CYPRESS

Deliverable 1.4 - Specification of a cyber-physical real-time test environment

Tohid Behdadnia, Nadia Boumkheld, Geraint Chaffey, Rick Loenders
KU Leuven / EnergyVille

Date: October 28, 2022

Contents

Executive Summary

It is clear that there is an ever increasing cyber threat to power systems worldwide. In order to mitigate this threat and ensure systems remain resilient and performant in the face of evolving adverse cyber events, simulating both the physical power system and the cyber system that provides control, protection and automation is increasingly essential. Numerous studies rely on standalone simulation, however, performing a co-simulation of both the physical system and the cyber system is essential when considering the impact of one system on the other (e.g. the impact of a cyber attack on the physical system).

In addition to simulating the cyber system and the physical system, there is also the possibility of integrating real hardware into the simulation using a real-time simulation and laboratory setup. Such a configuration allows for testing of the response of the hardware to simulated events, and allows for the system-level impact of hardware to be evaluated. As part of the CYPRESS project a real-time co-simulation environment will be developed, allowing for the simulation of electrical power systems, simulation of cyber networks and systems, and laboratory testing of power system protection, control and automation systems. This real-time co-simulation environment will allow for the testing of the interactions and co-dependence between the different system elements.

This document describes the design specifications of the cyber-physical real-time test environment, including detailed description and motivation for the choices made for each of the elements as well as their interfacing.

The power system simulation will use selected benchmark networks - in common with the rest of the CYPRESS project - in order to show the power system impact of adverse cyber events. The simulation will be performed in real time on an RTDS simulator such that real hardware can be interfaced to the simulation, and complemented with Electro-Magnetic Transient (EMT) simulation in order to represent all power system phenomena - from steady state to very fast transients.

A cyber simulation will be performed to allow simulation of a range of adverse cyber events - malicious attacks as well as unintentional faults. Two types of cyber simulation are chosen for the studies - firstly traffic (information flow) modification using SNORT and Linux traffic control, and secondly an advanced simulation platform NS-3 in order to model the elements of a modern cyber system. The cyber simulations will be connected to the power system simulator such that control actions can be taken on the power system, and that measurements from the power system are passed to the cyber simulation.

Physical industrial control and protection hardware - identical to that used in real power systems - will be integrated into the cyber-physical co-simulation. This setup will use a wide range of industrial devices, including protection Intelligent Electronic Devices (IEDs/relays), Remote Terminal Units (RTUs) and industrial networking equipment. The use of this hardware 'in-the-loop' allows the stud-

ies performed in this part of the CYPRESS project to move beyond pure simulation studies - allowing analysis of the cyber security and system impact of industrial solutions. This configuration allows for a highly powerful and reconfigurable test system, for example looking at *vulnerabilities in devices that are in use in real power systems*, and *examining the possible power system impact of a variety of cyber attacks on the devices*. The aim is to have a test setup as close as possible to that in place in real world power systems. Coupled with the cyber simulation and the power system simulation, the overall test setup allows for realistic cyber attacks to be simulated, and propagated to the in-lab network to examine device and system impact.

The connection between cyber simulation, power system simulation, and the industrial hardware will require an interfacing arrangement. An interface will be provided between the NS-3 cyber simulator and the RTDS power system simulator for signals that require a direct connection between the cyber and physical simulations. The NS-3 simulator will also have a standard Ethernet interface for communicating with the in-lab industrial networks and devices. The SNORT and Linux traffic control packet manipulation is designed for connection within an Ethernet network and therefore requires no additional interfacing.

Summarizing, this document describes the specification, motivation and design for the real-time cyber-physical co-simulation environment. In the next phases of the project, the designs detailed in this deliverable will be implemented, for in depth testing of cyber attacks, penetration testing on real industrial devices, and examination of power system impact and possible mitigation.

1

Introduction

Power systems are in a period of fundamental change, driven by climate change and the massive integration of renewable energy [1]–[3]. One major change in the power system is the integration of new technologies, such as digital control and protection systems, and increased integration of control and protection through digital communications systems [4]. Such 'cyber' systems bring massive benefits in both cost and operational flexibility, but the integration of these new technologies also brings additional vulnerabilities and risk [5], [6].

In the context of increasing interdependence between the electrical power system and the cyber elements used to control and protect it, cyber security and the mitigation of cyber threats to the power system is an increasingly vital field of study.

To determine risks to the power system - and to evaluate mitigation methods - there are several techniques in place to perform cyber-physical studies in a simulated environment. Firstly, it is possible to evaluate each element individually - i.e. perform studies of the cyber system, and separately perform studies on the electrical power system. Each of these standalone elements typically use vastly different tools - e.g. analysis of cyber systems is often performed using event based simulators, while transient analysis of power systems is typically performed using time domain electromagnetic transient (EMT) environments. These 'standalone' studies can have significant value when evaluating aspects that have limited dependence on the other field, however, considering the increasing interdependence between cyber and physical systems, it is increasingly important to look towards co-simulation - that is, a simulation environment that considers both cyber and power system elements [7], [8].

Many aspects of cyber-physical dependence and risk can be evaluated according to fully simulation based testing. This is expected to provide insights into the majority of challenges faced in existing and future power systems. However, it can also be highly valuable to integrate real hardware into the cyber-physical simulation. This 'hardware-in-the-loop' setup makes use of real-time simulation - so called because one second in simulation time must match one second in real time. This allows

real hardware (e.g. industrial control equipment, protection relays, communications and networking infrastructure, ...) to interact with the simulated environment. These techniques are typically used to validate the response of real hardware to realistic stimuli, and to validate the system-level response to the real hardware. In the context of cyber security in power systems, this allows, for example, to perform penetration testing on a real industrial protection relay in the lab, and then for the system-level impact of that attack to be evaluated on the simulated power system and the simulated cyber network.

