

Confidentiality Notice: This document is confidential and contains proprietary information and intellectual property of Trustwave, a Singtel company, its parent company and its subsidiaries. Neither this document nor any of the information contained herein may be reproduced or disclosed under any circumstances. Please be aware that disclosure, copying, distribution or use of this document and the information contained therein is strictly prohibited.

Power Generator and Substation

VERSION: CORPLAB-2021-T5F-A1

30/03/2021

Prepared by

Liu Yuancheng

Senior Security Development Engineer

[yuancheng.liu@trustwave.com](https://nusu-my.sharepoint.com/personal/dcsshch_nus_edu_sg/Documents/Work%20Related%20Files/Singtel%20OT%20Training/yuancheng.liu@trustwave.com)

Wong Jun Wen

Asst R&D Manager

[junwen.wong@trustwave.com](https://nusu-my.sharepoint.com/personal/dcsshch_nus_edu_sg/Documents/Work%20Related%20Files/Singtel%20OT%20Training/junwen.wong@trustwave.com)

Dr. Shantanu Chakrabarty

NUS Research Fellow

[shantanu1088@gmail.com](https://nusu-my.sharepoint.com/personal/dcsshch_nus_edu_sg/Documents/Work%20Related%20Files/Singtel%20OT%20Training/shantanu1088@gmail.com)

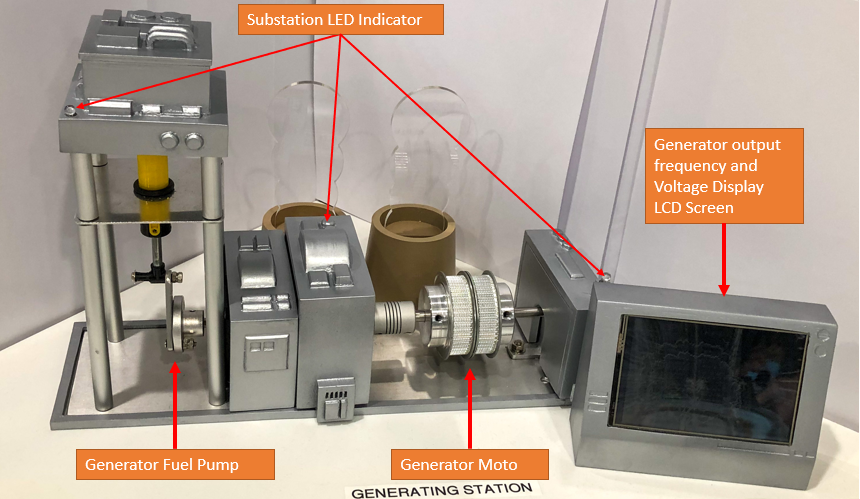
CSI OT 3D Platform Cyber Attack Demonstration

**CSI OT 3D Platform Cyber Attack Demonstration Power Generator and Substation Design Manual**

**1. Project Introduction**

The OT 3D Platform contents a power generator module which is used to simulate the power generation for the module’s inner and outer railway, train substation and system’s power supply. The power generator is directly controlled by an Arduino with voltage signal(H/L) change, we use the USB cable connect the Raspberry PI to the Arduino to control the LEDs, LCD screen, smoke indicator frequency and siren with serial commands. The Arduino also has 4 pins (A0, A1, A4, A5) to control the speed of moto and pump and the 4 pins are connected to the PLC with jumper wire. For the remote-control software part, we have implemented 2 programs: Power generator auto-control program (main controller) and Remote generator controller (HMI for user).

* 1. **View of the power generator hardware**



1.1.1 Generator Info LCD Display: Show current generator output frequency and voltage value. It also has two indicators bar to show the loads. (Green for normal, yellow for no no-load, red for overload)

1.1.2 Station Indicator LEDs [green/amber/red]: Frequency LED, Voltage LED, Moto LED and Pump LED.

1.1.3 Generator Fuel Pump: A pump with a driving moto and single cylinder moving up and down to show the fuel transfer speed to the generator. (speed selection: stop, slow and fast)

1.1.4 Generator Moto: A rotate wheel with a driving moto to show the moto speed/power output of the generator. (speed selection: stop, slow and fast)

1.1.4 Smoke indicator: Two smoke shape transparency panels with flashing blue LED at the bottom to show the speed generation speed. (slow/fast)

1.1.5 Siren: A built in speaker to play siren sound to show the alter to the user when the generator got exception situation.

