iTrust
Centre for Research
in Cyber Security

It is important that iTrust's triple-defence mechanisms are assessed by independent teams consisting of people well versed in the design and launch of cyber attacks.

S3-17: SUTD Security Showdown

Event Report

November 1, 2017

Francisco FURTADO, Lauren GOH, Sita RAJAGOPAL, Elaine CHEONG, Ericson THIANG

Anonymised Version.

Identities of Defence Teams are Not Included

iTrust
Centre for Research
in Cyber Security

SUTD Security Showdown (S^3-17)

Event Report

Event Date	:	8-9 June 2017
Venue	:	SUTD Campus, Building 2, Level 7, 2.705, Secure Water Treatment Testbed
Reported by	:	Francisco FURTADO, Lauren GOH, Sita RAJAGOPAL, Elaine CHEONG, Ericson THIANG
Report updated	:	October 2, 2017-10-02
Reviewed by	:	TOH Jing Hui
Edits	:	Ivan LEE
Final edits	:	Aditya MATHUR
Organiser	:	iTrust, Centre of Research in Cyber Security
Supported by	:	Ministry of Defence, Singapore
Attack teams	:	Good Hacker's Alliance (GHA), South Korea; LosFuzzys (Graz University of Technology); H4x0rPsch0rr (Technical University of Munich), Germany ; MWR (Lancaster University), United Kingdom ; Ox002147 (University of Oxford), United Kingdom
Defence teams	:	WaterDefense (SUTD), Singapore; WaterDefense_Historian (SUTD), Singapore; Company_A, Russia; Company_B, Israel

2017 S^3 Organisers

Daniele Antonioli, John Henry Castellanos Alvarado, Juan David Guarnizo Hernandez, Sandra Siby, Ahnaf Siddiqi, Hamid Ghaeini, Ragav Sridharan, Francesco Scandola, Amit Subhashchandra Tambe, Randolph Wong, Sridhar Adepu, **Nils Ole Tippenhauer**, **Martin Ochoa**, Kaung Myat Aung, Muhamed Zhaffi Bin Mohamed Ibrahim

2016 S^3 Organisers

Daniele Antonioli, Hamid Reza Ghaeini, Sridhar Adepu, **Martín Ochoa**, **Nils Ole Tippenhauer**, Kaung Myat Aung, Muhamed Zhaffi Bin Mohamed Ibrahim

Contents

1 Introduction

Research in iTrust is aimed at the development of methods and supporting tools to aid in the design of secure critical infrastructure. Such infrastructure must be resilient to cyber attacks. Resiliency requires integration into the infrastructure software and hardware devices for preventing attackers from entering a plant, detecting attacks in the event the prevention mechanism has been bypassed, and ensuring that doubly authenticated commands are allowed to pass to actuators such as pumps, generators, and circuit breakers. Researchers at iTrust engage in research and development activities aimed at the creation of a robust and practical triple-defence approach that includes prevention, detection, and control in the face of cyber and cyber-physical attacks.

While researchers design and perform experiments to assess the effectiveness of various components of the triple-defence mechanisms they develop, it is important that such assessment be also carried out by independent teams consisting of people well versed in the design and launch of cyber attacks. It is with this goal in view that iTrust began organising the SUTD Security Showdown event. This event, dubbed as S^3, was first held in June 2016 at iTrust. It is organised by a team consisting of faculty, research staff, and administrative staff in iTrust. Several attack and defence teams are invited to iTrust to participate in S^3. Two such events have been organised so far, one in 2016 and the other in 2017. This report focuses on the organisation of the S^3-17 event and the performance of various attack and defence teams.

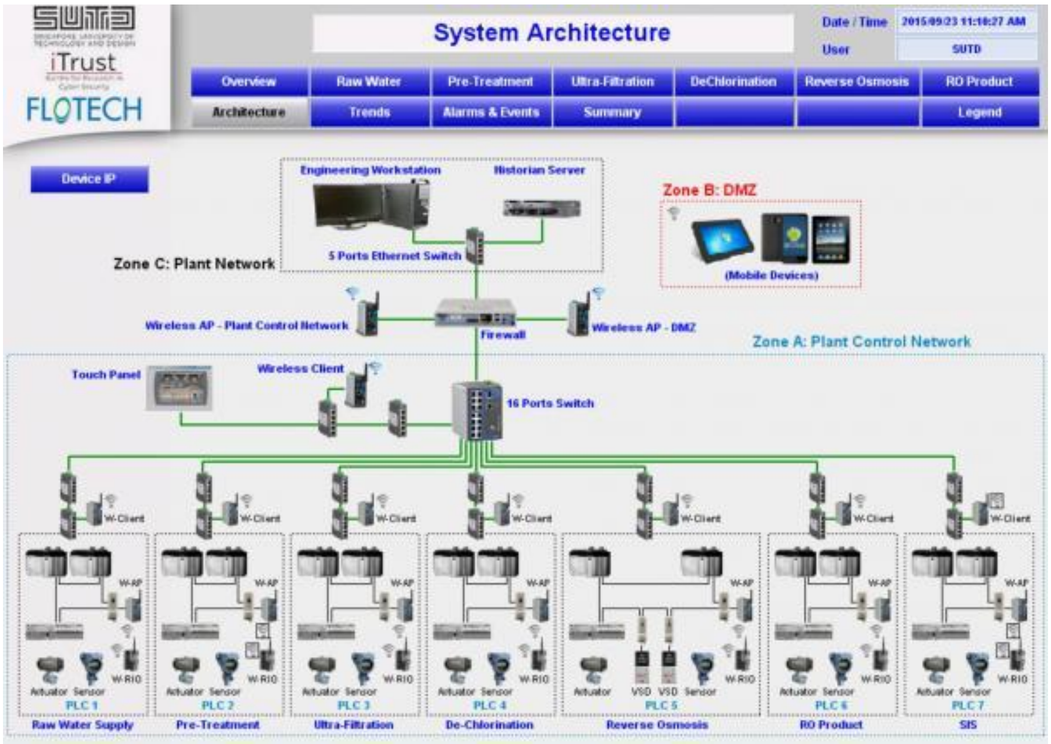
The Ministry of Defence, Singapore, and the SUTD-MIT International Design Centre, funded the S3-17 event. The event consists of two key phases – an online qualifier and a live event held at iTrust. Following the online qualifier, five international teams were invited to participate in the live event. Each team was given the opportunity to design attacks against a realistic testbed, namely Secure Water Treatment (SWaT). The goal of each attack team was to meet as many pre-defined challenges as possible within the pre-allocated time.

2 Live Phase Setup

2.1 Platforms used

2.1.1 Network Architecture

Details of the architectures of SWaT and WADI can be found here. However, as a means of demonstrating the scale of these testbeds, and framing the discussion of the attacks conducted, SWaT is discussed briefly below. The architecture of SWaT is as follows.



Date / Time 2015/09/23 11:10:27 AM

User SUTD

Figure 1:Network Architecture of SWaT

SWaT consists of a modern six-stage process. The process begins by taking in raw water, adding necessary chemicals to it, filtering it via an Ultrafiltration (UF) system, de-chlorinating it using ultra-violet (UV) lamps, and then feeding it to a Reverse Osmosis (RO) system. A backwash process cleans the membranes in UF using the water produced by RO. The cyber portion of SWaT consists of a layered communications network, Programmable Logic Controllers (PLCs), Human Machine Interfaces (HMIs), Supervisory Control and Data Acquisition (SCADA) workstation, and a Historian. Data from sensors is available to the SCADA workstation and is recorded at the Historian for subsequent analysis.

2.1.1.1 Layer 1 (L1)– Plant Control Network

Layer 1 refers to the communication infrastructure that enables communications among the PLCs responsible for controlling the entire plant. It is implemented using star topology. Devices on the network include PLCs, a SCADA workstation, HMI and the Historian.

2.1.1.2 Layer 0 (L0) – Process

Layer 0 refers to the communication infrastructure at each stage of SWaT between sensors, actuators and the PLC that controls the sub-process at that stage. “Device Level Ring” which also includes a Remote IO (RIO) device implements it. The RIO rather than the PLCs are connected to the physical sensors and actuators, with monitoring and control information being sent across the Distributed Logical Router (DLR). The ring topology allows active PLC controller to serve as “Ring Supervisor” and is able to tolerate single-node failure.

SWaT is equipped with Allen Bradley ControlLogix PLCs. Therefore, some of the attacks described below required consideration of the protocols used by these components: EtherNet/IP for Allen Bradley PLCs and Modbus over TCP for the Schneider RTUs.

2.2 Setup for Attackers

From the 3rd to the 7th of June, each attack team was given two sessions of four hours each to conduct reconnaissance on the testbeds. During these sessions various active attacks were prepared and tested with the assistance of the SWaT laboratory engineer.

During the actual event, held on the 8th and 9th of June, each team was given two hours to demonstrate their attacks prepared in advance. These attacks were graded using the scoring system described in Section 3. Attack teams were also given a separate network for Internet uplink and up to three Virtual Machines (VMs) running either Linux or the Windows operating system. Each team could choose to adopt different attacker profiles (please refer to Section 3.3) that position them at various points in the network for launching the attacks.

Teams were allowed to interact with and attack nearly all of the testbed except for the following:

a) the server in the control room was not to be attacked by physical means;
b) the Historian cannot be compromised directly though manipulation of data sent to the Historian was allowed; and
c) attacks cannot affect the overall setup of SWaT on a scale that affects items located outside the physical limits of the testbed (e.g., trigger university wide fire alarm).

2.3 Setup for Defenders

Three teams, throughout the event, deployed four defence systems. The objective of the defence systems was to detect and raise alarms upon detection. The following defence systems were deployed: WaterDefense (WD) and WaterDefense Historian (WDH) from iTrust, Industrial Cyber Security (Product_A) from Company_A, and an Intrusion Detection System (IDS) from Company_B, Israel. WD and WDH are attack detection systems installed, respectively, inside each PLC and at the Historian. Product_A and Company_B are advanced IDS.

IDS are considered a second line of defence in the Cyber Physical System (CPS); firewalls being the first. IDS are designed to detect the tampering of CPS processes after the attackers have successfully obtained network or physical access into the system, beyond traditional network-centric infrastructure such as the firewall, or physical security. In its totality, an IDS detects intrusions, and is not equipped with control strategies to mitigate the impact of cyber-attacks. The defence teams were only required to install an IDS, without any counter-attack add-ons. This isolated the S^3-17 attacker teams during their stipulated time-slots from any possible interferences of the defence systems, thereby ensuring fair judging. WD and WDH are process anomaly detectors and focus, at all times, on process state and not network traffic.

3 Scoring

Points were awarded if the attacker was able to undo the impact of an attack (to minimise any risk of permanent damage). Equation 1 below defines how an attack was awarded:

$$s = g * c * p \qquad , \qquad (1)$$

where s is the final score, g denotes the base value of the goal, c a control modifier to value the level of control the attacker has, and p is the attacker profile modifier. Most modifiers were in the range [1,2], while the base value for the targets was in the range [100,200]. The attackers get points for each attack launched. However, if more than one attack is performed successfully on a similar goal, the highest score for each goal is used as the final score. For example, if an attack on a pump was successful using both the strong attacker model (e.g., 130 points), and the cybercriminal attack profile (e.g., 200 points), then 200 points will be counted for that category (attack on pump). Goals are pre-defined and can be selected from two separate lists, namely the physical process and sensor data.