The real-time cyber-physical co-simulation is developed primarily to perform testing with real protection, automation and control hardware (e.g. digital substations, protection relays, IEDs, industrial control systems, SCADA, ...). This testing with real industrial hardware ('hardware-in-the-loop') is aimed at examining the performance of industrial hardware following cyber attacks, and also examining the system-level impact on both the power system and the cyber system.

In addition to testing with industrial hardware, the cyber-physical co-simulation environment will also be used for other simulated tests without real power system hardware - for example running entirely simulated cyber events and examining the impact on the simulated power system.

This task (T1.4) and deliverable (D1.4) aims to set out the specification, requirements and plans for the real-time test environment. The information set out in this document will be used to implement and develop the test environment and perform testing in later tasks. Chapter 2 describes the purpose of and and requirement for each of the elements within the system. Chapter 3 details the specification of each element along with an implementation plan. Chapter 4 then discusses the possible risks associated with the development and provides mitigating actions to reduce the impact of each risk.

2

Required elements for cyber-physical system co-simulation

This section describes the purpose of and requirement for individual elements of the cyber-physical real-time test setup. These elements will later be combined in order to perform the real-time co-simulation.

2.1 Electrical simulation

The electrical simulation will provide a time domain representation of the physical power system within the cyber-physical co-simulation.

The purpose is to simulate the electrical power system with all relevant steady state and transient behaviours to an appropriate accuracy (time step, model accuracy, parameter accuracy) such that power system phenomena and the impact of cyber events can be thoroughly analysed. The electrical system should represent the interactions and impact of attacks performed on the cyber-physical system. Let us consider the simulation of attacks at the substation level as an example. In this case the operation of devices handling the protection and control of a substation bay will be influenced by the cyber attack, which in turn interacts with the switch gear of all bays connected to a particular busbar. Power system simulation should therefore include the operation of these different switch gear elements to be able to observe the consequential steady-state and transient effects. The latter will depend among others on the modelling detail of generation sources, transformers, cables and lines. The level of detail mainly is decided by the steady-state and transient effects one wants to observe. Effects are not limited to the targeted substation bus, but can include other parts of the same substation and/or include cascading effects outside the substation to the remaining power system. Although the overall size and complexity of the power system model depends on the analysed effects, in reality

they are limited by the capabilities of computational hardware. Therefore, one needs to reduce size and complexity of models while retaining simulation results with sufficient accuracy concerning the electrical phenomena of the system.

The electrical phenomena broadly under study are:

- Analysis of steady state currents and voltages in normal operation and during contingencies
- Electro-mechanical transients for studies including frequency and transient stability
- Electro-magnetic transients (EMT) for subsynchronous transient analysis including studies with protection automation and control systems

These phenomena are in line with identified power system challenges observed in the literature for attacks on the cyber-physical system, and cover all foreseen electrical power system challenges.

A key requirement for this part of the CYPRESS project is for the simulation to be executed in real-time, such that it can be interfaced to real hardware in the laboratory as well as to the cyber simulation. The electrical simulation should have sufficient complexity (e.g. number of buses) to be of relevance for cyber studies, physical studies, and cyber-physical studies. Electrical power system test cases have been studied in more detail in CYPRESS Task 1.3, and the recommendations from this task will be followed in the implementation of the real-time test environment. The time step of the simulation model should be sufficiently small to represent the phenomena under study in a particular case, and also should be small enough so that the connected hardware (e.g. protection IED) receives stimuli at a suitable rate - as it would in a real power system / substation. The model should also be sufficiently accurate in order to represent the phenomena under study. The parameterisation of the components within the model should be such that the overall model and response is representative of a real power system for the phenomena under examination.

In summary, the electrical simulation therefore has the following key requirements:

Requirement E1 Simulation of all key identified electrical phenomena (steady state, electro-mechanical transients, electro-magnetic transients)

Requirement E2 Identified case study systems simulated in real-time, either in fully detailed represented or with suitable dynamic equivalent

Requirement E3 Capability to connect to physical Protection Automation and Control Systems (PACS) hardware in the laboratory as well as to the cyber simulation

Requirement E4 With the inclusion of hardware and/or software in a (closed) loop with the simulation, it is required that all electrical elements that take direct input or provide output to these objects should be modelled with the necessary interactive interface. The level of detail is defined by the case study.

2.2 Cyber simulation

The cyber simulation is a means to represent all of the relevant cyber elements of the cyber-physical systems. The purpose is to simulate all cyber contingencies, including both malicious attacks and unintentional faults, that affect the operation of cyber-physical power systems by generating changes to the information flow, i.e., disordering, distorting, dropping, or delaying the data transmitted throughout cyber information and communication technology (ICT) network. These malicious actions of changing or modifying the transmitted data are known as cyber attacks. Their objective is to compromise the confidentiality, availability and integrity of the cyber physical system. Attackers can target the network of a transmission or distribution system operator and cause mal-operation of computers and electronic controls, tripping of motors and generators, load shedding and cascading failure of the

system, disturbances of the power grid... Attacks including Denial of Service (DOS), Man in the Middle (MITM), false data injection, jamming channels... have severe impacts on the power system from blackouts to the destruction of critical infrastructures. Several attacks have been reported over the years, such as Stuxnet, a computer worm that infected the software of at least 14 industrial sites in Iran, including a uranium-enrichment plant[9]. it caused a physical malfunction in the centrifuges and reported that the system was working steadily by providing false feedback to the control system[10]. Another attack was generated in 2016 against the Ukrainian power grid. By gaining access to the distribution grid operator consoles, 30 substations were switched off, and around 230.000 peoples were affected by blackouts [11]. To deeply understand and evaluate the impact of cyber attacks on the power grid, we need to simulate the communication network of the cyber physical system under different attack scenarios, which will also help developing efficient counterattack measures to secure the overall cyber-physical system. To simulate, coordinate, and integrate the different elements of the communication network (computer devices,routers, switches...), advanced simulation platforms (i.e. NS-3, EVE-NG, OMNeT++, etc) are used in order to model the network entities using different types of connections/links, and to add network protocols and applications for the communication between these different network components. In addition to that, features of software like SNORT and Linux traffic control (TC) can be used to directly model the end-to-end information flow. Such end-to-end simulation avoids the complicated model of cyber systems and, subsequently, decreases the negative impacts of system uncertainties on the cyber contingency assessment.