**1.2 Power generator auto-control program**

The power generator auto-control program will be running in the Raspberry PI to control the OT-Power Generator Module's hardware. It will send Modbus TCP command to 3 PLCs and serial command to the Arduino, then receive the control request from the remove HMI controller. The control program will do the adjustment of the generator's motor and pump speed based on the loads in the system. The program contents 4 thread to finish the different tasks parallel:

1.2.1 Main thread: In the main thread, we will read the 3 PLCs load, do the generator frequency and voltage adjustment, handle the PLC reconnection if any of the PLC was disconnected.

1.2.2 UDP Communication thread: A thread with UDP server to handle the control request from the generator control HMI.

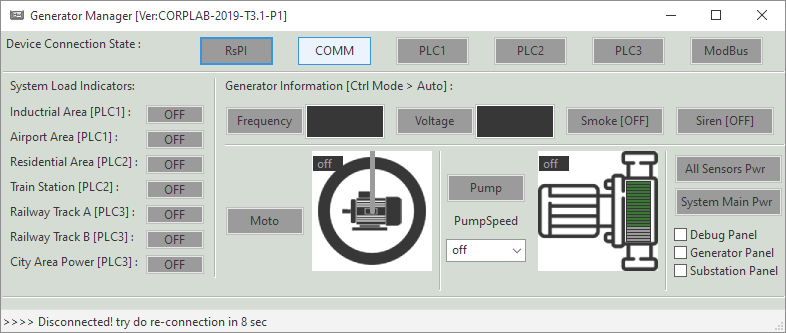
1.2.3 Modbus TCP communication thread: A thread with TCP server to response the parameters Modbus message fetching request from the HMI.

1.2.4 Attack handling thread: The thread will active and simulate the attack scenario.

**1.3 Remote generator controller HMI**

This module will provide a HMI user interface to connect to the power generator auto control program by UDP and display the generator states. The user interface contents 4 part: main control dashboard, debug panel, generator display panel and Substation panel.

1.3.1 View of the main control dashboard



The Device connection state area will show the connection between all the components in the system: - RsPI: UDP connection between HMI program and the Raspberry PI controller.

- COMM: USB serial connection between Raspberry PI and the Arduino.

- PLC 1: Modbus TCP connection between Raspberry PI and M221 PLC 1

- PLC 2: Modbus TCP connection between Raspberry PI and S7-1200 PLC 2

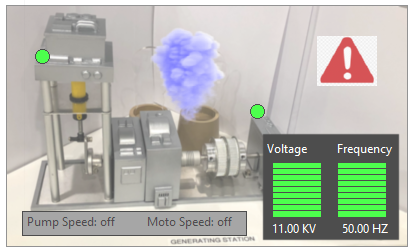
- PLC 3: Modbus TCP connection between Raspberry PI and M221 PLC 3

- Modbus: Modbus TCP connection between HMI program and the Raspberry PI controller.

The Load Area will show the power supply on/off state of the seven main components in the 3D Platform: Industrial area background light, Airport runway light, Residential area back ground light, Train station power supply and train stop control, Inner railway (track A) power supply, outer railway (track B) power supply and City area background power supply.

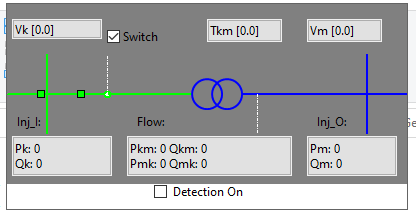
Generator information display area will show the information show on the generator module and the currently generator control mode is shown next to the title line. The right side extend function control area all the checkboxes are used to active the extend display panel shown on the screen. The System main power button is an emergency cut off button which used to turn off all the power of the generator module. The all-sensors power is the power of all the train position power around the 2 railways.