3.1 Goals $[g]$

3.1.1 Physical Process Goals: Control over a physical actuator or the process

a) 100 points: Motorised Valves (open/close/transitioning/intermediate)
b) 130 points: Water Pumps (on/off)
c) 145 points: Pressure
d) 160 points: Water Tank Level (true water amount, not sensor reading)
e) 180 points: Chemical dosing

3.1.2 Sensor Data Goals: Demonstrate control over sensor readings at different components

a) 100 points: Historian values
b) 130 points: HMI/SCADA values
c) 160 points: PLC values
d) 200 points: Remote I/O values

3.2 Control modifiers $[c]$

The control modifier determines the amount of control precision the attacker has during the execution of an attack. As a guideline, the modifier is:

a) 0.2 if the attacker can randomly (value and time) influence the process, and
b) 1.0 if the attacker can precisely influence the process or sensor value to a target value chosen by the judges.

3.3 Attacker Profile $[p]$

3.3.1 Cybercriminal Attacker Model (Factor 2)

In the cybercriminal attacker model an attacker was assumed to have an average knowledge of the system and good knowledge of offensive capabilities such as ARP (Address Resolution Protocol) poisoning, exploits, and brute force attacks. While a cybercriminal has access to L1 network through a compromised machine, they do not have access to L0 ring network. They also do not have access to Industrial Control System (ICS) specific tools such as Studio 5000 (IDE used to configure PLCs in SWaT) or access to testbed administrator accounts. This model yields the highest factor due to its highly complex and difficult attack position outside the end-point protection system.

ARP poisoning is a type of attack in which a malicious actor sends falsified ARP messages over a Local Area Network (LAN). This resulted in the linking of an attacker's media access control address (MAC address) with the IP address of a legitimate computer or server on the network. Once the attacker's MAC address is connected to an authentic IP address, the attacker can begin receiving any data that is intended for that IP address.

3.3.2 Insider Attacker Model (Factor 1.5)

The insider attacker model was assumed to be a disgruntled employee with good knowledge of the system including administrator passwords and the ability to operate the HMI, but with no prior experience in launching attacks, and limited computer science skills. The insider also has physical access to the system where control valves and network topology can be manipulated. In addition, such an attacker has access to ICS specific tools such as Studio 5000.

3.3.3 Strong Attacker Model (Factor 1)

The strong attacker model effectively combined the cybercriminal and insider attacker models resulting in the most advantageous attacker model with a yield factor of 1. This model assumes that the attacker has the most information about, and directs access to, the ICS. This attacker under this model has the least challenges in gaining entry into the ICS and, hence, yields the lowest factor.

4 Description of Attacks Launched

4.1 GHA (Good Hackers Alliance), South Korea

4.1.1 Insider Attacker Model

4.1.1.1 Control of the Motorised Valve through Manual Intervention

Objective : Change the motorised valve status

Attack method : Manual manipulation

Tools : Nil

Description : The intention of this attack was to manually change the status of the motorised valve through manual operation mode. As an insider, the attacker had access to the motorised valve and knowledge of how to operate it. As such, the insider turned the valve to manual operation mode and then disrupted the status of the valve's control.

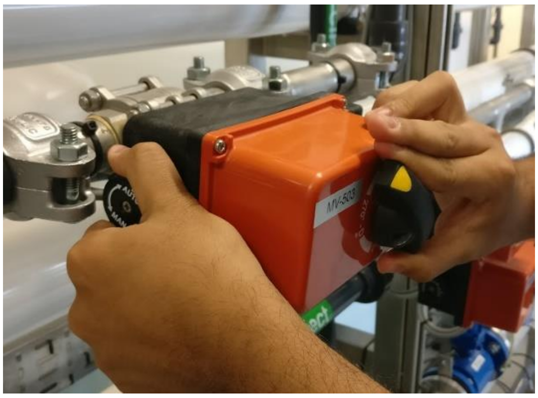


Figure 2: Changing the motorised valve to manual operation mode

4.1.2 Cybercriminal Attacker Model

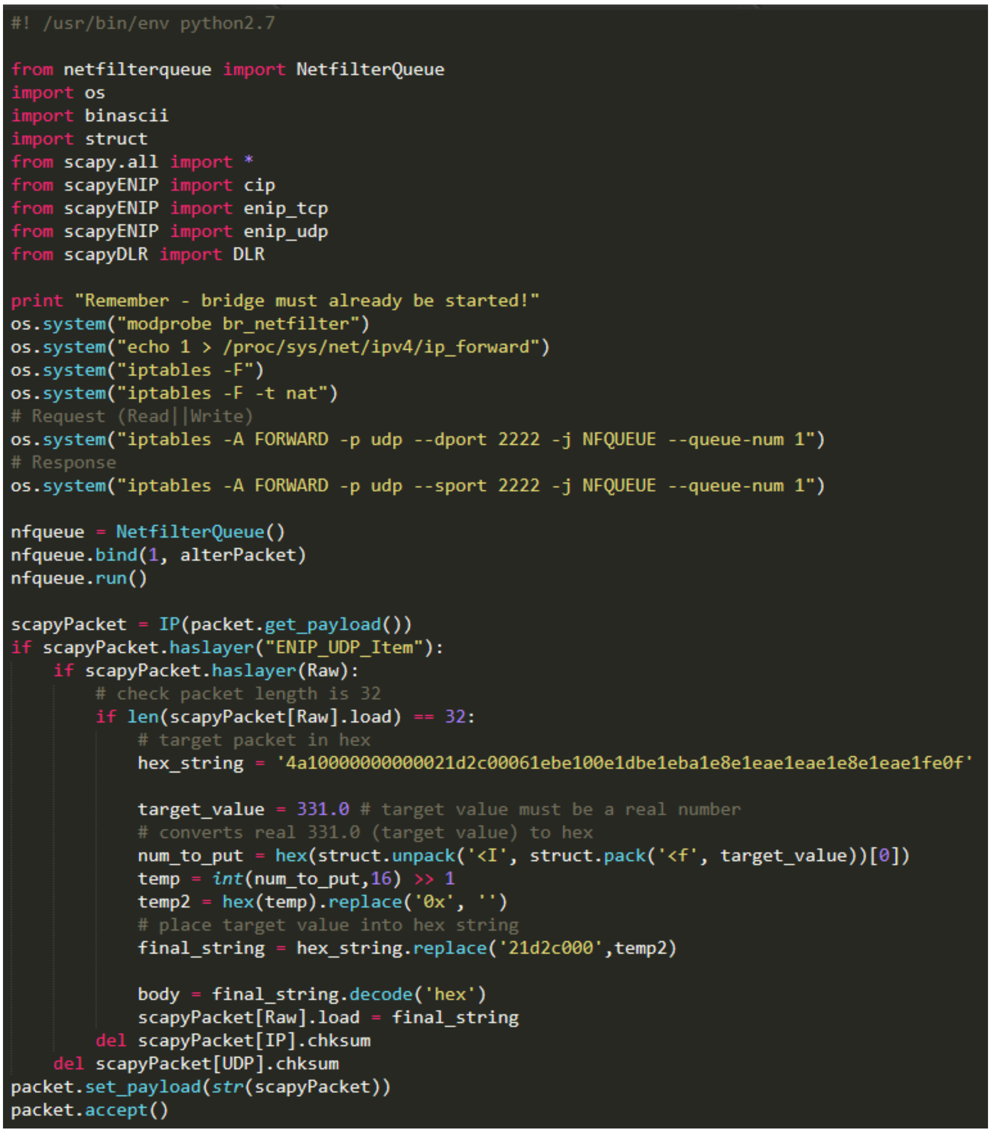
4.1.2.1 Control of the PLC through the Bridged Man-in-the-Middle (MiTM) at Level 0

Objective : Alter the commands send by a PLC and manipulate sensor values received by a PLC

Attack method : Bridged Man-in-the-Middle (MiTM)

Tools : NetFilterQueue, Scapy

Description : This attack was to change the values and commands which the PLC receives and sends, respectively. The attack was conducted at two levels. At level 0 the attacker placed a device between the RIO and the PLC thus creating a bridge between the two network interfaces. An analysis of the network traffic revealed the packets that the MiTM should edit. As the target was the tank water level, the MiTM sets it to a constant value to hide the increasing water level of the tank. Before the packet is forwarded, NetFilterQueue reroutes the targeted packet into a queue which can be read and modified by scripts. To prevent all packets from going into the queue, in order not to disrupt other processes, IPTABLES is used to identify targeted packets entering the queue. Using Scapy and a custom dissector previously developed by SUTD, the attacker edited the payload of the targeted packet which was then forwarded to its original destination. Figure 3 illustrates an example of an MiTM script. The script works on both Level 0 and Level 1 as EtherNet/IP is used to transmit the packets.



```
#! /usr/bin/env python2.7

from netfilterqueue import NetfilterQueue
import os
import binascii
import struct
from scapy.all import *
from scapyENIP import cip
from scapyENIP import enip_tcp
from scapyENIP import enip_udp
from scapyDLR import DLR

print "Remember - bridge must already be started!"
os.system("modprobe br_netfilter")
os.system("echo 1 > /proc/sys/net/ipv4/ip_forward")
os.system("iptables -F")
os.system("iptables -F -t nat")
# Request (Read||Write)
os.system("iptables -A FORWARD -p udp --dport 2222 -j NFQUEUE --queue-num 1")
# Response
os.system("iptables -A FORWARD -p udp --sport 2222 -j NFQUEUE --queue-num 1")

nfqueue = NetfilterQueue()
nfqueue.bind(1, alterPacket)
nfqueue.run()

scapyPacket = IP(packet.get_payload())
if scapyPacket.haslayer("ENIP_UDP_Item"):
    if scapyPacket.haslayer(Raw):
        # check packet length is 32
        if len(scapyPacket[Raw].load) == 32:
            # target packet in hex
            hex_string = '4a1000000000000021d2c00061ebe100e1dbe1eba1e8e1eae1eae1e8e1eae1fe0f'

            target_value = 331.0 # target value must be a real number
            # converts real 331.0 (target value) to hex
            num_to_put = hex(struct.unpack('<I', struct.pack('<f', target_value))[0])
            temp = int(num_to_put,16) >> 1
            temp2 = hex(temp).replace('0x', '')
            # place target value into hex string
            final_string = hex_string.replace('21d2c000',temp2)

            body = final_string.decode('hex')
            scapyPacket[Raw].load = final_string
        del scapyPacket[IP].chksum
    del scapyPacket[UDP].chksum
packet.set_payload(str(scapyPacket))
packet.accept()
```

Figure 3: Bridged MiTM script targeting EtherNet/IP

4.1.2.2 Control of the chemical dosing system through a Python script (pycomm)

Objective : Change chemical dosing

Attack method : Compromised Virtual Network Computing (VNC)

Tools : Python script (pycomm), Wireshark

Description : The intention of this attack was to change the chemical dosing at the end of the De-chlorination System (P4). After gaining access to the HMI through the compromised VNC, the cybercriminal attacker captured packets between the HMI and PLC4 using Wireshark. Through analysis of the packets, the attacker retrieved the controller tag that influences the PLC used for chemical dosing. The attackers changed the tag value to control the chemical dosing function using the pycomm framework.

4.1.2.3 Control of the Historian through the Aircrack WiFi

Objective : Compromise Historian data

Attack method : Crack WiFi password, ARP poisoning, MiTM payload manipulation

Tools : Aircrack, Ettercap

Description : The intention of this attack was to compromise the data stored in the Historian. As the PLC was in the wireless mode, the cybercriminal attacker used Aircrack to obtain the password for connecting to the ICS Access Point (AP). ARP poisoning was executed to reroute traffic between the PLC and Historian through the attacker's rogue terminal. The attacker then used an Ettercap filter to manipulate the network packets. The attackers changed a small part of the payload to an arbitrary value before releasing the packets to the Historian.