In summary, the cyber simulation therefore has the following key requirements:

Requirement C1 Ability to simulate a cyber network containing the necessary number of nodes that participate in the specific application to be represented (state estimation, outage report, sending periodic measurements...).

Requirement C2 Ability to interface the cyber network simulation with the real time power simulator and the PACS hardware in the laboratory.

Requirement C3 Ability to interface the end-to-end information flow modelling with the real-time power simulator and the hardware in the laboratory.

Requirement C4 Capability to send and receive packets in real time, and performing various cyber attacks to intercept, modify, drop or delay the packets i.e. DOS attacks, False data injection...

Requirement C5 Evaluating the impact of cyber attacks on the normal operation of the cyber-physical system.

2.3 Physical lab hardware - protection, automation and control systems

The key motivation for designing a real-time cyber-physical co-simulation is to include hardware 'in-the-loop'. This real industrial hardware is then effectively integrated into the simulated power system.

The purpose is to include dedicated hardware relevant to both the power system and the communication network, including all required protection, automation and control systems according to the requirements of each case study. The hardware is able to confirm the expected performance using elements of real-world power systems, and allows elements of studies to be fully representative of a real power system.

The reason to include hardware is due to the inability to simulate every detail of a system, and all components, mainly because there is not enough reliable data available (including e.g. hardware or implementation related vulnerabilities), or there is no reliable method of benchmarking the simulation model, plus any model inevitably retains an unknown simulation error. Even when data is ample, the inclusion of hardware can often be the most efficient solution, given that it could save time which would otherwise be spent on modelling and its verification.

For cyber-security, the hardware component is either related to the attack vector, as part of the cyber-security detection and prevention, or utilizes/transmits the information stream. The inclusion of hardware eventually depends on the test objective. The cyber-physical power system both contains components related to the electrical system, such as devices related to the Protection, Automation, and Control System (PACS) of substations, and those from the communication network, such as Ethernet switches, routers, and servers.
In summary, the lab equipment (PACS) therefore has the following key requirements:

Requirement L1 Be representative (in function and configuration) of (relevant elements of) a real digital substation PACS.

Requirement L2 Provide suitable devices and interfaces such that penetration testing and cyber-physical security can be evaluated in a realistic environment.

3

Specification and implementation plan

In order to meet the requirements set out in Section 2, the following sections will set out the specification of each element of the system, and details will be provided of a plan for implementation according to the timeline of the CYPRESS project.

3.1 Electrical simulation hardware capabilities and test cases

The electrical simulation will be performed in an RTDS real-time simulator. At EnergyVille, there is a four rack CPU (RTDS PB5) based simulator with a large amount of auxiliary cards, for example analogue and digital inputs and outputs, communication cards, and FPGA simulators for intensive application specific simulation.

The exact simulation infrastructure used depends on the case to be simulated. Each rack can simulate up to 144 nodes, therefore over four racks 576 nodes can be simulated. This is a maximum limit and application of other elements (e.g. machines, circuit breakers, inputs, outputs, communications, controls,...) also requires computation.

The cases to be simulated will be defined, in part, by the work performed in CYPRESS T1.3 (resulting in D1.3) [12]. This activity has defined several case studies that will be used within the project.

Table 3.1 lists the selected case studies which were identified in CYPRESS deliverable 1.3. For the real-time testing work described in this document, systems with transient data are likely to be preferred. In case of computational constraints, reduced order systems will be developed to allow real-time simulation (with any reduced order models validated for the phenomena under study). Four distribution cases were also selected in CYPRESS deliverable 1.3 [12] but are not expected to be used in the real-time testing described in this document.

An additional test case is being used in the lab for initial testing of interfacing. The test case is based on the IEEE 9 bus system with three synchronous generator and three loads.

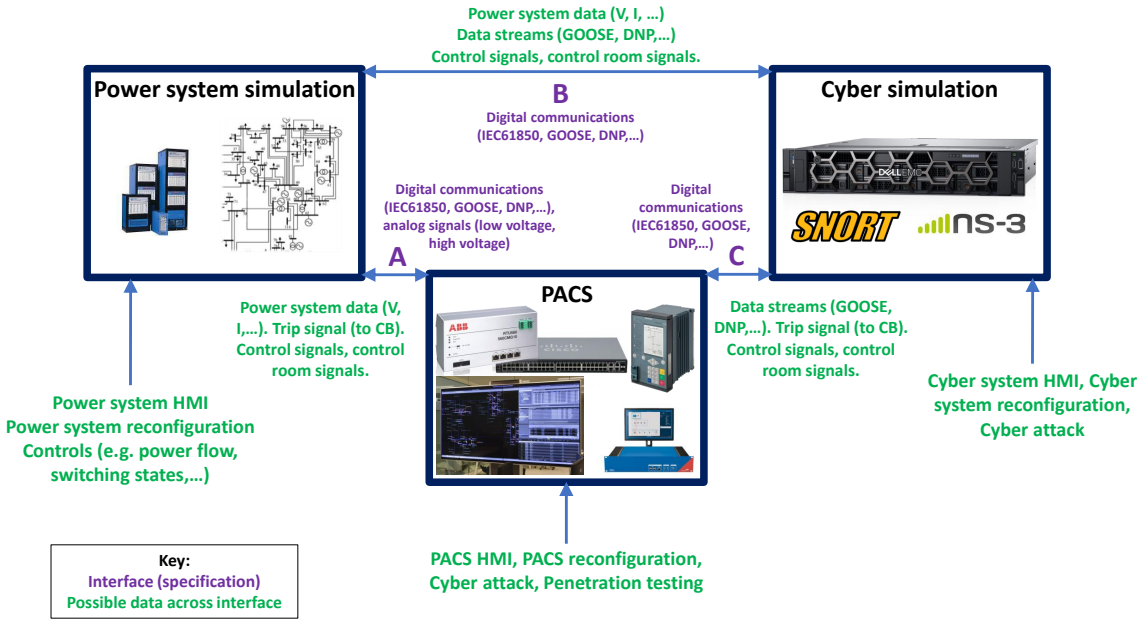

Figure 3.1: Summary of interfaces between different elements of co-simulation environment (power system simulation, protection automation and control systems (PACS) and cyber simulation). Diagram provides conceptual connections, note that not all connections are used in each case study.