1.3.2 View of the generator display panel



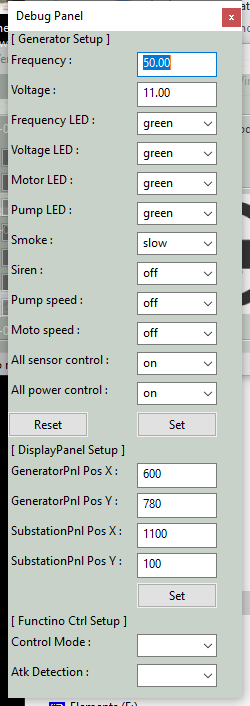
This panel will show the currently situation of all the components (introduced in the section 1.1 hardware introduction) of the Generator, when the smoke is set to “slow”, the smoke icon shown in the panel will be changed to gray color. Blue color icon means setting is “fast”. When the alter raised, the top right “Alert” icon will appear and flash. Check the “Generator Pnl” checkbox in the main UI, this panel will pop up on the bottom of the screen and layout above all the windows.

1.3.3 View of the substation display panel



In this panel we will show 10 parameters measured based on the substation status. To detect the stealthy substation attack based on the parameters, press the “Detection on” check box. The “Switch” check box is used to cut off all the power transmission of the power substation, the power supply from the substation to the platform will be lose. Check the “Substation Pnl” checkbox in the main UI, this panel will pop up on the top right of the screen and layout above all the windows.

1.3.3 View of the debug panel:



------------------------------------------------------------------------

Generator module setting:

'Freq': '50.00', # frequency (dd.dd)

'Volt': '11.00', # voltage (dd.dd)

'Fled': 'green', # frequencyled (green/amber/off)

'Vled': 'green', # voltage led (green/amber/off)

'Mled': 'green', # motor led (green/amber/off)

'Pled': 'green', # pump led (green/amber/off)

'Smok': 'off', # smoke indicator (fast/slow/off)

'Pspd': 'off', # pump speed (high/low/off)

'Mspd': 'off', # moto speed (high/low/off)

'Sirn': 'off', # siren (on/off)

'Spwr': 'off', # sensor power (on/off)

'Mpwr': 'on', # main power (on/off)

'Mode': 0 # control mode.

'Reset': button # reset all the value above to default.

'Set': Button # send the current setting to controller.

---------------------------------------------------------------------------

HMI Subpanel display position setting:

‘GeneratorPnl Pos X’:600 # Horizontal position of generator panel in the screen (pixel to top left corner).

‘GeneratorPnl Pos Y’:780 # Vertical position of generator panel in the screen (pixel to top left corner).

‘SubstationPnl Pos X’:1100 # Horizontal position of substation panel in the screen (pixel to top left corner).

‘SubstationPnl Pos Y’:100 # Vertical position of substation panel in the screen (pixel to top left corner).

---------------------------------------------------------------------------

Special function control selection setting:

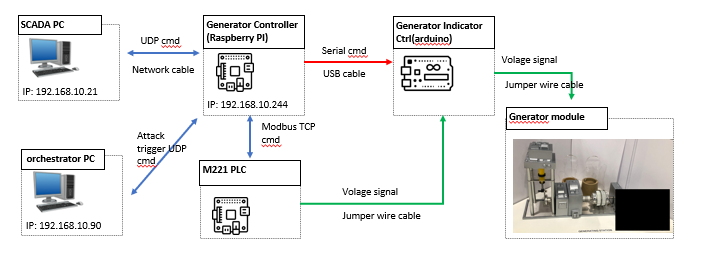
‘Control Mode’: ‘manual’ # Generator 'Freq', 'Volt', 'Fled', ‘'Smok'’ and 'Sirn' auto adjustment control.(auto/manual)

‘Atk Detection: ’off’ # Stealthy substation attack detection (on/off)

The debug panel is used to override the currently setting of the system. It is used for testing different function and has higher control priority than the other panels. Check the “Debug Pnl” checkbox in the main UI, this panel will pop up on the right side of the main window.