4.1.2.4 Control of the pressure through the Server Message Block (SMB)

Objective : Disrupt valves operation of Ultrafiltration and Backwash (P3)

Attack method : Vulnerability CVE-2008-2160

Tools : Nil

Description : The intention of this attack was to change the pressure at P3. The cybercriminal attacker used Microsoft's Server Message Block (SMB) to obtain files from the HMI. As the HMI was running Windows CE, it has a vulnerability that allows the attacker's terminal to execute arbitrary code on the HMI (CVE-2008-2160). As such, the attacker was able to retrieve the files to create a copy of the workstation. From this copied workstation, the attacker manually changed the valves in P3 such that the pressure at Differential Pressure Indicating Transmitter 301 (DPIT301) became dangerously high. This led to Motorised Valve 301 (MV301) closed, MV302 closed, MV303 closed and MV304 open. Figure 4 below shows the diagram of P3 where different valves were affected during the attack.

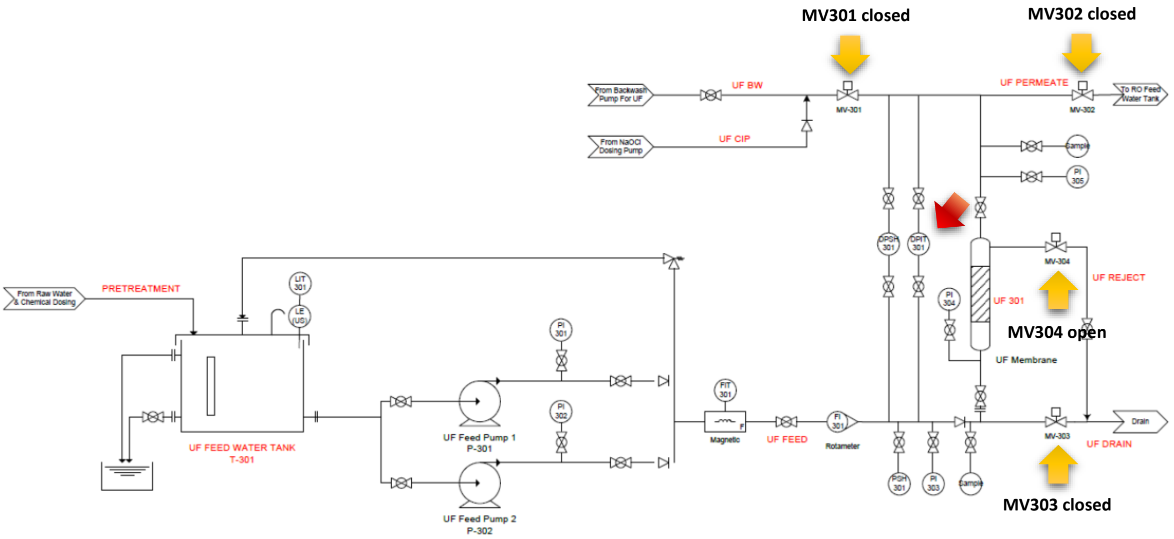


Figure 4: Diagram of illustrating MV301, MV302, MV303, MV304 and DPIT301 in P3

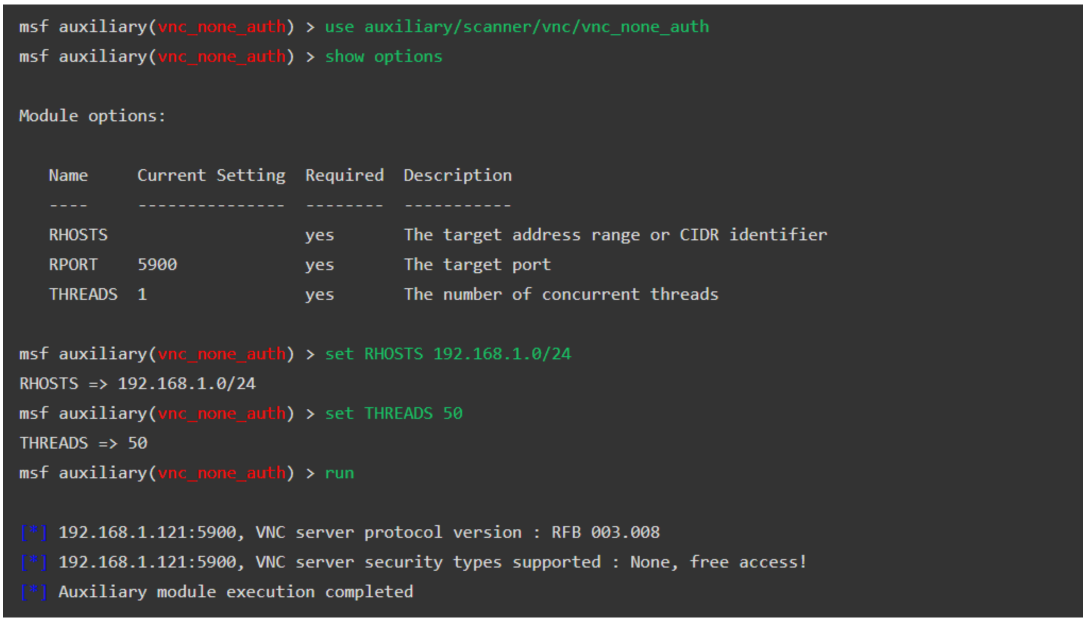
4.1.2.5 Control of the water level in the tank through the Metasploit VNC Scanner

Objective : Change the water level of the tank

Attack method : VNC server without password protection

Tools : Metasploit VNC Authentication None Scanner

Description : The intention of this attack is to change the water level in the tank. The cybercriminal attacker used Metasploit VNC Authentication None Scanner to check for nodes running a VNC Server without a password. Figure 5 illustrates how the scanner was used.



```
msf auxiliary(vnc_none_auth) > use auxiliary/scanner/vnc/vnc_none_auth
msf auxiliary(vnc_none_auth) > show options

Module options:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS                    yes       The target address range or CIDR identifier
   RPORT    5900             yes       The target port
   THREADS  1                yes       The number of concurrent threads

msf auxiliary(vnc_none_auth) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(vnc_none_auth) > set THREADS 50
THREADS => 50
msf auxiliary(vnc_none_auth) > run

[*] 192.168.1.121:5900, VNC server protocol version : RFB 003.008
[*] 192.168.1.121:5900, VNC server security types supported : None, free access!
[*] Auxiliary module execution completed
```

Figure 5: Sample of a Metasploit VNC Scanner

Once the scanner detected the VNC Server was running without any authentication, the attacker penetrated the server through a VNC Client connection. As the VNC Server was hosting the HMI which controlled the ICS, the attacker changed the simulation tag of the water level of the tank.

4.1.2.6 Control of the pump through a rogue router

Objective : Disrupt pump control operation

Attack method : Evil twin (rogue access point)

Tools : KisMAC, Password cracking tool, 3vilTwinAttacker, Telnet, Scapy

Description : The intention of this attack was to turn on the pump when it should be off. The cybercriminal attacker used KisMAC to scan for wireless networks in the ICS. Once the targeted wireless network was identified, the attacker used dictionary attack to crack the password. As the password contains "sutd," the attacker has a very high success rate by using variations of patterns. After the password was cracked, the attacker created a rogue wireless router with a similar SSID and configuration. It then sent a de-authentication packet to disassociate the PLC and the original router. The attacker used Telnet to log into the original router and shut it down. Scapy was then used to modify the packets to turn the pump on.

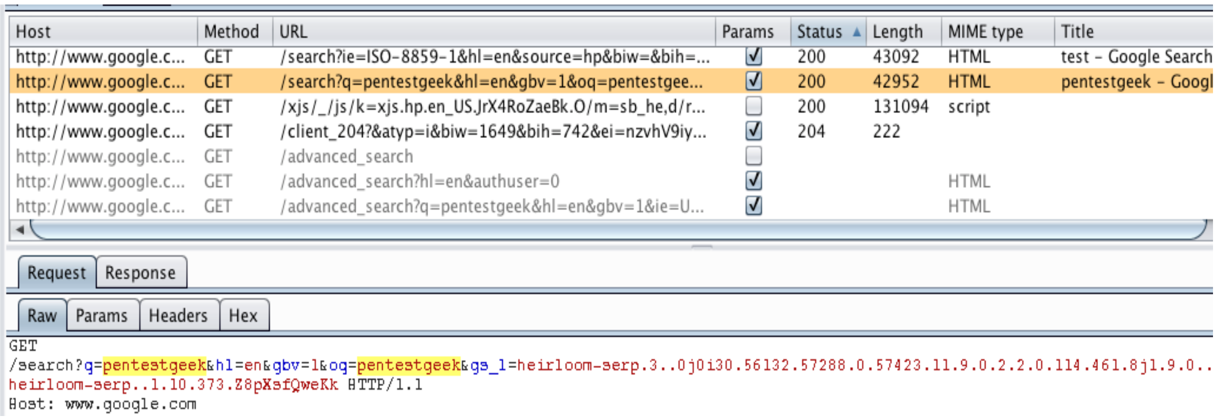
4.1.2.7 Control of the pump through the FactoryTalk and password vulnerability

Objective : Disrupt pump control operation

Attack method : Password policy (no password), MiTM (interception proxy)

Tools : Silverlight, BURPSUITE (Java based Web Penetration Testing framework)

Description : The intention of this attack was to turn on the pump when it should be off. The cybercriminal attacker gained access into the HMI through VNC by exploiting the password policy. With access to the files, the attacker copied the FactoryTalk ViewPoint (FTVP) files and executed a new instance of the HMI using Silverlight. As Silverlight had restricted functions, shutting the down original HMI was not possible. Instead, the attackers used the BURPSUITE proxy framework acting as Man-in-the-Middle to capture and rewrite traffic controlling the pump and turning it on.



Host	Method	URL	Params	Status ▲	Length	MIME type	Title
http://www.google.c...	GET	/search?ie=ISO-8859-1&hl=en&source=hp&biw=&bih=...	☑	200	43092	HTML	test – Google Search
http://www.google.c...	GET	/search?q=pentestgeek&hl=en&gbv=1&oq=pentestgee...	☑	200	42952	HTML	pentestgeek – Googl
http://www.google.c...	GET	/xjs/_/js/k=xjs.hp.en_US.JrX4RoZaeBk.O/m=sb_he,d/r...	□	200	131094	script	
http://www.google.c...	GET	/client_204?&atyp=i&biw=1649&bih=742&ei=nzvhV9iy...	☑	204	222		
http://www.google.c...	GET	/advanced_search	□				
http://www.google.c...	GET	/advanced_search?hl=en&authuser=0	☑			HTML	
http://www.google.c...	GET	/advanced_search?q=pentestgeek&hl=en&gbv=1&ie=U...	☑			HTML	

```
GET
/search?q=pentestgeek&hl=en&gbv=1&oq=pentestgeek&gs_l=heirloom-serp.3..0j0i30.56132.57288.0.57423.11.9.0.2.2.0.114.461.8j1.9.0..
heirloom-serp..1.10.373.Z8pXsfQweKk HTTP/1.1
Host: www.google.com
```

Figure 6: Sample of the BURPSUITE tool

4.2 LosFuzzys (Graz University of Technology), Austria

LosFuzzys is a team of people from Graz University of Technology, Austria, who are interested in information security. They have occasionally participated in Capture-The-Flag (CTF) competitions since 2014.