The primary purpose of performing real-time simulation is in order to apply real hardware (or users) 'in-the-loop'. In this manner, the physical hardware receives realistic stimuli as if it were in a real power system (with stimuli updated in real time) and the simulation receives feedback from the physical hardware such that the system-level response can be evaluated. In the CYPRESS project, the purpose of the simulations (and overall tests) in the real-time simulation lab will be focused on testing the foreseen hardware, discussed further in Section 3.3. The hardware includes various elements of a digital substation and modern transmission system, and are generally considered part of the Protection, Automation, and Control System (PACS) of a substation (e.g. protection IEDs, control IEDs, merging units, and SCADA systems), and further includes elements from the communications/networking infrastructure (e.g. Ethernet switches and routers).

The electrical simulation must be suitable for the electrical phenomena under study, as well as being at a suitable time step for the hardware to be connected, and being executed in real-time.

In order to satisfy all of these requirements with one model, an electro-magnetic transient (EMT)

Test system	System type	Buses	Common applications	Transient data
IEEE Reliability	Transmission	73	Reliability, planning, operation, new technologies	
Roy Billinton	Transmission	6	Reliability, planning, analysis, new technologies	
IEEE 118-bus	Large	118	Planning, state estimation, cyber-security	x
ACTIVSg500	Large	500	AC power flow, transient stability	x
ACTIVSg2000	Large	2000	AC power flow, transient stability	x

Table 3.1: Selected power system case studies for transmission applications. Adapted from CYPRESS Deliverable 1.3 [12].

solver will be used in the real-time simulator, which is able to represent EMT as well as the slower phenomena (steady state, electro-mechanical transients).

Although three fully detailed case studies have been developed for use across the CYPRESS project, it may not be strictly necessary to model all of the details in the real-time studies. Given that the focus of these studies is on the physical hardware in the lab, as well as on the interaction with the cyber simulation platform, the electrical simulation must only represent the electrical elements in the vicinity to the physical hardware, while the rest of the electrical network could be aggregated into a dynamic equivalent. This could be beneficial due to either computational constraints (particularly important in real-time simulation) or due to modelling complexity and time. Where dynamic equivalents are applied, validation will be performed against the fully detailed network model in a different simulation environment.

Although the exact simulation capability is dependent on implementation and cannot be predicted with complete accuracy, the following is a provisional assessment according to expected implementation. Provisionally, it is expected that all of the selected test power systems can be simulated with the available real time simulation resources, although this will be confirmed later in the project when the implementation takes place.

3.2 Cyber system simulation capabilities

The cyber simulation will consist of two distinct elements - firstly a cyber network simulator and secondly end-to-end information flow modeling. The cyber network simulator is well suited for simulating operations of the entire network of the cyber-physical system, while the end-to-end information flow modeling will directly model the end-to-end information flow between a starting node and a terminal node.

NS-3 will be used for the network simulation of the cyber-physical system, to model the network components, add the communication links between them, generate network traffic, carry out attacks such as DDOS, MITM... and collect useful information from the simulations.

The information flow modeling will allow the testing of different phenomena - rather than simulating the operation of the whole cyber system one can implement the simulation from a perspective of the information flow, simulating an individual communication path. The simulation platform that we have proposed for modelling the end-to-end information flow takes advantage of two open source tools:

1. SNORT Intrusion Prevention System (IPS)
2. Linux Traffic Control (TC)

To perform an efficient packet modification using SNORT, already existing dynamic preprocessors, i.e., DNP3 and Modbus should be properly modified based on the simulation purpose. Furthermore, dynamic preprocessors for the protocols PMU, IEC104 and MMS can be added for further processing and modification (it is also possible to add preprocessors to process and modify the Generic Object Oriented Substation Event (GOOSE) and Sample Value (SV) frames). For simulating the traffic conditions, the Linux TC together with the “netem” queuing discipline is used.

3.2.1 Cyber network simulation: NS-3

NS-3 is a discrete-event simulator written in C++, and is developed for research and educational purposes. It is an open source software, licensed under the GNU GPLv2 license. NS-3 provides a platform for networking research and education - it allows users to perform simulation experiments, especially some experiments that are complicated and hard to conduct using real systems. NS-3 models internet networks and protocols, but it’s not limited to that. Non internet based systems can also be modeled. NS-3 has some characteristics which make it different from other network simulators such as:

- It is modular as it combines a set of libraries. Unlike other network simulators that offer a single interface where all tasks are performed, NS-3 uses external animators and data analysis tools.

- NS-3 is not a commercial product - it is supported by the ongoing contributions of the NS-3 research community.

