental reconnaissance attack assumed the presence of an attacker A, that has access to the L1 plant network. We used a standard laptop with wired and wireless network interfaces, with open source networking tools such as Wireshark and Zenmap. With that setup, we were able to quickly map the local networking setup, and determine the available services on the hosts. We discovered a range of web interfaces on the local PLCs and networking devices. In

particular, devices such as PLCs provide an informative web interface with a summary of their configuration and setup. In addition, the local HMI device (AB PanelView Plus Terminal) is running an embedded Windows OS, an FTP server that allows anonymous login, and a remote desktop protocol (RDP) server. Anonymous FTP login enabled the discovery of *hidden* files that appear to contain the complete HMI configuration in a proprietary format (Composite Document File V2).

As all PLCs, the HMI and the SCADA system are within the same Link-layer broadcast domain, it was possible to launch ARP spoofing attacks using Ettercap [3]. For more details on that attack, refer to [1]. As a result of the attack, the attacker is able to arbitrarily re-direct local traffic through his machine, and eavesdrop or manipulate the content. We found that the industrial protocol used, ENIP, does not feature any authentication or encryption in our testbed. Protocol analyzers such as Wireshark are able to decode ENIP to some degree, so that exchanged data can be extracted. We are currently also working on extensions for the Scapy tool, to enable automated processing and generation of ENIP traffic.

In addition to sensor data and actuator commands, it was also possible to capture actions such as remote firmware and logic updates, from the SCADA to individual PLCs. The new programming seems to be sent around in cleartext (as binary file), so it would be possible for attacker A to obtain detailed information on the used logic on the PLCs, and also to reprogram the PLCs with manipulated logic.

C. Compromise through Wireless Network

Following the first set of experiments on reconnaissance, further investigation was carried out on potential compromise by attacker B, an attacker within WiFi range of the plan control system. In particular, the SWaT testbed has the option to replace the wired Ethernet-based L1 network with a WiFi-based wireless solution. The alternate wireless network uses industrial access points (the MOXA AWK-5222-EU) to connect the devices, and employs the WPA2 security scheme with pre-shared keys. Assuming that the pre-shared key is strong enough to not be guessed outright, there are several options for the attacker: a) the attacker can try to perform a (cloud-based) brute force attack, b) the attacker can perform a well-known *evil twin attack*[11] to impersonate the legitimate AP, and trick the PLCs to connect to it instead. Suitable tools for both attacks exist, for example the Aircrack-Ng tool. The feasibility of the brute-force approach depends greatly on the strength of the chosen password. In our case, we left the choice of the password to the system integrator, and we found that the chosen password would easily be guessable with a dictionary.

In addition, it was noted during the wired reconnaissance of attacker A, that the AP provides a web interface for configuration, and that its default user account for that web interface had the default password. That password enabled cleartext password for the wireless network of that AP in the HTML source code (the rendered HTML replaced it with bullet points). As a result, attacker A would be able to obtain the WiFi password quickly, and be able to use it later to remotely connect to the WiFi network to launch attacks from a safe distance.

D. Compromise through Direct Physical Access

Attacker C, who has direct physical access to the network, has a range of additional options to attack the SWaT testbed. In particular, the attacker could arbitrarily re-wire networking cables, insert passive taps (such as [9]), or manipulate sensors [14]. It was also found that the PLC model (1756 ControlLogix) used in SWaT features SD card slots that can be used to update the coded control logic in the PLC. While this is yet to be tested in practice, this seems like a promising attack vector for attacker C.

E. Conclusion on Attacker Model

Several ways were identified as to how attackers A,B, and C can fully manipulate the communication in the L0 ring or L1 networks. As a result, the outlined attacker will be able to insert itself as man-in-the-middle between any two parties (e.g. two PLCs), and will be able to eavesdrop on all exchanged sensor and command data. In addition, the attacker can use the Ettercap rules we designed to re-write sensor or command values on-the-fly. While the outlined attacks are well known in traditional computer networks, and several countermeasures are available, the initial configuration of the SWaT system did not provide any means to detect or prevent the ARP spoofing based attack. As a result, the attacker must be assumed to be able to obtain full knowledge of the topology, technology, and operational parameters of the attacked system.