4.2.1 Insider Attacker Model

4.2.1.1 Control of the RIO/Display through manual configuration on the sensor

Objective : Change the pH value shown at HMI

Attack method : Manual manipulation

Tools : Nil

Description : The intention of this attack was to falsify the pH value of the water shown at the HMI. As an insider, the attacker had access to the physical sensors in the plant. From the pH sensor, the analogue output of minimum (*Aout_min*) and maximum (*Aout1_max*) reading can be configured manually through the pH device. The pH device obtained its reading from the physical pH sensors. The default pH settings used were *Aout_ min* = 2 and *Aout1_max* = 12. By changing these settings manually at the pH device, only the values displayed on the HMI would be affected. The actual pH value of the physical sensor remained unchanged. When the *Aout_min* increased, the pH value at the HMI decreased. When the *Aout1_max* is decreased, the displayed pH value at the HMI increased. With proper calculation, the displayed pH value at the HMI can be adjusted. Thus, the actual pH value of the water was different from the pH value shown at the HMI display.

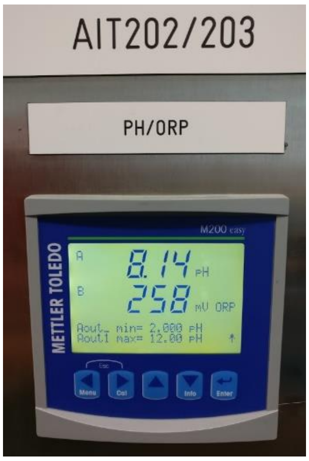


Figure 7: AIT202 analogue output Min/Max values of the pH device

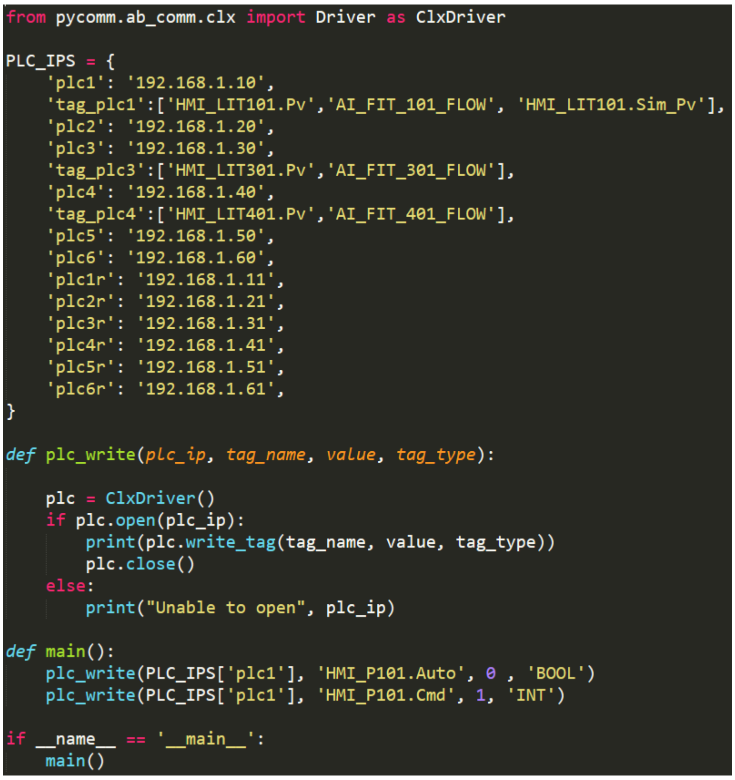
4.2.1.2 Control of the water pump P101 through the Python script (pycomm)

Objective : Change the pump values sent to PLC

Attack method : Manual manipulation, MiTM@SCADA

Tools : Python script (pycomm)

Description : The intention of this attack was to change the pump status. As an insider, the attacker had access to the SCADA workstation. From there, a script was executed to affect the system. *pycomm* is a Python package that has a collection of modules that allow communication with PLCs. To communicate with the Controllogix of the PLCs in SWaT, the clx class in ab_comm module was used. Four items were needed to use this class to modify the pump or pressure value: the IP of the target PLC, the controller tag name, the tag type and the new value to write into the tag. With this script shown below, the attacker can change the pump status.



```python
from pycomm.ab_comm.clx import Driver as ClxDriver

PLC_IPS = {
    'plc1': '192.168.1.10',
    'tag_plc1':['HMI_LIT101.Pv','AI_FIT_101_FLOW', 'HMI_LIT101.Sim_Pv'],
    'plc2': '192.168.1.20',
    'plc3': '192.168.1.30',
    'tag_plc3':['HMI_LIT301.Pv','AI_FIT_301_FLOW'],
    'plc4': '192.168.1.40',
    'tag_plc4':['HMI_LIT401.Pv','AI_FIT_401_FLOW'],
    'plc5': '192.168.1.50',
    'plc6': '192.168.1.60',
    'plc1r': '192.168.1.11',
    'plc2r': '192.168.1.21',
    'plc3r': '192.168.1.31',
    'plc4r': '192.168.1.41',
    'plc5r': '192.168.1.51',
    'plc6r': '192.168.1.61',
}

def plc_write(plc_ip, tag_name, value, tag_type):

    plc = ClxDriver()
    if plc.open(plc_ip):
        print(plc.write_tag(tag_name, value, tag_type))
        plc.close()
    else:
        print("Unable to open", plc_ip)

def main():
    plc_write(PLC_IPS['plc1'], 'HMI_P101.Auto', 0 , 'BOOL')
    plc_write(PLC_IPS['plc1'], 'HMI_P101.Cmd', 1, 'INT')

if __name__ == '__main__':
    main()
```

Figure 8: Sample code to turn on water pump (P101) from SCADA

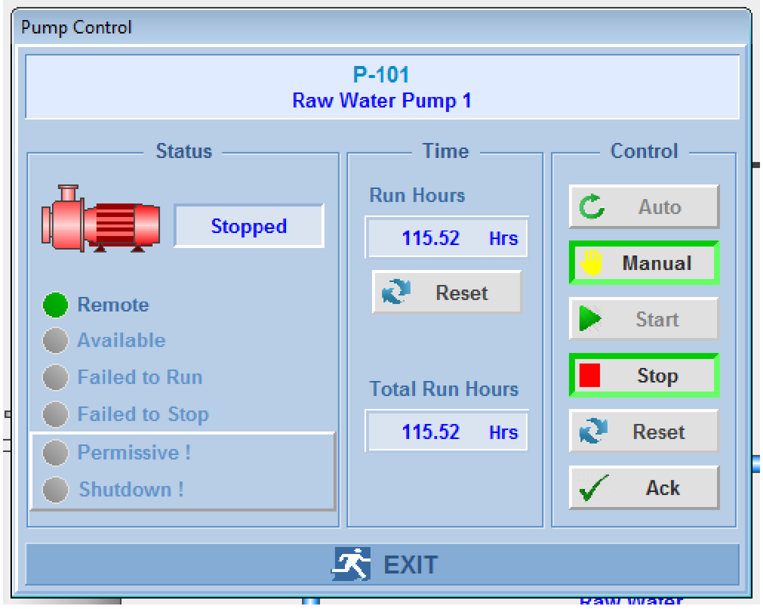
4.2.1.3 Control of the water pump P101 through manual operation of the HMI

Objective : Alternate the state [On:Off] of the water pump

Attack method : Manual manipulation

Tools : Nil

Description : The intention of this attack was to alter the state of the pump P101. By enabling the "manual mode" option through the HMI one could manually operate the pumps in SWaT. As an insider using the HMI, the state of the pump was quickly turned on and off manually. Through these repetitive manual actions, the pump potentially can be fused or explode. In addition, keeping the pump on throughout can cause flooding.



Status Stopped

Run Hours 115.52 Hrs

Total Run Hours 115.52 Hrs

Figure 9: Control options of the water pump P101 at HMI

4.2.1.4 Control of the pressure pump through Python script (pycomm)

Objective : Increase the pressure at P3

Attack method : Manual manipulation

Tools : Python script (pycomm)

Description : The intention of this attack was to turn on two pumps (P301 and P302) while the motorised valve (MV302) was closed. Using the same pycomm framework in 4.2.1.2, the insider attacker modified the pump and valve commands. As the valve was closed when the pumps were on, the pressure at DPIT301 increased to a dangerous level.

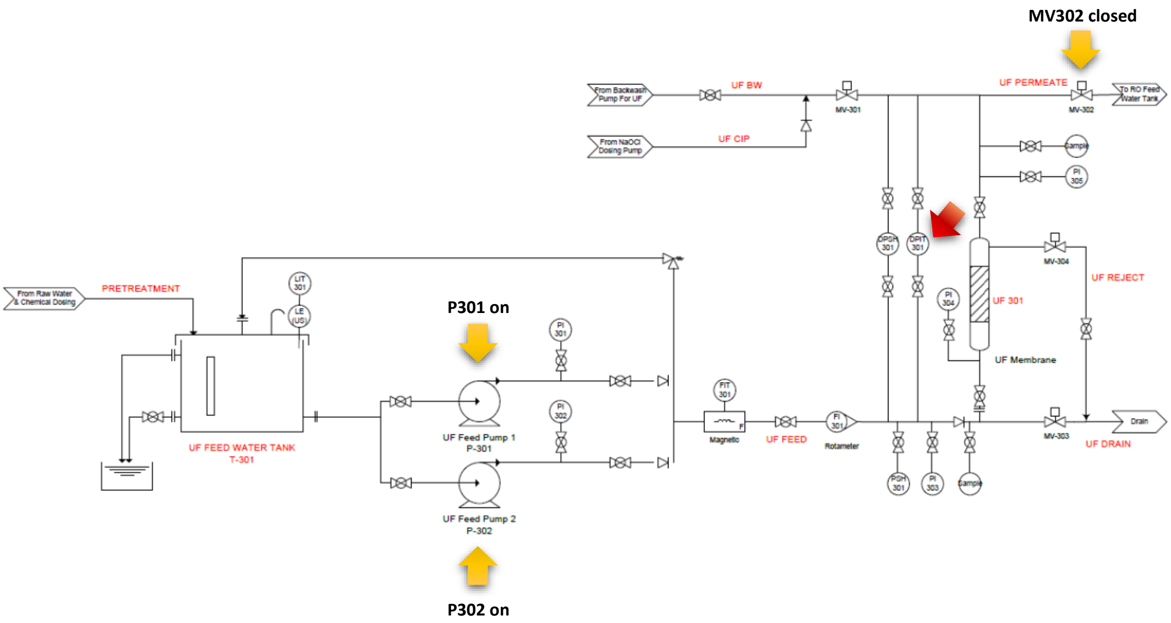


Figure 10: Diagram illustrating P301, P302, MV302 and DPIT301 in P3

4.2.2 Cybercriminal Attacker Model

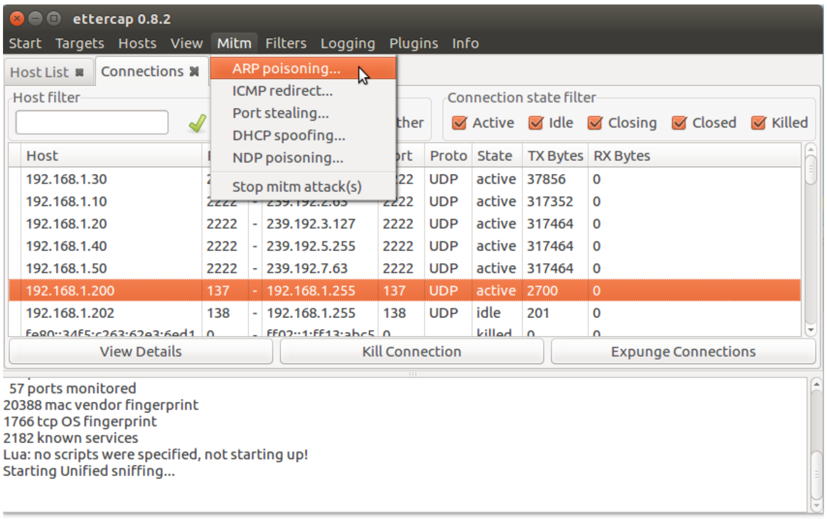
4.2.2.1 Control of the PLC through Man-in-The-Middle (MiTM) with ARP

Objective : Increase the pressure at P3

Attack method : ARP poisoning, MiTM

Tools : Ettercap

Description : The intention of this attack was to maintain the pump (P101) running even though the valve (MV201) was closed. The cybercriminal attacker intercepted the communication between PLC1 and PLC2 using MiTM. To achieve this, ARP poisoning was conducted. This caused the network packets transmitted between PLC1 and PLC2 to be rerouted to the cybercriminal attacker's terminal. The rerouting was achieved using Ettercap--an open source network security tool for MiTM attacks on LAN.