In NS-3 the computing devices are called nodes and can be connected with a network using communication channels. Examples of these channels are CsmaChannel, PointToPointChannel and WifiChannel. To enable a node in the simulation to connect with other nodes, a NetDevice is installed in the node. There are many versions of NetDevice such as: CsmaNetDevice, PointToPointNetDevice, and WifiNetDevice. NS-3 also allows adding internet stacks (TCP, UDP, IP, ...) to the nodes in the simulation, and allocating IP addresses to the devices on these nodes. Furthermore, applications are used to generate traffic between the nodes, such as UdpEchoClientApplication and UdpEchoServerApplication which compose a client/server application to generate and echo UDP packets [13].
Queues are another very important component in NS-3. Outgoing packets cross two queueing layers before they reach the channel object. The first one is the 'traffic control layer' where active queue management and prioritization take place. The second queuing layer is the one in NetDevice objects (LTE, Wi-Fi...), for example the DropTail queue of default size 100 packets in the PointToPointNetDevice. It's worth mentioning that the first layer is only effective when notified by a NetDevice when the device queue is full so that it stops sending packets to the NetDevice. The size of the queues can be adjusted dynamically with the BQL (Byte Queue Limits) algorithm [13].
The main objective of running any simulation is being able to generate some output and extract data for study and analysis. In NS-3 there is a mechanism called tracing which allows to reduce the amount of data to be collected, by only tracing the events of interest, unlike the pre-defined bulk output mechanisms which require parsing and filtering to extract the information of interest. another advantage of the tracing system is that it allows the control of the output format directly (for example generate a form that is accepted by gnuplot) and avoid postprocessing with sed, awk... The tracing system in NS-3 uses two concepts: trace sources and trace sinks. Trace sources signal an event when it happens in the simulation and provides access to data. (i.e. a trace source indicates when a packet was dropped or received by a net device). To consume the information provided by the trace source, another entity is connected to the trace source which is the trace sink. Ns-3 device helpers can be used for creating two forms of tracing files: ASCII files and PCAP (Packet capture) files. PCAP files can be read and displayed using Wireshark or other traffic analyzer tools such as tcpdump [13]. Table 3.2 summarizes the different NS-3 Components and their functions.

Function to Perform	NS3 Component used	Examples
Enable the connection between nodes	NetDevices	CsmaNetDevice, PointToPointNetDevice, WifiNetDevice
Media over which a node is connected to a network	Channels	CsmaChannel, PointToPointChannel, WifiChannel
Generate an activity to be simulated	Applications	UdpEchoClientApplication, UdpEchoServerApplication
Enqueing, dequeing and droping operations	Queues	DropTail
Generate an output and extract data from the Simulations	Trace Files	ASCII and PCAP

Table 3.2: NS-3 Components and their functions

3.2.2 End-to-end information flow modelling: SNORT + Linux TC

SNORT: SNORT is an open-source IPS, owned by Cisco, with the capability of performing real-time packet logging and traffic analysis on Internet Protocol (IP) networks. SNORT is capable of performing content matching/searching and protocol analysis, which is useful for detecting a variety of probes and attacks (i.e. such as stealth port scans, buffer overflows, Common Gateway Interface (CGI) attacks,Operating System (OS) fingerprinting attempts, Server Message Block (SMB) probes, etc). By default SNORT only supports DNP3 and MODBUS out of the SCADA/ICS protocols which is not sufficient for fulfilling requirement of cyber-physical power system studies. Thus dynamic-preprocessors (for the IEC104, MMS and PMU protocols) and preprocessors (for the GOOSE and the SV protocols) are added to process the application layer and Ethernet layer packets, respectively. Besides, for making the data processing more efficient, Glib library are used for building data structures such as linked lists, hash tables, and strings.

Linux TC: In the Linux Operating System (OS), the packets are classified based on filters before either entering the Kernel (form network interface card) or entering network interface (from kernel). Then they are joined ingress/egress queues based on the classification and gets processed according to the queuing discipline (queuing discipline is a packet queue with an algorithm which decides when to release the packet). In the ingress queue it is only possible to drop the packets, while in the egress queue we can use the queuing discipline network emulator "netem" to do any of the following:

- Add a delay to the packet
- Add jitter to the packet
- Corrupt a packet
- Drop a packet
- Duplicate packets

Defining filters, traffic classes and queuing disciplines: The Linux TC is part of the iproute2 package that is used to define filters, traffic classes, queuing disciplines. In our proposed approach, first a root queuing discipline is attached to a network interface. This root queuing discipline should be classful, so that classes can be defined under it. Then, for the defined classes, it is possible to attach queuing disciplines by user. Eventually, filters are used to filter the packets into these classes.

To add user specified variable, i.e. delay, drop, duplication, and corruption, a queuing discipline line "netem" on the defined class is used. In particular, the initial values of these variables can be either set based on the probability distribution (i.e. Uniform, Normal, Pareto and Pareto-normal) or defined in correlation with the previous values and/or fate of the earlier packet (The graphical user interface (GUI) will take care of this).

3.3 Physical lab hardware

The lab hardware will consist of various configurations of power system protection, automation and control systems, including e.g. protection IEDs, RTUs, SCADA systems and communications equipment.

The exact configuration depends heavily on the application case. Some first example cases are shown below. The final demonstration cases will be developed according to the experiences within the rest of the CYPRESS project. The example cases below are for the purposes of example and allow the discussion of hardware and interfacing requirements.

3.3.1 Available protection, automation and control hardware

The previous sections discuss example cases and specific lab hardware requirements. Given that the devices used are preferably common within a industrial setting - used in real substations - they provide effective validation of the applicability/relevance of the cyber attacks and mitigation developed in the CYPRESS project. To ensure relevance of results, the verification/validation of electrical power system and cyber system case studies must be ensured. There is a wide range of equipment available to the project, which can be used in various studies. For completeness, a list of relevant equipment available at the time of writing is provided here:

- Networking equipment
  - 2 x Ruggedcom RS900NC switch
  - 1 x Juniper ex2200 router
  - 1 x Juniper ex4300 router
  - 3 x TPLink unmanaged switch
  - 2 x Ubiquiti EdgeSwitch
  - 2 x CISCO IEEE4000 switch
- Protection relays/IEDs (IEC61850 enabled)
  - 1 x Schneider SEPAM S82 (overcurrent)
  - 3 x Siemens 7SJ85 (overcurrent)
  - 2 x Siemens 7UT85 (transformer differential)
  - 4 x Siemens 7SA86 (3pole distance)
  - 2 x Siemens 7SD86 (Line differential)
  - 1 x Siemens 7SS85 (busbar differential)
  - 1 x Siemens 6MU85 (merging unit)
  - 1 x ABB REL670 (overcurrent)
  - 1 x ABB REB670 (busbar differential)
  - 1 x ABB RET670 (transformer differential)
  - 1 x ABB SAM600 (4 module merging unit IO + TS + CT + VT)
  - 1 x ABB SMU615 (merging unit)
  - 1 x GE MICOM P45 (transformer differential)
  - 1 x GE B30 (busbar differential)
  - 1 x Efasec (overcurrent)
- RTU
  - 1 x ABB 540CMD01
  - 1 x Siemens AK3
- SCADA control
  - GE PowerOn: ADNM SCADA interface and control