**1.4 System data communication**

1.4.1 Component’s communication diagram:



1.4.2 Communication message format:

UDP control and response: **msg = {'Cmd': str(\*\*\*),'Parm': {}}**

TCP Modbus message: **msg = str('000040010C'+ 'ff00'+…'ff10')**

USB serial command: **msg = str(Freq:Volt:Fled:Vled:Mled:Pled:Smok:Sirn)**

**2. Program Setup and Configuration**

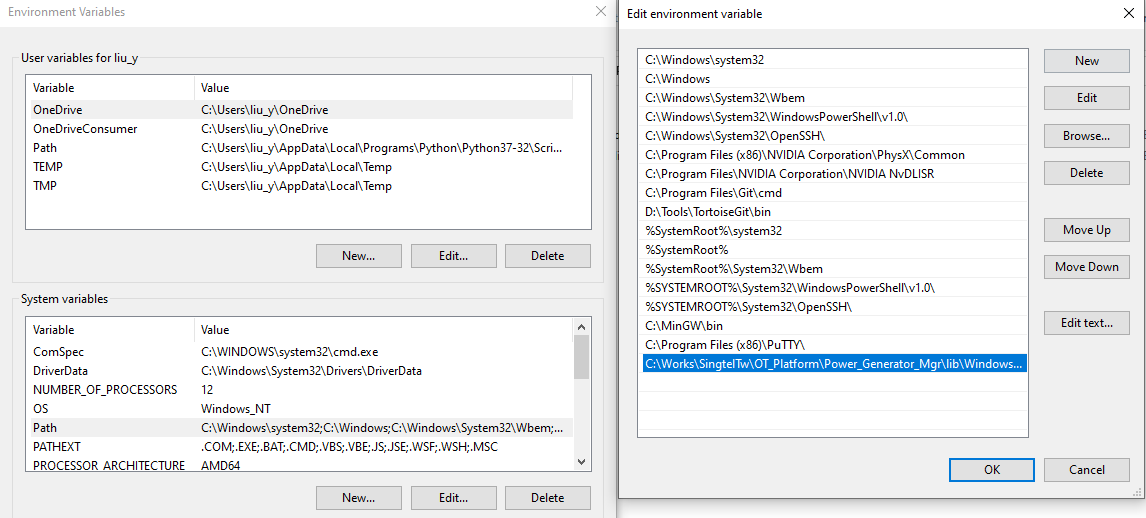
**2.1 Execution Setup**

2.1.1 Install Python 3.7.4 on Raspberry PI and the Windows machine.

2.1.2 Install snap7 + python-snap7 (need to install for S71200 PLC control) on Raspberry PI (Linux system): <http://simplyautomationized.blogspot.com/2014/12/raspberry-pi-getting-data-from-s7-1200.html>

2.1.3 Setup snap7 Used dll and lib file system environment path on windows system:

Computer > System Property > Advanced system settings > Environment Variables > System variable:



2.1.4 Install wxPython on windows machine: <https://www.wxpython.org/>

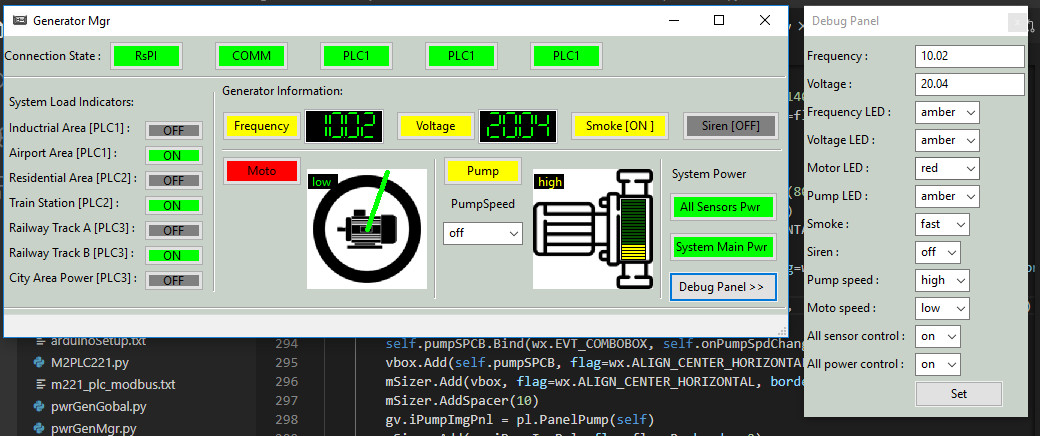
**2.2 Program Execution**

2.2.1 Set the test mode flag TEST\_MD to False of the pwrGenMgr.py and pwrGenRun.py, set the IP address in pwrGenRun.py to the Raspberry PI's IP.