Host	I			rt	Proto	State	TX Bytes	RX Bytes
192.168.1.30	2			22	UDP	active	37856	0
192.168.1.10	2222	-	239.192.2.63	2222	UDP	active	317352	0
192.168.1.20	2222	-	239.192.3.127	2222	UDP	active	317464	0
192.168.1.40	2222	-	239.192.5.255	2222	UDP	active	317464	0
192.168.1.50	2222	-	239.192.7.63	2222	UDP	active	317464	0
192.168.1.200	137	-	192.168.1.255	137	UDP	active	2700	0
192.168.1.202	138	-	192.168.1.255	138	UDP	idle	201	0
fe80::34f5:c263:62e3:6ed1	0	-	ff02::1:ff13:abc5	0		killed	0	0

```
57 ports monitored
20388 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!
Starting Unified sniffing...
```

Figure 11: Screenshot of the Ettercap before ARP Poisoning

Once the ARP poisoning was successful, the MiTM attack was executed. The cybercriminal attacker modified the network packets from the "closed" status of MV201 to "open." As PLC1 noted that MV201 was open, it sent a command to start pump P101. Without any intervention under increasing pressure condition, this attack could cause the water pipe to burst.

4.2.2.2 Control of the water tank level LIT101 through Python script (pycomm)

Objective : Falsify the water level reading of the tank displayed at SCADA

Attack method : Compromised VNC, unprotected HMI, evil twin

Tools : Python script (pycomm)

Description : The intention of this attack was to control the water level in tank (LIT101) at 500mm instead of 800mm steadily at the SCADA display. Through the compromised VNC and unprotected HMI, cybercriminal attacker gained access and control to the HMI. The attacker then changed the IP of their computer to the IP of the HMI. This was achieved using the Python script (pycomm) as the evil twin to adjust the alarm set points.

4.2.2.3 Control of chemical dosing through modified PLC Logic

Objective : Change the level of the chemical used for dosing

Attack method : Compromised VNC, unprotected HMI, manual manipulation

Tools : Studio 5000

Description : The intention of this attack was to change the level of the chemicals used for dosing during the pre-treatment phase (stage 2 of SWaT). From the compromised HMI achieved as in the previous attack, the attacker used Studio 5000 to change the level of the chemical dosage.

4.3 H4x0rPsch0rr (Technical University of Munich), Germany

This is a group of students from TUM (Technische Universität München) and play CTFs for fun.

4.3.1 Insider Attacker Model

4.3.1.1 Control of the RIO through disconnecting Analogue Input/Output pin

Objective : Disrupt the sensor reading send to PLC through remote I/O (RIO)

Attack method : Manual manipulation

Tools : Nil

Description : The intention of this attack was to disrupt the sensor values to the RIO. As an insider, by reading the Operation Manual, the specific I/O pin was identified and disconnected manually.

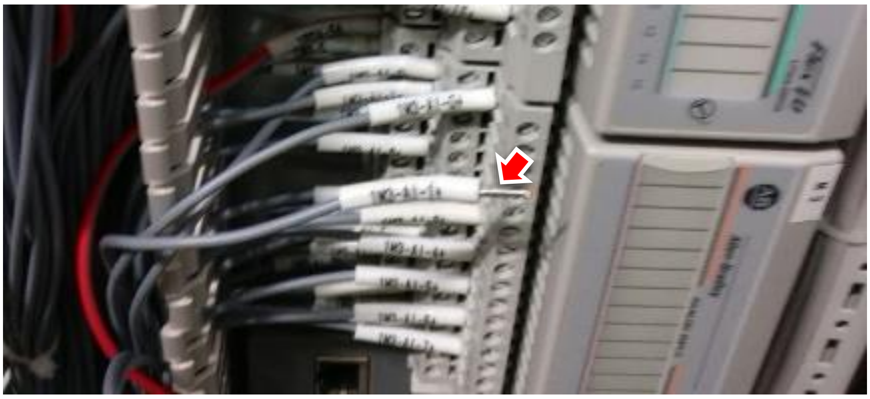


Figure 12: Disconnected AI/O pin

4.3.1.2 Control of the water pump P101 through manual operation of the HMI

Please refer to 4.2.1.3 for a similar attack model.

4.3.2 Cybercriminal Attacker Model

4.3.2.1 Control of the amount of chemical dosing through Python script

Objective : Increase the chemical dosage during water pre-treatment phase

Attack method : MiTM

Tools : Python scripts

Description : The intention of this attack was to control the level of chemical dosing of the water during the pre-treatment phase. As a cybercriminal attacker, two Python scripts were used to enumerate the tags and send new control values to toggle the pump state.

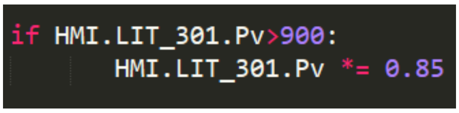
4.3.2.2 Control of the PLC through the modification of PLC logic in Studio 5000

Objective : Falsify water level display at SCADA even though the water level was at extremely full level

Attack method : Manual manipulation

Tools : Studio 5000

Description : The intention of this attack was to modify the PLC logic at the SCADA workstation so that the water level display remains normal even though in reality it was at extremely full level. As the attacker had a compromised SCADA workstation, he changed the displayed value of the water tank level by reducing the value if it went above a certain threshold. Hence, the operator would be tricked into thinking that the water level was acceptable when it was not and thus flooding could occur. The pseudo-code is presented below.



```
if HMI.LIT_301.Pv>900:
        HMI.LIT_301.Pv *= 0.85
```

Figure 13: Pseudo-code to falsify the water level of the tank

4.3.2.3 Control of HMI/SCADA through MiTM with ARP

Please refer to 4.2.2.1 for similar attack model.

4.4 MWR (Lancaster University), United Kingdom

Established in 2003, MWR is an independent cyber security consultancy with research at the heart. MWR has a dedicated commitment to research, with each of their consultants given 25% of their time to devote to security research. They investigate new software, hardware or protocols and push the boundaries of what is possible.

4.4.1 Insider Attacker Model

4.4.1.1 Control of the RIO through disconnected Analogue Input/Output pin

Please refer to 4.3.1.1 for similar attack model.

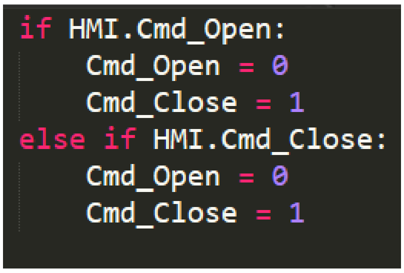
4.4.1.2 Control of the motorised valve through modification of PLC logic in Studio 5000

Objective : Permanently closed the motorised valve regardless of commands issued

Attack method : Manual manipulation

Tools : Studio 5000

Description : The intention of this attack was to disable the motorised valve from opening through the HMI. As an insider attacker, the PLC logic code was modified such that the targeted motorised valve remained permanently closed regardless of command issued from the HMI. The pseudo-code is presented below.



```
if HMI.Cmd_Open:
    Cmd_Open = 0
    Cmd_Close = 1
else if HMI.Cmd_Close:
    Cmd_Open = 0
    Cmd_Close = 1
```

Figure 14: Pseudo-code to disable valves from opening through the HMI

4.4.2 Cybercriminal Attacker Model

Objective : Establish back-door connection

Attack method : Phishing attack

Tools : Mimikatz, Word document with malicious VBA Macro, SOCKS proxy

Description : The cybercriminal attacker performed a phishing attack which targeted the system. A Microsoft Word document that contained a malicious VBA Macro was sent to the SCADA/engineering workstation. The malicious macro executed a PowerShell command to establish a connection between the compromised workstation and the command and control (C2) server. The HTTP connection was established and beacons were constantly emitted to the C2 server for tracking the user's activity, without being detected by outbound filtering or security controls.

The cybercriminal attacker escalated his privileges to a SYSTEM-level access. Mimikatz was used to retrieve password credentials of users and the administrator from memory.

The HMI with connection to SWaT running on an unprotected VNC server was targeted. As there was no authentication, the cybercriminal attacker deployed a SOCKS proxy which allowed them to run a VNC client on their own system. This client then communicated with the VNC server over a HTTP beaconing C2 channel. Now, the cybercriminal attacker has direct control over the system through a compromised HMI. A back-door was established.

4.4.2.1 Control of the pump through the compromised HMI

Objective : Set the level indicator transmitter LIT301 to dangerously high level

Attack method : Back-door connection, attack through compromised HMI

Tools : Nil

Description : The intention of this attack was to control the pumps to fill up the water tank to a dangerous level without detection through the back-door connection that was established previously. As the cybercriminal attacker had access to the HMI through the HTTP beacon as illustrated in 4.4.2, he controlled the system as a regular operator. Hence, pump P101 was switched into manual mode operation and turned on (please refer to 4.2.1.3 attack model). Level indicator transmitter LIT301 showed the water level was at a dangerous level.

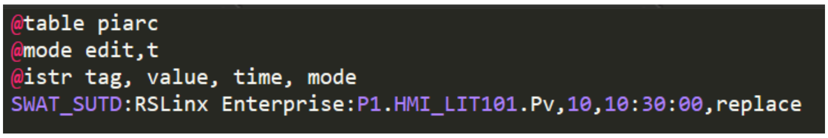
4.4.2.2 Overwriting data stored at Historian

Objective : Overwrote specific data stored at Historian

Attack method : Back-door connection, attack through compromised SCADA workstation

Tools : Microsoft PsExec, ipconfig

Description : The intention of this attack was to overwrite specific data stored at the Historian through the back-door connection established previously. Having obtained the user credentials, the cybercriminal attacker used PsExec to gain access to the Historian database server. PsExec is a light-weight telnet-replacement that lets you execute processes on other systems, complete with full interactivity for console applications, without having to manually install client software. Using Server Message Block (SMB), the cybercriminal attacker sent commands to the Historian through the compromised SCADA workstation. Within the Historian server, the attacker found a custom command-line utility, piconfig that allows reading and writing of data to the database. The piconfig utility is modal and interactive that allows user to maintain and configure real-time database management systems' (PI Server) databases and tables, such as the PI point database and the digital state table. Using piconfig, the cybercriminal attacker overwrote the original value of the raw water tank level tag (HMI_LIT101) for 10.30 AM with the value 10. In summary, the attacker used PsExec to gain access to the Historian and executed ipconfig remotely. Ipconfig was executed with the following commands.



```
@table piarc
@mode edit,t
@istr tag, value, time, mode
SWAT_SUTD:RSLinx Enterprise:P1.HMI_LIT101.Pv,10,10:30:00,replace
```

Figure 15: Commands executed by piconfig to overwrite data

4.4.2.3 Control of the PLC through modification of PLC logic in Studio 5000

Please refer to 4.3.2.2 for similar attack model.