F. System Response to Attacks

It is important to know how a CPS will respond to cyber and physical attacks. This information is useful in designing detection and defense mechanisms. While one could obtain this information based on modeling and simulation of the design using tools such as LabView [5] or Simulink [13], the assumptions made to create a working model might taint the results; recent comprehensive experimental work with a robot [12] serves to emphasize this point.

Example 1: The purpose of this attack is to degrade the performance of SWaT from the nominal 5 gallons/minute to a lower value. To understand how this could be done, consider the fact that the UF unit contains micrometer sized membranes to remove small particles from the water to be filtered. PLC 6 (stage P6 in Figure 2) is programmed to clean the UF every 30 minutes by using a backwash process. However, depending on the quality of the incoming water, UF may need to be cleaned sooner. PLC 3 (stage P3 in Figure 2) is responsible for checking whether or not UF should be cleaned. This is determined by an examination of the data received from the differential pressure sensor, DPIT 301. This sensor checks the pressure difference across the UF, i.e., across the incoming and outgoing streams of water. A differential pressure higher than 0.40kpa indicates that the UF ought to be cleaned soon.

A simple attack is to compromise the link from DPIT301 to PLC 3 and send false data to the PLC. This attack

requires continuous reconnaissance of the DPIT data link and compromising it at an appropriate instant which could be any time before the programmed backwash cycle is to begin. This attack was launched by changing the DPIT301 value sent to PLC 3 from 20Kpa to 42kPa. Consequently PLC 3 initiated a backwash process as the pressure drop, as assumed by PLC 3, was more than the maximum acceptable, i.e., 40kpa. In a similar attack, sensor LIT401 was compromised. The impact of attacking sensor LIT401 was measured on the flow rate of water at the output of the RO unit. According to system specifications, this flow rate must remain at about 1.2cm/hr which leads to nearly 5 gallons/minute of treated water. The single point attack on LIT401 changed the level of the RO feed water tank, as known to the PLC in stage 5, from 800mm to 200mm. This caused the PLC to stop pump P401.Doing so reduced the amount of water produced to 113 gallons from the expected 155 gallons during the observation period. ■

Example 2: A number of experiments were performed to investigate the effectiveness of process invariant(physics) based approaches in the detection of cyber attacks. One outcome of these experiments is a list of emerging design parameters that ought to be considered while designing a secure ICS. These parameters include the number of data points to be used by the PLC control logic to decide on what control action to take, and the number of data points to be used by the detection algorithm before it announces an attack or no attack. Several parameters that define an attack have also emerged. For example, in intermittent attacks, an attacker may control the width of the attack pulse to thwart the detection algorithm. Perhaps the most interesting outcome of these experiments was the realization that an attack launched on a sensor immediately prior to power outage, or immediately following power outage, is the most difficult to detect using invariant based approaches.■

IV. Similar Testbeds

There exist a number of testbeds in the areas of power and water. Some of these allow simulation of large systems and do not actually produce power or water. Other testbeds are operational in the sense that they actually generate a usable product though in smaller quantities than their real counterparts. Both types have their pros and cons. Simulation-based testbeds allow large scale attack analysis, e.g., a large number of buses in a transmission system. Operational testbeds allow the conduct of experiments that input data from actual sensors and command actuators thus enabling more realistic validations than their simulation counterparts.

In [7], the authors present a set of small scale physical processes and control systems from the domain of gas pipelines, water distribution, and manufacturing at the Mississippi State University. The individual process stages involve few sensors and RTU units, and industrial control software. In contrast to SWaT, the underlying physical processes are of much smaller scale. In particular, the SWaT testbed involves the full cycle of water treatment with several filtration stages, with a significant throughput of 5 US gallons per minute. In addition, the distributed control implemented in SWaT is significantly more complex than the controls implemented in that system. A mini-water testbed is available at the University of Lancaster [10]. It allows communications with field emulated sites using multiple communications media such as telephone or leased line, and satellite. The key advantage of having small tanks that fit in a bookcase, is that the impact of attacks can be observed quickly in contract to SWaT where the tanks are much larger that requires significant wait time before the impact of an undetected attack can be observed visually or via sensors.