2.2.2 Run the pwrGenMgr.py in the Raspberry PI: **python3 pwrGenMgr.py**

2.2.3 Run the HMI in Windows machine: double click the run.bat or run cmd **python3 pwrGenRun.py**

The UI will show as below:



**2.3 Program File List**

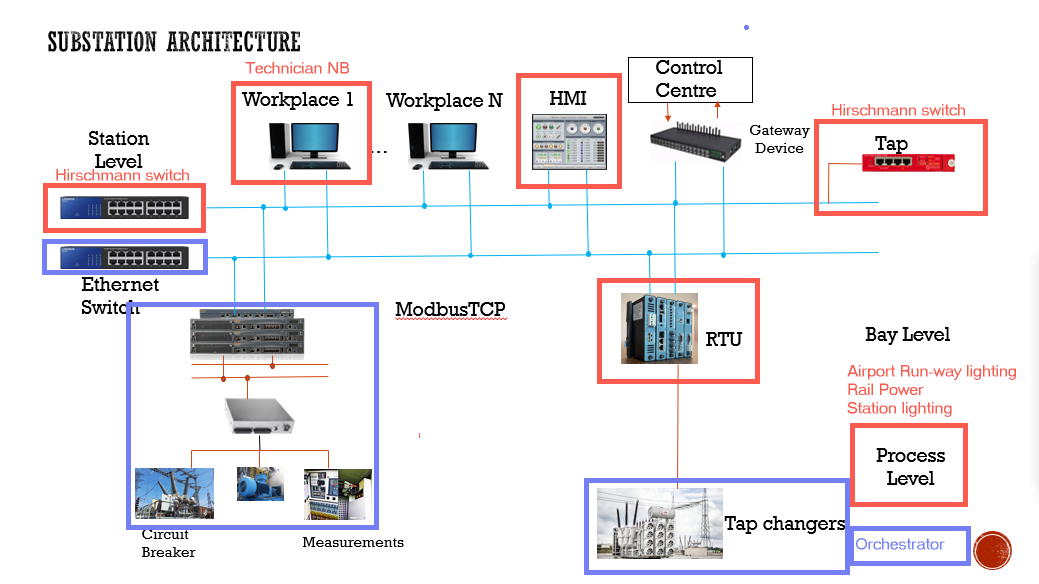
| **Program File** | **Execution Env** | **Description** |
| --- | --- | --- |
| M2PLC221.py | python 2/3 | This module is used to connect to the Schneider M2xx PLC. |
| pwrGenGlobal.py | python 3 | This module is used as a local config file to set constants, global parameters which will be used in the other modules. |
| pwrGenMgr.py | python 3 | Power generator auto-control manager. |
| pwrGenPanel.py | python 3 | This module is used to create different function panels. |
| pwGenRun.py | python 3 | This module is used to create the control panel to connect to the Raspberry PI generator control by UDP. |
| S1new\_192\_168\_10\_72.smbp | PLC | PLC-1 ladder diagram. |
| S3new\_192\_168\_10\_71.smbp | PLC | PLC-3 ladder diagram. |
| S7PLC1200.py | python 3 | This module is used to connect to the siemens s7-1200 PLC. |
| serialCom.py | python 3 | This module will inheritance the python built-in serial module with automatically serial port search and connection function. |
| pwrSibDisplay.py | python3 | This module is used to create a transparent, no window board live substation control parameter simulation display frame which shows overlay on top of the CSI OT-Platform main HMI program. |
| udpCom.py | python 3 | This module will provide a UDP client and server communication API. |
| tcpCom.py | python 3 | This module will provide TCP client and server communication API. |
| pwrSubParm.csv | excel | This file will save the 10 parameters data which used to simulate the substation working situation. |
| run.bat |  | Program execution script for Windows user. |
| BgCtrl.py | python | This module is used to create a background program controller: When we want to run a program in back ground with a loop inside, some time it is difficult for us to track whether the program is running or stopped if there are lots of other similar background programs are also running. This module is used to create a record file to record the current background program situation for the user to check and control. |