4.4.3 Strong Attacker Model

4.4.3.1 Control of the HMI/SCADA through a bridged MiTM at Level 1

Objective : Change the display value of the HMI

Attack method : Back-door connection, MiTM

Tools : NetFilterQueue, Scapy

Description : The intention of this attack was to control the values displayed at the HMI. Using the bridged technique described in 4.1.2.1, the attacker placed a device between the PLC and the HMI. As such, the attacker overwrote the tag value transmitted from the PLC to the HMI.

4.5 Ox002147 (University of Oxford), United Kingdom

Ox002147 is a team from the University of Oxford that is interested in information security. They regularly participate in Capture-The-Flag (CTF) events and took first place in the Deloitte CTF Final in 2016.

4.5.1 Insider Attacker Model

4.5.1.1 Control of the motorised valve MV201 through the modification of PLC logic

Objective : Change the status of the motorised valve MV201

Attack method : Changing of PLC logic code

Tools : Studio 5000

Description : The intention of this attack was to alter the states of the valves and sustain the altered state for a minute. A PLCs' logic code defines the way it controls the physical process through the actuators based on the information it receives from the sensors. The attacker modified the existing PLC code using Studio 5000 to change the state of the valve MV201 in the pre-treatment P2 phase.

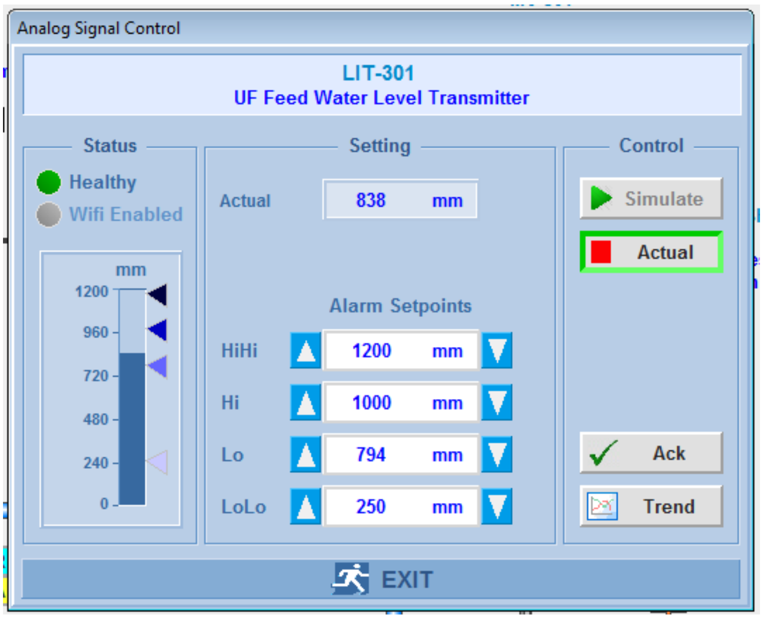
4.5.1.2 Control of the water tank level LIT301 through adjusting alarm levels

Objective : Decrease the water tank level without raising the alarm

Attack method : Manual manipulation

Tools : HMI

Description : The intention of this attack was to lower the water tank level from 820mm to 420mm without raising any alarm. The attacker changed the low-low alarm threshold of LIT301 through the HMI. Gradually, the water level identified by level indicator transmitter LIT301 decreased till 320mm; hence successfully bypassed the 420mm alarm threshold.



Analog Signal Control

LIT-301
UF Feed Water Level Transmitter

Status
● Healthy
● Wifi Enabled

mm
1200
960
720
480
240
0

Setting

Actual 838 mm

Alarm Setpoints

HiHi	1200 mm
Hi	1000 mm
Lo	794 mm
LoLo	250 mm

Control
Simulate
Actual
Ack
Trend

EXIT

Figure 16: LIT301 display at HMI

4.5.1.3 Control of the water pump P101 through manual operation of the HMI

Please refer to 4.2.1.3 for similar attack model.

4.5.1.4 Control of the chemical dosing pump P205 through manual operation of the dosing meter

Objective : Change the chemical dosage of sodium hypochlorite (NaOCl) in P2

Attack method : Manual manipulation

Tools : Nil

Description : The intention of this attack was to change the chemical dosage of the water during the pre-treatment P2 phase. The dosing pump meter P205 that controlled the amount of NaOCl for water disinfection was targeted. The insider attacker manually turned the adjustment knob to increase the concentration of the NaOCl in the pre-treated water. Solutions containing more than 40% sodium hypochlorite by weight are considered hazardous oxidizers.

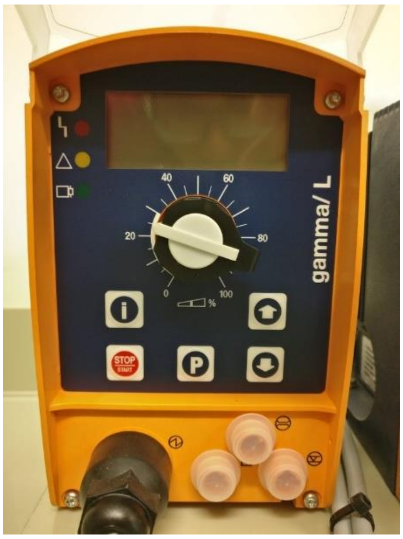


Figure 17: Metering Pump of P205

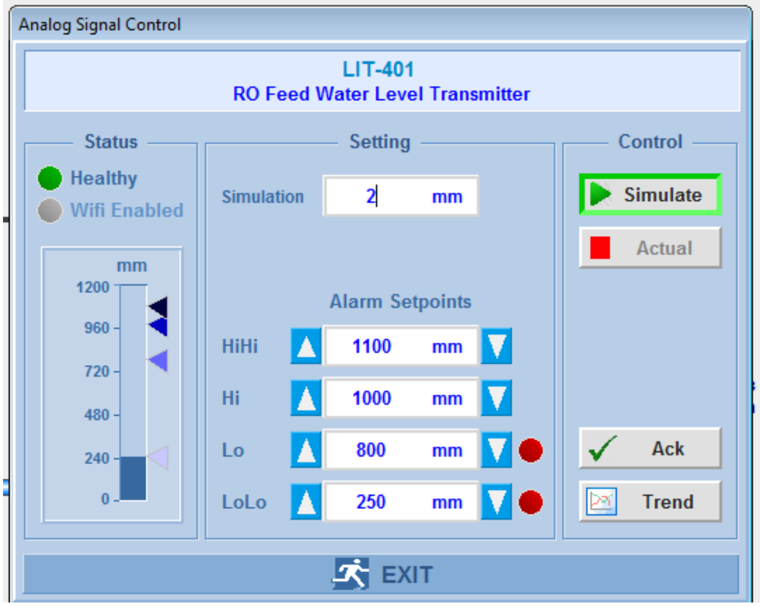
4.5.1.5 Control of the HMI/SCADA through simulation control

Objective : Change the display value of level indicator transmitter LIT401 in the HMI

Attack method : Manual manipulation

Tools : Nil

Description : The intent of this attack was to control the HMI thereby modifying the display value of the water level identified by LIT401. This was achieved through activating the simulation mode in the HMI. This allowed the insider attacker to change the numerical value that was displayed on the HMI.



Analog Signal Control

LIT-401
RO Feed Water Level Transmitter

Simulation: 2 mm

Alarm Setpoints

HiHi	1100	mm
Hi	1000	mm
Lo	800	mm
LoLo	250	mm

Figure 18: HMI display of LIT401

4.5.1.6 Control of the PLC through disconnected network cables

Objective : Disrupt sensor values from remote input/output (RIO) to the PLC

Attack method : Manual Manipulation

Tools : Nil

Description : The intention of this attack was to disrupt the flow of sensor values from RIO to the PLC. The insider attacker disconnected the cables between the RIO and PLC. On the HMI display, the targeted PLC was highlighted to be down.

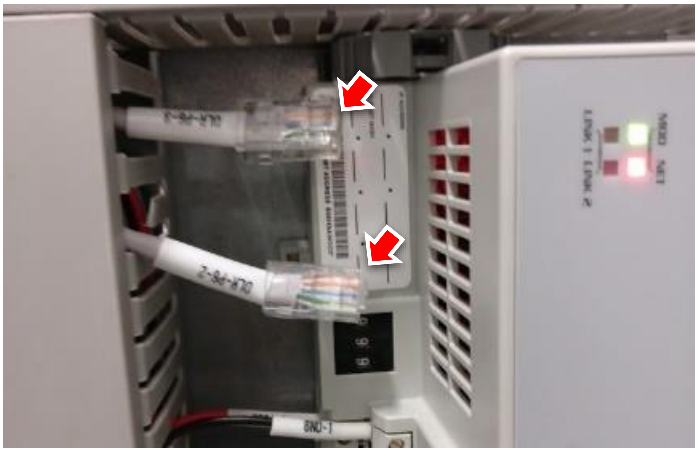


Figure 19: Disconnected network cable at RIO

4.5.1.7 Control of the RIO through disconnected Analogue Input/Output pin

Please refer to 4.3.1.1 for similar attack model.

4.5.2 Cybercriminal Attacker Model

4.5.2.1 Control of the Historian through MiTM using ARP

Objective : Change the value stored at Historian

Attack method : MiTM with ARP

Tools : Ettercap

Description : The intention of this attack was to change the values stored in the Historian. Using similar ARP poisoning technique described in 4.2.2.1, the cybercriminal attacker changed the values from the PLC to the Historian. Hence, the value stored in the Historian was adulterated.

5 Defence Teams

5.1 WaterDefense from iTrust, Singapore

5.1.1 Background

WaterDefense is an attack detection system developed in-house by a team of researchers in iTrust. Unlike the commercial products of Company_A and Company_B, WaterDefense is a product in development with its patent filed. Through the course of its development, WaterDefense was iteratively improved through extensive experimentation in SWaT.