3.3.2 Foreseen implementations

The range of PACS industrial hardware can be used in a variety of configurations, and will be arranged according to the requirements of the project. In each case, the PACS will be configured and arranged in a manner representative of a real world (present or future) power system substation such that results have relevance and potential impact.

For this laboratory work, PACS devices that have a digital communications interface (i.e. Ethernet, IEC 61850,...) will be the focus of the investigations, given that these network connected devices are arguably the more likely target for a cyber attack. There are numerous PACS devices available to the project that do not have a digital communication interface, therefore there is the possibility of also performing penetration testing and simulated cyber attacks on these devices if deemed of interest in the later stages of the project.

The first implementations will be focused on penetration testing activities in order to determine and exploit weaknesses in the industrial hardware. This will first be performed within the laboratory cyber network, before the possibility of integrating the exploit into the cyber network simulation is explored.

3.4 System-level configuration and connection/interfacing (system integration)

The described system consists of a cyber simulation (NS-3), an information flow modifier (SNORT + Linux TC), a power system simulation (RTDS) and a variety of in-lab industrial hardware (PACS). A key element of the overall real-time cyber-physical co-simulator is the interfacing between each of these different elements. The following sections describe the interfacing requirements between the different elements.

3.4.1 Interfacing of the RTDS simulator with other system elements (through the GTNET card)

The GTNETx2 card provides a real-time communication link to and from the RTDS simulator via Ethernet. Each GTNETx2 card has two modules. Each module has one Ethernet port, and each port is operated separately and may be equipped for one of three connection options: 100/1000 Copper, 100BASE-FX, or 1000BASE-SX. Each module can have 1 active network protocol at any given time. In other words GTNETx2 cards can operate two network protocols at the same time. This is the list of protocols supported by GTNET [14]:

- GTNET-SKT: the GTNETx2 communicates with external software and physical equipment over a Local or Wide Area Network connection using TCP or UDP sockets. GTNET-SKT is capable of sending up to 300 input and output data points (4 bytes for each point) per packet. The data transmitted can be of either integer or floating-point (IEEE 754) type. GTNET-SKT can communicate with multiple external devices (up to 10), and can manage up to 30 input and 30 output data points for each channel.
- GTNET-MODBUS Modbus communicates over TCP/IP networks using the GTNETx2 hardware. There is a support for Modbus TCP, Modbus RTU over TCP, and Modbus ASCII over TCP. The Modbus server (slave) communicates to a Modbus master station (eg: an external IED)through the GTNETx2's Ethernet port.
- GTNET-GSE Provides IEC 61850 Ed 2.0 and 2.1 GOOSE communication and MMS server services. GTNET-GSE can simulate one to four IEDs, which can be configured to publish/subscribe (up to 16 message streams) regular or routable GOOSE. GTNET-GSE receives messages from up to 32 external IEDs. The GTNET GOOSE configuration is done via the IED configuration tool (within RSCAD). IEDs simulated by the GTNET-GSE component can include standard IEC 61850

data models. The GTNET-GSE also supports simulated switch-objects to represent and test switchgear operations, test (simulated) GOOSE messages, and GOOSE subscription monitoring.

- GTNET-SV Provides IEC 61850-9-2 / IEC 61869-9 Sampled Value messaging for power system voltages and currents. GTSYNC card is used to timestamp the sampled values and GOOSE messages. GTNET-SV can publish up to two independent SV streams simultaneously in output mode, with each IEC 61869-9 stream having up to 24 channels (sampling rates of 80 and 96 samples/ cycle and 4800 Hz). GTNET-SV can subscribe to one SV stream in Input mode. It can be configured to publish/subscribe regular SV or routable SV. In addition to that GTNET-SV supports connection-oriented IEC 61850 MMS client/server communication.

- GTNET-PMU Two PMU streaming options are available for GTNET-PMU. The first option is a single GTNET-PMU firmware which can represent and provide output for up to eight PMUs (with a rate between 1 and 60 frames per second) with symmetrical component information related to 3-phase sets of voltage and current using UDP or TCP connections. The second option is a single GTNET-PMU firmware which can provide output for up to twenty four PMUs (Frame rates up to the system frequency (50/60 Hz)) containing only positive sequence data.

- GTNET-DNP, -104 There are two options: GTNET-DNP and GTNET-104 firmware which allow GTNETx2 to act as a DNP3 or IEC 60870-5-104 slave. GTNET-DNP can be configured as 1 outstation communicating with up to 4 masters, or 10 outstations communicating with up to 2 masters per outstation. GTNET-104 represents one outstation communicating with up to 4 masters.

- GTNET-PB Is used to read large data files stored on a PC hard drive (using TCP/IP Ethernet communication) and allow them to be played back in an RTDS simulation. Sampling frequencies of 20 kHz can be maintained in a true real time simulation, and up to 48 channels of floating point data can be imported from an external data file.