**3. Stealthy Substation Attack**

**3.1 Attack Introduction**

In the context of Smart Grids, our research has established that it is possible to craft stealthy attacks that can evade the attention of both the control centre (a computer system) and the human operator. Such stealthy attacks when crafted to introduce a set of malicious commands are referred to as a False Command Injection (FCI) attack in our research. These attacks are catastrophic resulting in black outs or widespread damages to grid users. For a smart grid or even a user of electrical energy, voltage of the supply is crucial. In other words, an erratic or abnormal voltage can damage equipment, and in certain cases, result in collapse of the entire grid. Voltages in a smart grid are controlled using various electrical devices or machines. One such device is the tap changing transformers. In our research, vulnerabilities of this device to stealthy attacks are studied along with techniques to detect intrusions that exploit these vulnerabilities. In this demonstration, our research is implemented on the platform.

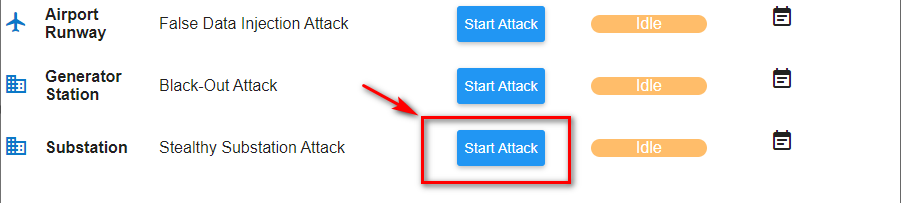
During the attack, the attack device will send the fake Modbus bus control message to PLCs to flick the PLC output, then confused the generator’s power auto control function. During the generator power supply doing the auto adjustment, the attack device will also send control command directly to the generator control raspberry PI to change the status of the generator and substation to paralysis the whole system. The attack flow diagram is shown below:



**3.2 Attack Launching and Detection**

3.2.1 Check the “Detection on” Checkbox (as shown in the section 1.1.3) on the substation information display window to turn on the Stealthy substation attack detection function.

3.2.2 Refer to <CSI OT 3D Platform Cyber Attack Web Design.pdf>; press the “Start attack” button under the Stealthy attack section to start the attack. (As shown below)



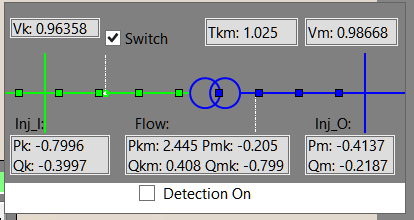
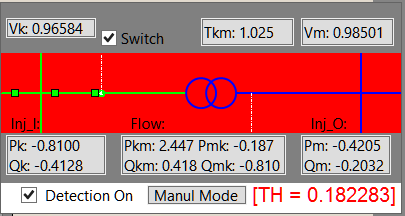
The attack will start after 10 seconds.

3.2.3 After the attack started, the attack situation will be different base on whether we have turned on the detection function.

|  |  |  |
| --- | --- | --- |
| **Idx** | **Without attack detection algorithm** | **With attack detection algorithm** |
| **0** | Airport runway light start flickering | Airport runway light start flickering |
| **1** | Inner track train stop/start moving | Inner track train stop/start moving |
| **2** | Effect lasted for 30 secs | Attack detected - Sound Alarm and Pop up on HMI |
| **3** | Switch off airport runway light | Effect lasted for 30 secs |
| **4** | Wait for 10 secs | Operator clicks on [Manual] button on HMI to switch the control to manual --- if not follow without detection scenario |
| **5** | Switch off train running in the inner track | Stop all the attack |
| **6** | Wait for 10 secs | Everything back to init state |
| **7** | City light change to red |  |
| **8** | Generator alerts stop |  |