5.1.2 Technology Description

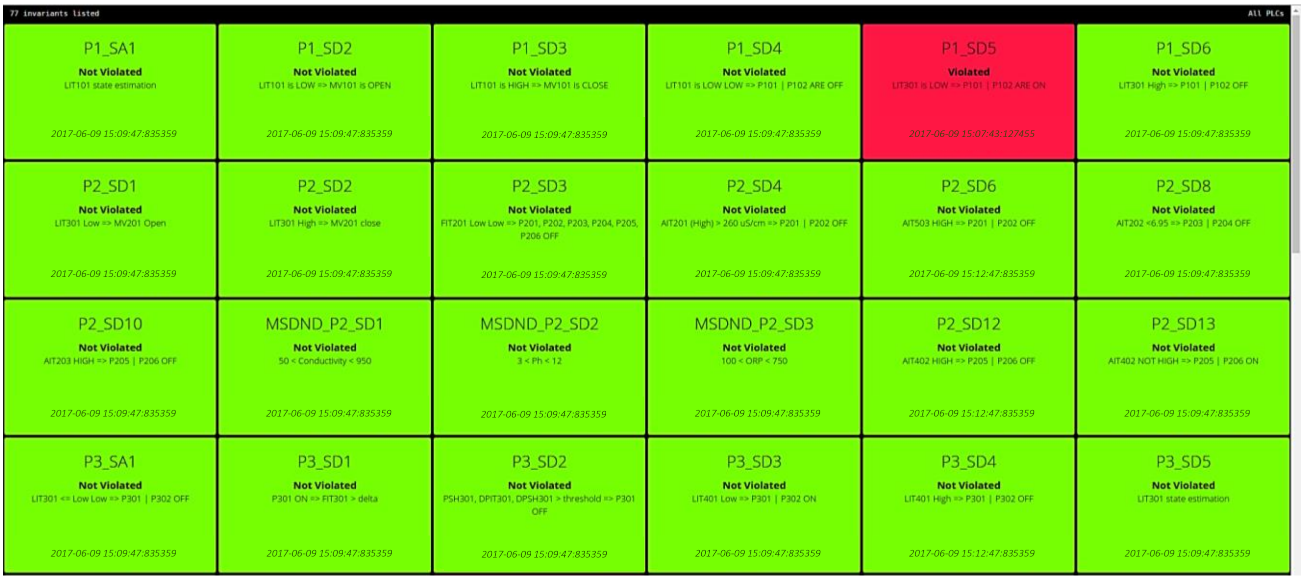
WaterDefense can be considered as a host-based intrusion detection system (HIDS). Specifically, it collects data on the various sensor measurements of physical processes such as water pH value, water level and flow indicator of the CPS, for analysis and process anomaly detection. By using all 52 sensor values of SWaT, it can detect single-stage multipoint and multi-stage multi-point cyber-attacks (Adepu, Mathur, 2016) in a distributed control system.

WaterDefense is a novel technology because it is a reliable detection mechanism using "security by design" for many basic and advanced attacker models. Based on the rules of physics, it directly verifies the process variables of the CPS within the distributed Programme Logic Controllers (PLCs) to check for abnormal behaviour.

Process variables are time-dependent and interrelated within the entire plant process. Hence, their values are constrained by the relationship they have with the other process variables, as governed by the fundamental laws of physics and/or chemistry. The relationships among these constrained variables lead to process invariants - WaterDefense' rule-based algorithms.

The invariants are embedded in PLCs as well as special hardware devices known as intelligent checkers (ICs) with wired interfaces to sensors and actuators. The invariants are checked constantly to ensure the underlying processes are behaving as intended. When an invariant is violated, the underlying CPS process has diverged from its intended behaviour and an alarm is triggered.

Figure 20 below shows an instance of the WaterDefense' interface for which an alarm for the invariant P1_SD5 was triggered as it was detected as being violated. The physical rule that was violated was part of the encoded control logic in SWaT, whereby the Raw Water pumps P101 and P102 should be turned on when the Ultrafiltration Feed Water Tank Level (LIT301) downstream was low.



77 invariants listed

All PLCs

P1_SA1 **Not Violated** LIT101 state estimation *2017-06-09 15:09:47:835359*	P1_SD2 **Not Violated** LIT101 is LOW => MV101 is OPEN *2017-06-09 15:09:47:835359*	P1_SD3 **Not Violated** LIT101 is HIGH => MV101 is CLOSE *2017-06-09 15:09:47:835359*	P1_SD4 **Not Violated** LIT101 is LOW LOW => P101 | P102 ARE OFF *2017-06-09 15:09:47:835359*	P1_SD5 **Violated** LIT301 is LOW => P101 | P102 ARE ON *2017-06-09 15:07:43:127455*	P1_SD6 **Not Violated** LIT301 High => P101 | P102 OFF *2017-06-09 15:09:47:835359*
P2_SD1 **Not Violated** LIT301 Low => MV201 Open *2017-06-09 15:09:47:835359*	P2_SD2 **Not Violated** LIT301 High => MV201 close *2017-06-09 15:09:47:835359*	P2_SD3 **Not Violated** FIT201 Low Low => P201, P202, P203, P204, P205, P206 OFF *2017-06-09 15:09:47:835359*	P2_SD4 **Not Violated** AIT201 (High) > 260 uS/cm => P201 | P202 OFF *2017-06-09 15:09:47:835359*	P2_SD6 **Not Violated** AIT503 HIGH => P201 | P202 OFF *2017-06-09 15:12:47:835359*	P2_SD8 **Not Violated** AIT202 <6.95 => P203 | P204 OFF *2017-06-09 15:09:47:835359*
P2_SD10 **Not Violated** AIT203 HIGH => P205 | P206 OFF *2017-06-09 15:09:47:835359*	MSDND_P2_SD1 **Not Violated** 50 < Conductivity < 950 *2017-06-09 15:09:47:835359*	MSDND_P2_SD2 **Not Violated** 3 < Ph < 12 *2017-06-09 15:09:47:835359*	MSDND_P2_SD3 **Not Violated** 100 < ORP < 750 *2017-06-09 15:09:47:835359*	P2_SD12 **Not Violated** AIT402 HIGH => P205 | P206 OFF *2017-06-09 15:12:47:835359*	P2_SD13 **Not Violated** AIT402 NOT HIGH => P205 | P206 ON *2017-06-09 15:09:47:835359*
P3_SA1 **Not Violated** LIT301 <= Low Low => P301 | P302 OFF *2017-06-09 15:09:47:835359*	P3_SD1 **Not Violated** P301 ON => FIT301 > delta *2017-06-09 15:09:47:835359*	P3_SD2 **Not Violated** PSH301, DPIT301, DPSH301 > threshold => P301 OFF *2017-06-09 15:09:47:835359*	P3_SD3 **Not Violated** LIT401 Low => P301 | P302 ON *2017-06-09 15:09:47:835359*	P3_SD4 **Not Violated** LIT401 High => P301 | P302 OFF *2017-06-09 15:12:47:835359*	P3_SD5 **Not Violated** LIT301 state estimation *2017-06-09 15:09:47:835359*

Figure 20: WaterDefense Alarm Screenshot

5.2 WaterDefense_Historian from iTrust, Singapore

5.2.1 Background

WaterDefense_Historian was created by an expanded team that includes members of the team that developed WaterDefense. It is a variant of WaterDefense. Essentially, the it has same detection capability but relies on data from the Historian instead of from the PLC. In addition, WaterDefense_Historian can be deployed on a separate server for an added defence layer towards orthogonal defence.

5.2.2 Technology Description

WaterDefense_Historian leverages on the same principle of process invariants and uses the same algorithms employed in WaterDefense. The difference is that WaterDefense_Historian takes in data from Historian of a CPS instead. In this manner, WaterDefense_Historian is still able to detect process abnormalities when the PLC has been compromised by hackers. WaterDefense_Historian is useful in operational legacy systems since legacy systems might not be able to support the deployment requirements for WaterDefense.

5.3 Product_A from Company_A

5.3.1 Background

TEXT REMOVED INTENTIONALLY.

5.3.2 Technology Description

TEXT REMOVED INTENTIONALLY.

TEXT REMOVED INTENTIONALLY.

Table 1: Company_A IDS Functions

TEXT REMOVED INTENTIONALLY.

Figure 21: Product_A Alarm Screenshot

5.4 Product_B from Company_B

5.4.1 Background

TEXT REMOVED INTENTIONALLY.

5.4.2 Technology Description

TEXT REMOVED INTENTIONALLY.

6 Evaluation of Defence Mechanisms

WaterDefense, WaterDefense_Historian, Product_A and Product_B were each evaluated based on (a) detection rate, (b) breadth of defence and (c) forensics analysis capability, as presented in this section.

6.1 Detection Rate

The detection rate of each IDS was computed as a metric as the setup of S^3-17 did not allow for an evaluation by the more comprehensive metric of detection precision. Detection precision would have considered false positives and evaluated the accuracy of each IDS precisely. However, the calculation of number of false positives during S^3-17 was highly inaccurate because of the multiple concurrent and/or sequential attacks launched by the attacker teams during their 2-hour timeslot. An alarm raised that was logically attributed to a recent attack could be either correct, a false positive, or a result of a cascading effect from an earlier attack. Therefore, the detection rate of the IDS was the next best alternative considered by the team.

The team cross-referenced the detection logs of each IDS with the time-stamped attacks during S^3-17 to gauge the detection rates. The criteria - If an alarm *attributable to the attack* is raised *within 5 minutes* of the attack launched, the attack was considered detected. Hence, only immediate and apparent detections were credited. Such examples of detections are described in the following subsections 6.1.1 - 6.1.4.

The IDS detected nine different categories of attacks during S^3-17, that of five are Physical Process Goals and four are Sensor Data Goals, as described in Section 3.1. Table 3 below tabulates percentages of attacks detected by each IDS for each attack category. Figure 22 is an accompanying bar chart.

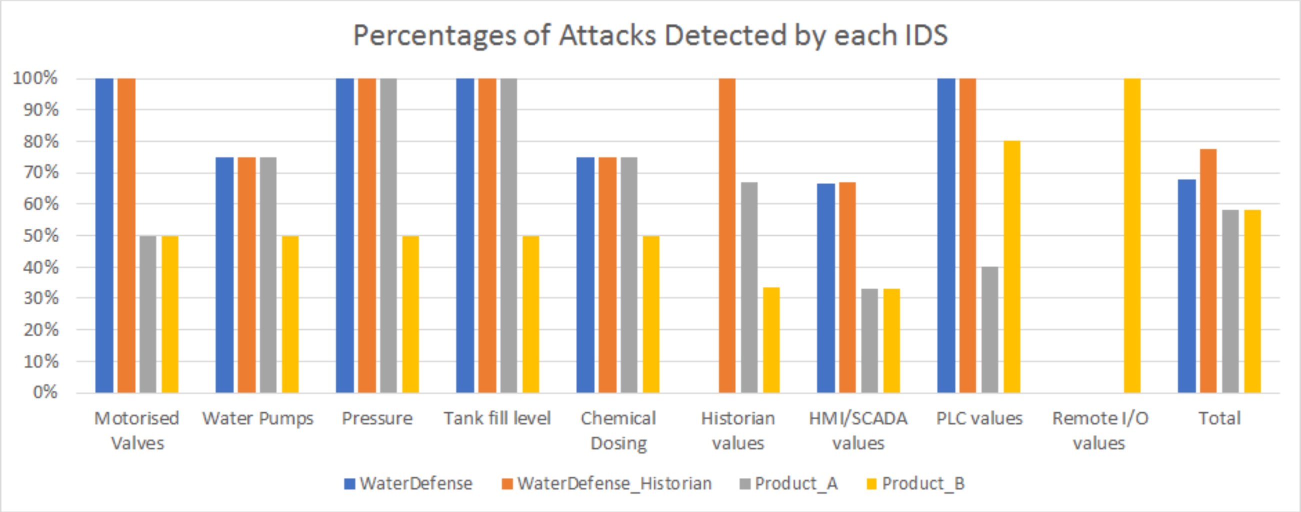


Figure 22: Percentages of Attacks Detected by each IDS

Category of Attack	No. of Successful Attacks Recorded for all Teams Over Two Days of S^3-17	Percentage of Attacks Detected			
		WaterDefense	WaterDefense_ Historian	Product_A	Product_B
Physical Process Goals					
a) Motorised Valves	2	100%	100%	50%	50%
b) Water Pumps	4	75%	75%	75%	50%
c) Pressure	2	100%	100%	100%	50%
d) Water Tank Level	4	100%	100%	100%	50%
e) Chemical Dosing	4	75%	75%	75%	50%
Sensor Data Goals					
a) Historian values	3	0%	100%	67%	33%
b) HMI/SCADA values	3	67%	67%	33%	33%
c) PLC values	5	100%	100%	40%	80%
d) Remote I/O values	4	0%	0%	0%	100%
Overall Percentage	**31**	**68%**	**77%**	**58%**	**58%**

Table 2: Percentages of Attacks Detected by each IDS

6.1.1 WaterDefense

6.1.1.1 Detection on Physical Process Attacks

WaterDefense performed as one of the best in detecting attacks on the physical process of SWaT. All attacks on the Motorised Valves, Pressure and Water Tank Level were detected. Majority of the attacks on the Chemical Dosing and Water Pumps of SWaT were also detected.