3.4.2 Interfacing of NS-3 with RTDS and other physical hardware

Sometimes there is a requirement to mix real and simulated entities. To do this with NS-3 or to connect NS-3 with a real device we can use File Descriptor NetDevice (FdNetDevice) which reads the traffic to and from a file descriptor that may be associated with a socket, a user-space process... To create and configure a file descriptor, helpers can be used, such as EmuFdNetDeviceHelper which creates a raw socket to an underlying physical device, and provides the socket descriptor to the FdNetDevice. This allows a simulated ns-3 node to send and receive packets to and from a real network device. The interaction between an NS-3 node and a real device can be run in real-time emulation mode. In order to implement the real time feature between the NS3 environment and a real device, RealTime scheduler is used. This scheduler synchronizes the progression of the simulation clock with an external time base (e.g Wall clock) [15].
Tap NetDevice is another device that can be used to allow a host system or virtual machines to interact with a simulation.

The connection/interfacing between RTDS and NS-3 that runs on a Linux machine (with one or more network interface cards) is achieved through a GTNET card of the RTDS simulator as shown in Figure 3.2, where the emulated NS-3 node is connected to the Ethernet port of the Linux machine. For example to send packets from a relay that has an interface with the GTNET card to an NS-3 node, This is done in the following way:

- The data is calculated at the RTDS simulator

- The GTNET interface corresponding to the relay sends the data to the emulated NS-3 node

- The emulated node forwards the data to the destination node in the simulation.

We can make use of the NS-3 TAP interface to modify the DNP3 data (i.e function code, group, variation, index, value...) between a client that runs RSCAD, and a server that is interfaced with the GTNET card. NS-3 can also be used to generate attacks such as DOS and MITM. This representation is shown in Figure 3.3

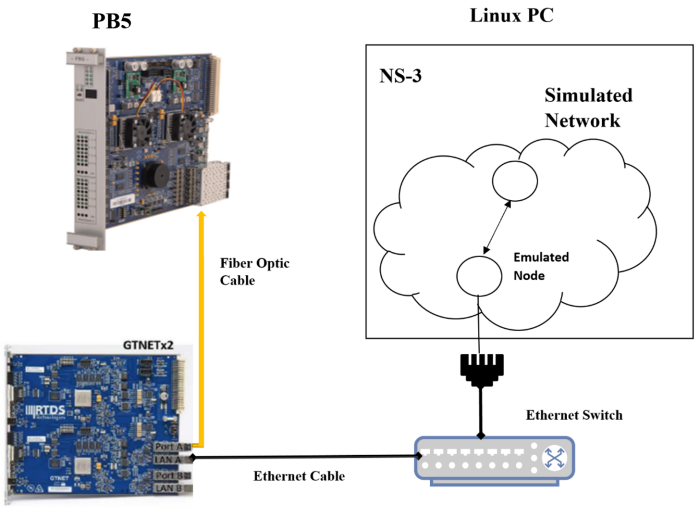

Figure 3.2: NS-3 and RTDS Interfacing using EMU Interface

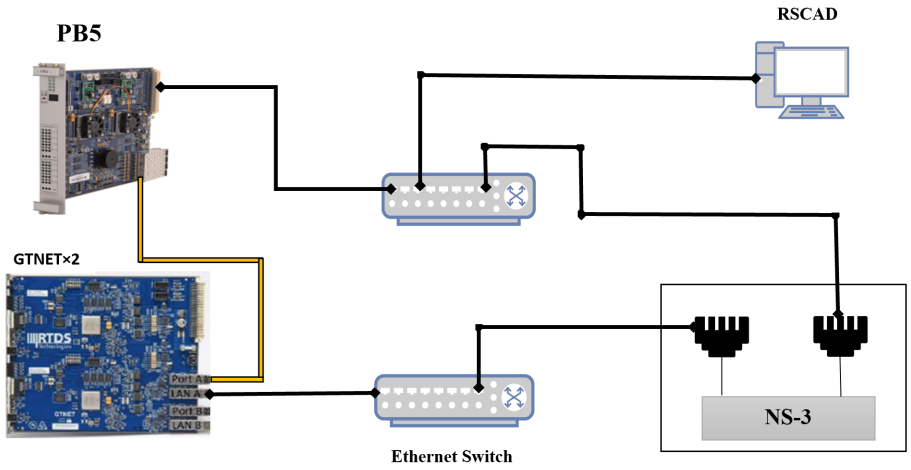

Figure 3.3: NS-3 TAP interface for the modification of DNP3 data

3.4.3 End-to-end information flow modelling - SNORT + Linux TC

In our purposed end-to-end information flow modelling approach, an inline packet modifier, that leverages inline mode of SNORT, is used. The position of the inline packet modifier can change depending on the protocol (whether it is an Ethernet layer protocol or an application layer protocol). To make the data modification easier, a graphical user interface is developed and connected to the Linux PC running SNORT via SSH, so that, the user can use the graphical user interface to modify the values in the given protocols.

For the Ethernet multicast protocols, if it is not intended to drop/delay the frames, it is not necessary to directly connect the inline packet modifier to the connection between the server/publisher and the client/subscriber, while for the session based application layer protocols, the inline packet modifier have to directly connect to the connection between the server/publisher and the client/subscriber.

As shown in Figure 3.4 the PC that runs the inline packet modifier has five Ethernet interfaces. These Ethernet interfaces can be connected via a bridge for running SNORT and Linux TC. The inner network which contains the GTNETx2 cards, is connected to one of the interfaces that runs SNORT, while the outer network is connected to one of the interfaces that runs TC. The last Ethernet interface is connected to the Windows machine for running graphical user interface. As it is mentioned earlier, depending on the protocol our designed packet modifier can be positioned in the following ways:

- Positioning at the server side (see Figure 3.5)
- Positioning at the client side (see Figure 3.6)
- Positioning for Ethernet Multicast protocols (see Figure 3.7)

Note that Figures 3.4 to 3.7 show typical configurations for a large proportion of the tests. Depending on the case study under examination, and the cyber event being simulated, adapted configurations may be applied.

In the first and second setups (positioning at the server side and positioning at the client side), the GTNETx2 perform the role of the Master/server for MODBUS, DNP3, IEC104, MMS and PMU protocols. Since all the data are travelling through the Inline packet modifier,it can change the data according to what is specified in the GUI. This modified data is received by protection, automation, and control (PAC) system and affects it. Consequently, the users can find out the robustness of the PAC system to data manipulation. Furthermore, the user can find out the adverse effects of manipulating certain data points.