During the attack, the substation information display window will show the threshold value calculated based on the substation working parameters and changed to red color:

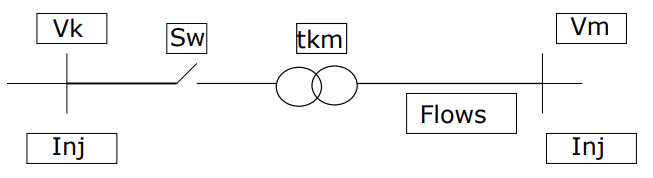
Normal state scenario: Attack on state scenario:

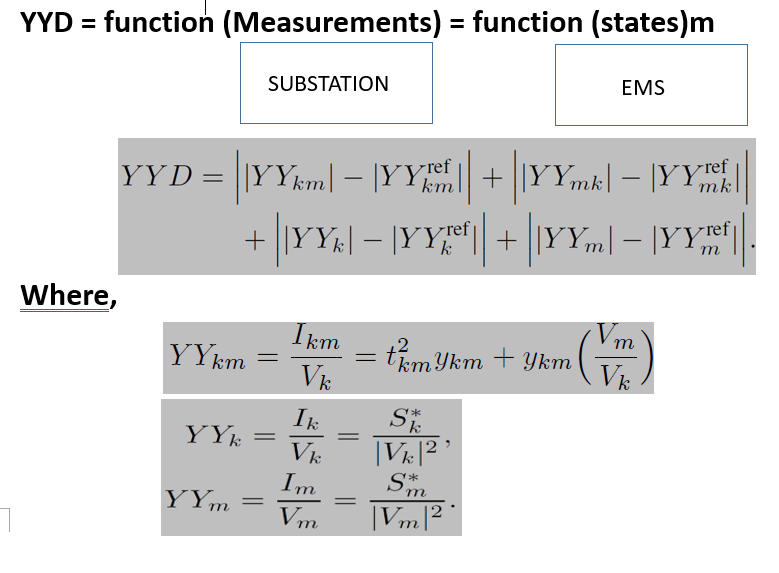
3.2.4. To STOP the attack, press the green color “Stop attack” button at the ‘Orchestration PC’. The ‘Training HMI’ will get back to normal state automatically after 5 to 10 seconds.

**3.3 Attack Detection Algorithm**

3.3.1 Parameter measurement position in substation:



3.3.2 Thresh hold value calculation formular:



=> If the calculated YY(m) is more than the trigger value (0.15), we specify the system was under attack.

3.3.3 Data sample:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **V1** | **V2** | **SW** | **t** | **P1** | **Q1** | **P2** | **Q2** | **P12** | **Q12** | **P21** | **Q21** |
| **Normal scenario** | 0.9703 | 0.9877 | 1 | 1.05 | -0.0226 | 0.0358 | 0.0071 | 0.0358 | 2.4487 | 0.9709 | -2.4277 | -0.7417 |
| **Attack demo**  **(Not stealthy)** | 1.0587 | 0.9837 | 1 | 0.9 | 0.0055 | -0.0106 | -0.0076 | 0.0142 | 2.4588 | -0.4022 | -2.4549 | 0.6642 |
| **Stealthy attack modifications** | 0.0878 | 0.9837 | 1 | 0.15 | 0.2781 | 0.2694 | -0.1606 | -1.6322 | 0.1606 | 1.6848 | -0.1606 | -1.6322 |

Normal conditions: Detection metric reads 0.06 which is less than trigger value(0.15).

Attack: Detection metric reads 2.73 which is more than trigger value(0.15).

**4. Reference**

S. Chakrabarty and B. Sikdar, "Detection of Hidden Transformer Tap Change Command Attacks in Transmission Networks," in IEEE Transactions on Smart Grid, vol. 11, no. 6, pp. 5161-5173, Nov. 2020.  
doi: 10.1109/TSG.2020.3005238

End (last edited 13/04/2021)