A detection example was when the attacker team LosFuzzys took control of the pressure pump through a Python script (pycomm) to raise the pressure at DPIT301 to a dangerous level. WaterDefense raised the alarm of the invariant P3_SD3. This invariant rule was derived from PLC3's command rule that pump P301 is required to turn off when the pressure at DPIT301 was above a certain threshold. During the attack, the rule was violated because the pump was not turned off while the DPIT301 was above the allowed threshold. As a result, the alarm P3_SD3 was raised immediately.

In certain cases, multiple invariant alarms were raised in a single physical process compromise. For example, when the tank level LIT101 was compromised separately by attack teams GHA and Lancaster, three invariants (P1_SA1, P1_SD2 and P1_SD3) related to the state and threshold values of variable LIT101 were violated and three alarms were raised.

Hence, the process integrity of certain variables, such as LIT101, were well encoded within WaterDefense' invariant rules. Attacks upsetting the integrity of these variables are well-protected by WaterDefense, in comparison to the protection of the chemical dosing or water pump variables, for which some attacks on them were not detected.

6.1.1.2 Detection on Sensor Data Attacks

WaterDefense detected attacks on HMI/SCADA and PLC values because these attacks directly compromised the physical processes. These attacks either compromised the Chemical Dosing, Water Tank levels or Pump status through hacking of the HMI/SCADA or PLC. Hence, the robustness of WaterDefense in detecting unusual physical processes was effective in these attacks.

On the other hand, WaterDefense was unable to detect insider attacks of pulling out Remote I/O cables. This is because WaterDefense will trigger the alarm when the rules of physics and chemistry of the plant are violated. In usual case, for a period of time, PLC continues to run based on the last known state and/or values.

6.1.2 WaterDefense_Historian

6.1.2.1 Detection on Physical Process Attacks

Having the same algometric core as WaterDefense, WaterDefense_Historian yielded the same detection rate on physical process attacks as its kin. Responses to the physical process attacks were almost identical. The attacks detected were exactly the same, although the evidence may differ.

WaterDefense_Historian had fewer alarms triggered: It triggered 49 while WaterDefense triggered 70 alarms during the attacks launched by LosFuzzys and H4x0rPsch0rr. Hence, WaterDefense_Historian is less sensitive to the physical process deviations as compared to WaterDefense. This could be due to the design of WaterDefense_Historian where its invariants are evaluated from data in the Historian instead of live data available to the PLCs. Despite so, both WaterDefense and WaterDefense_Historian performed equally well in detecting attacks on the physical process of SWaT.

6.1.2.2 Detection on Sensor Data Attacks

Like WaterDefense, WaterDefense_Historian detected the attacks on HMI/SCADA and PLC values because these attacks directly compromised the physical processes. WaterDefense_Historian also did not detect any attack launched against the Remote I/O by pulling the cables.

Albeit strong similarities with WaterDefense, WaterDefense_Historian could strikingly detect all attacks against the Historian when WaterDefense did not detect any at all. This is because WaterDefense_Historian was accessing data on the server directly. When attacker team GHA compromised the Historian values by changing the water tank level LIT101 values (Section 4.1.2.3), WaterDefense_Historian triggered two invariant alarms (P1_SA1, P1_SD3) related to the LIT101 variable very quickly.

6.1.3 Product_A

6.1.3.1 Detection of Physical Process Attacks

Product_A detects attacks by monitoring the network traffic for intrusions. For this reason, the physical process attacks were detected based on network intrusions.

For example, Product_A triggered a 'critical' alert for the attack on motorised valve MV201 manipulated by attacker team Ox002147 when they used Studio 5000 to reprogram PLC2's control logic (Section 4.5.1.1). Product_A detected a compromised PLC2 because of the established communication with PLC2 by an unauthorised machine of an unidentified MAC and IP addresses. In addition, alarm had been triggered as PLC2 had performed unauthorised functions such as memory clearing and rewriting, and restarting. This 'critical' event had triggered the alarm immediately.

Furthermore, Product_A triggered a sequence of 'important' alerts for physical process attacked by Pycomm Python script. For example, when H4x0rPsch0rr compromised the chemical dosing pump of P205 (Section 4.3.2.1), Product_A detected separate connections to multiple PLCs by a foreign IP address, then finally a communication to PLC2 in this MiTM attack.

By detecting unauthorised network intrusions in the above-mentioned way, Product_A was also able to detect many of physical process attacks. These instances include attackers exploited the VNC vulnerability connecting to the HMI (Section 4.1.2.2), or established a back-door access (Section 4.4.2) and connected to the Telnet port (Section 4.1.2.6).

However, Product_A was unable to detect the manual physical process attacks as there was no network intrusion involved. These instances were insider attacks that had successfully compromised the physical process of the plant. An example was the control of the Motorised Valve by GHA in Section 4.1.1.1.

6.1.3.2 Detection of Sensor Data Attacks

Product_A did not detect some sensor data attacks as seen in Table 3. All the Remote I/O attacks (pulling of cables) were achieved manually as an insider attacks, which rendered Product_A insufficient to access the hacking activities. For example, insider attacks such as HMI and PLC manipulation as described in Section 4.5.1.5 and 4.5.1.6 respectively.

As for the network intrusions, Product_A was able to detect most but not all of these attacks. Examples were the MiTM attack launched by GHA compromising the PLC (Section 4.1.2.1) and sending modified packets to the Historian (Section 4.1.2.3). The attack launched by the attacker team Ox002147 using ARP spoofing was also detected because the attack involved the Historian broadcasting, which was an unusual behaviour and hence an alarm was triggered. Contrastingly, when attacker teams Lancaster and H4x0rPsch0rr modified the PLC logic using Studio 5000, Product_A was unable to detect such attacks. Similarly, Product_A did not trigger any alarm for the back-door access (Section 4.4.2) attack launched by Lancaster. In this attack, commands were valid (malicious intent) looked unsuspicious in the usual network traffic.

6.1.4 Product_B

6.1.4.1 *Detection of Physical Process Attacks*

Product_B had a mixed performance on the detection on physical process attacks. Table 2 shows that Product_B detected half of the total attacks of each attack category.

For which Product_B detected the attacks, it was mostly the detector modules which monitors the unexpected physical process behaviour that triggered alarms. For example, when the attacker team Ox002147 manipulated the motorised valve MV201 and the chemical dosing pump P205 on two occasions (Section 4.5.1.1 and 4.5.1.4), the Material balance detector (for LIT301's process valve) and Range detector (for AIT202's process valve) triggered alarms respectively.

However, it is interesting to note that the break of material balance equations for water tank level LIT301 (by the Material balance detector) triggered a 'High' severity alarm, while having chemical process values out of normal range (by the Range detector) triggered a 'Low' severity alarm for the chemical dosing attack.

In some detections, Product_B detected both physical process being compromised and network intrusion. An example was the chemical dosing attack launched by the attacker team LosFuzzys (Section 4.2.2.3). There were alarms triggered by both the difference detector on chemical dosing meter, and the groups communication detector for which PLC2 was recognised to have less tags in data than expected, due to the hack.

6.1.4.2 *Detection of Sensor Data Attacks*

Product_B also had a mixed performance on the detection on sensor data attacks.

While some of the attacks were detected by network detectors, others were detected by the physical process detectors due to the process being compromised during the attack. The attack on PLC3 by the attacker team H4x0rPsch0rr (Section 4.3.2.2) involved changing the water tank level LIT301. In this instance, both the group communication detector, which detected an increase in number of tags updated in tags group of PLC3, and range detector, which detected values out of normal range in five tags for the water tank level triggered alarms.

Product_B detected most of the Remote I/O attacks (pulling cables out). In some instances, it detected unusual update rate in tags/data received by the PLC, as such in the attack by attacker team Lancaster (Section 4.4.1.1). In another instance, it detected an abnormal difference in nine tags for the AIT203 chemical dosing meter, for which team LozFuzzys disconnected the Remote I/O from the PLC. On the other hand, Product_B was unable to detect attacks on the Historian values.

6.2 Breadth of Defence

The four IDS supplied varied in terms of their detection coverage. The breadth of the defence is wide when the IDS can detect intrusions across a varied attack surface.

WaterDefense and WaterDefense_Historian might not be able to detect network intrusions, but by monitoring the integrity of the physical process, which is usually compromised as the goal of an attacker, they are able to detect attacks regardless of the attack method.

Product_A was able to detect most of the network intrusions, but unable to detect insider attacks, such as some attacks launched by the attacker team Ox002147 (Section 4.5.1).

Product_B had the widest breadth of defence as a hybrid of NIDS and HIDS. Its multi-pronged detection approach with 11 detector modules ensured that it maintained a certain level of detection throughout the attack surface. In comparison to WaterDefense or WaterDefense_Historian, it could detect attacks on the remote I/O. In contrast to Product_A, it detected insider attacks that compromised the physical process. The only attack surface it did not cover is attacks on Historian, for which it did not monitor the network traffic, nor was connected to its physical process variable data.

6.3 Forensics Analysis Capability

Post-attack detection, it is important for the operator of CPS to recover its process functionality quickly. The IDS were assessed based on the level of detail provided about each intrusion that enables which correlates to the survivability of the CPS in an event of a Cyber-Physical attack.

To begin with, at the time of alarm, Product_A was able to provide a clear-cut distinction between an actual intrusion versus a false alarm. It detects unauthorised access in the network layer, and provides details on the intruder's IP or MAC addresses.

In contrast, WaterDefense, WaterDefense_Historian and Product_B are unable to distinguish between operational fault (e.g., failure of physical components) and an actual cyber-attack for a flagged physical process.

With regards to assisting system recovery, Product_B, WaterDefense and WaterDefense_Historian can directly pinpoint the part of the physical process that has been compromised, while Product_A might not. Yet, information on network intrusions provided by Product_A could potentially support system isolation to block out the attacker's access.

To support attack diagnosis, important information is the amount of physical process deviation from normal when the CPS is under attack. However, all four IDS used during S^3-17 seem to lack the presentation of the information, although WaterDefense, WaterDefense_Historian and Product_B should be able to compute these information from the sensor values and expected behaviour. The information would be good forensics material to study the extent of control the attacker has on the CPS.

7 Final Remarks for Defence Systems

A variety of IDS were demonstrated during S^3-17. Most significantly, the IDS performance varied across different products. While it is important to improve detection accuracy, the importance of having a hybrid of NIDS and HIDS is apparent. Product_B has a broader defence surface, which is more ideal. Having a hybrid of NIDS and HIDS also assists in system recovery and forensics analysis. It is therefore recommended to deploy a variety of IDS for orthogonal and effective defence.