V. Lessons Learned from SWaT

We now summarize a number of lessons learned from the process of designing and implementing the SWaT testbed, related to the industrial networking protocols, industrial software used, physical layout of the testbed, and the raw water used for filtration.

A. Physical Layout of the Testbed

The availability of a *real* physical testbed complete with process and controls continues to be a great benefit for our researchers. In addition, the testbed also developed to be a major attraction for guests visiting our campus. While we planned the testbed to be easily supervisable through a window from the neighboring room housing the SCADA system, it turns out that the physical layout of the testbed is not well suited towards groups of more than 5 visitors. In particular, larger groups of visitors have to spread out more in the lab, and have problems to understanding the guide. As a result, we are designing our future testbeds with such use cases in mind: larger open spaces allow visitors to both have an unobstructed view on the process, and stay in contact with the guide. In addition, we plan to have simple barriers to discourage guests from manipulating devices without consent.

B. Industrial Protocols

At the time of designing SWaT specifications for the system vendor, no specification was given for any particular industrial protocol to be used (e.g., Modbus/TCP); only Ethernet-based communication was a requirement. Consequently, the vendor proposed the use of EtherNet/IP and CIP, which was accepted by the tender evaluation team. That decision has turned out to be both advantageous and disadvantageous, as there is little existing open-source tool support to interact with EtherNet/IP protocols. That requires us to extend existing tools, and contribute to the community. If we had chosen to use Modbus/TCP instead, that effort (and contribution) would not have been required.

C. Industrial Software

Perhaps unsurprisingly, the industrial software used on the SCADA and Historian system is not very open towards integration with other tools or libraries. In particular, exporting collected data from the Historian to a *comma separated value* file format currently requires manual intervention for each tag

value. While it might be possible to write own connectors to the used database in the future, the effort to obtain the measured data from the DB is exceeding our expectations. We would recommend others to address similar problems already in the specification stage of future testbeds.

D. Sensor Availability

Several limitations of SWaT have been observed during the course of experimentation to assess the effectiveness of methods to detect cyber attacks. While SWaT consist of a rich set of sensors, not all stages are equipped adequately. For example, there is no pH sensor at the output of the UF unit (see Figure 2). The lack of this sensor requires the use of a pH sensor immediately following the UV dechlorinator. Thus, the impact of UF on water pH cannot be measured directly. This might impact system identification studies where a linear dynamical model of only stages P1 through P3 is to be constructed. For future testbeds, it is thus advisable to "over-instrument" the testbed to allow for more flexibility in experimentation and re-configuration. We note that there are several trade-offs to consider in that context, in particular, physical limitations to the number of sensors that can be used without the sensors themselves influencing the process. As a result, sensors such as flow meters cannot just be inserted every 30 cm of pipe section. In the context of water systems, many sensors must also be placed at specific locations to obtain representative values. For example, sensors should have a certain distance to pipe bends, or other sensors or actuators. In addition, industrial sensors have non-negligible hardware cost, and require wiring, IO slots on the PLCs/RIOs, and appropriate programming of the PLCs.

E. Raw Water

The current design takes in raw water (stage P1) from the campus water line. This water is pure in the sense that it is drinkable. To conduct experiments that detect attacks aimed at impacting water quality, it would be helpful to add water of quality similar to what is input to the treatment system in a selected city, e.g., Singapore. Doing so is not possible in SWaT as the overall system is designed for certain minimal quality of the water in tank T101. The design is currently being modified to make it possible to add significantly impure water to T101 without any damage to the remaining sub-processes.

VI. Conclusion and Future Work

A number of ICS security related projects in iTrust are currently using SWaT. In addition collaborators from organizations within and outside of Singapore have begun using SWaT. To make collaboration easier, it is proposed that access to SWaT be online thus allowing authorized researchers to access it from anywhere. Obviously, doing so comes with its own challenges such as secure access, visibility into every system component, 24/7 availability, etc.

Currently, the treated water in SWaT is recycled within the treatment process itself. In the future, the product water of SWaT will also be used as input water for a second testbed, which is currently under construction. That second testbed will focus on a water distribution testbed. That interconnection will allow the assessment of impact of attacks propagation multiple testbeds.

Cascading effects of cyber attacks across multiple ICS is a challenging research problem. It is planned that the electric power testbed under design in iTrust will be linked to SWaT. Doing so will allow experimentation to assess the impact of cyber attacks on the power grid on the operation of SWaT. The ICS interconnection will also make it feasible to study the impact of multiple simultaneous attacks on two ICS.

Acknowledgements

This work was supported in part by a grant from the Ministry of Defense, Singapore. Thanks to Sridhar Adepu and Kaung Aung for collecting and sharing some of the data and SWaT related information included in this paper, and Nicolas Iooss for experimental work on the attacks.

References

[1] D. Antonioli and N. O. Tippenhauer. MiniCPS: A toolkit for security research on CPS networks. In *Proceedings of Workshop on Cyber-Physical Systems Security & Privay (SPC-CPS), co-located with CCS*, Oct. 2015.

[2] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry. Attacks against process control systems: Risk assessment, detection, and response. In *ACM Symp. Inf. Comput. Commun. Security*, 2011.

[3] Ettercap Project. Ettercap. https://ettercap.github.io/ettercap/.

[4] D. Hadžiosmanović, R. Sommer, E. Zambon, and P. H. Hartel. Through the eye of the PLC: Semantic security monitoring for industrial processes. In *Proceedings of the 30th Annual Computer Security Applications Conference*, pages 126–135, New York, NY, USA, 2014. ACM.

[5] http://www.ni.com/labview/.

[6] Y. Mo and B. Sinopoli. Secure control against replay attacks. In *47th Annual Allerton Conference on Communication, Control, and Computing, 2009. Allerton 2009*, pages 911–918, 2009.

[7] T. Morris, A. Srivastava, B. Reaves, W. Gao, K. Pavurapu, and R. Reddi. A control system testbed to validate critical infrastructure protection concepts. *International Journal of Critical Infrastructure Protection*, 4(2):88 – 103, 2011.

[8] ODVA. Ethernet/IP technology overview. https://www.odva.org/Home/ODVATECHNOLOGIES/EtherNetIP.aspx.

[9] M. Ossmann. Throwing star LAN tap. https://greatscottgadgets.com/throwingstar/.

[10] B. Paske, B. Green, D. Prince, and D. Hutchison. Design and construction of an industrial control system testbed. In *PG Net - The 15th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting*, 2014.

[11] V. Roth, W. Polak, E. Rieffel, and T. Turner. Simple and effective defense against evil twin access points. In *Proceedings of the first ACM conference on Wireless network security*, pages 220–235. ACM, 2008.

[12] G. Sabaliauskaite, G. S. Ng, J. Ruths, and A. Mathur. Experimental evaluation of stealthy attack detection in a robot (in press). In *The 21st IEEE Pacific Rim International Symposium on Dependable Computing (PRDC 2015)*, 2015.

[13] http://www.mathworks.com/products/simulink/.

[14] D. Urbina, J. Giraldo, N. O. Tippenhauer, and A. Cárdenas. Attacking fieldbus communications in ICS: Applications to the SWaT testbed. In *Proceedings of Singapore Cyber Security Conference (SG-CRC)*, Jan. 2016.

[15] S. Weerakkody, Y. Mo, and B. Sinopoli. Detecting integrity attacks on control systems using robust physical watermarking. In *IEEE 53rd Annual Conference on Decision and Control (CDC)*, pages 3757–3764, Dec 2014.

[16] L. Xie, Y. Mo, and B. Sinopoli. False data injection attacks in electricity markets. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 226–231, 2010.