The third setup (positioning for Ethernet Multicast Protocols) is used for the Ethernet multicast protocols like GOOSE and SV, where the packet modifier should be positioned in between the publisher and the network switch if dropping or delaying the packets are needed.

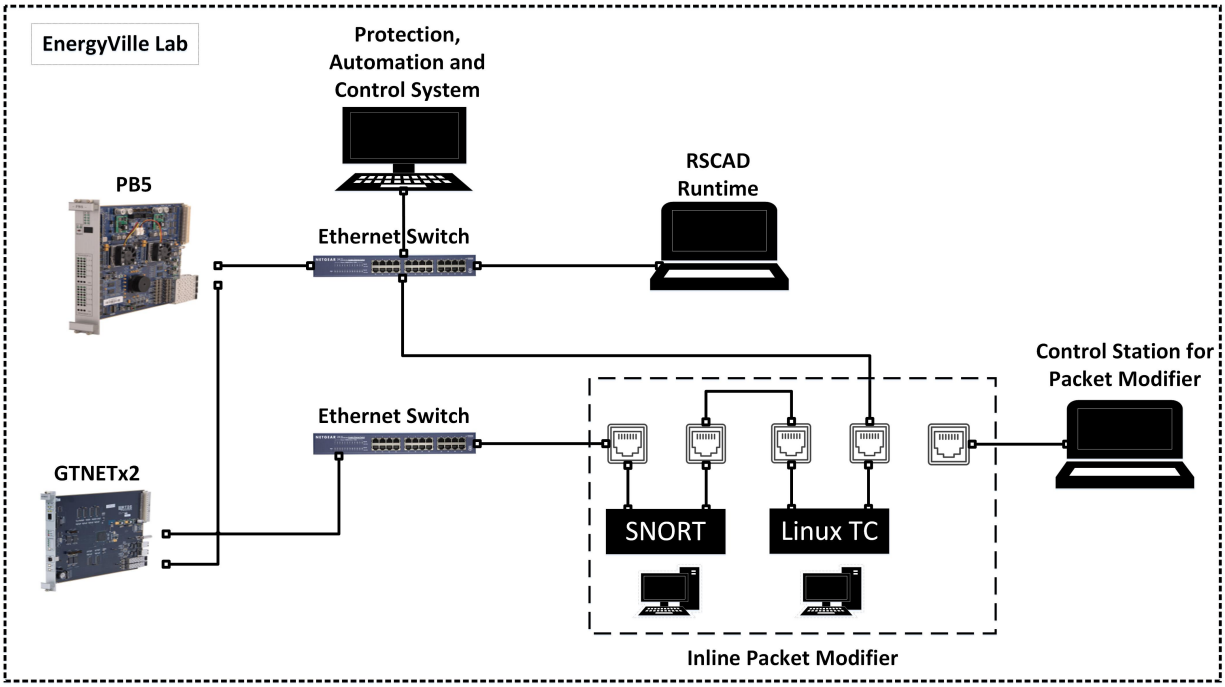

Figure 3.4: Overall Connection Diagram "SNORT+TC <-> RTDS" (typical configuration)

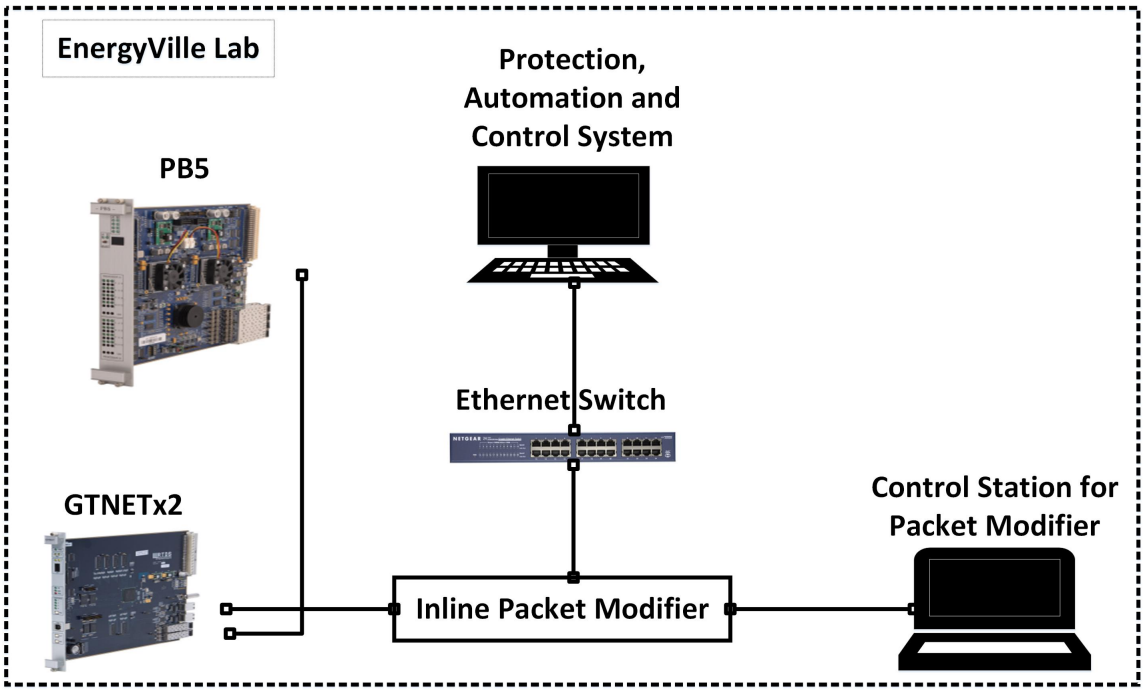

Figure 3.5: Positioning at the Server Side (typical configuration)

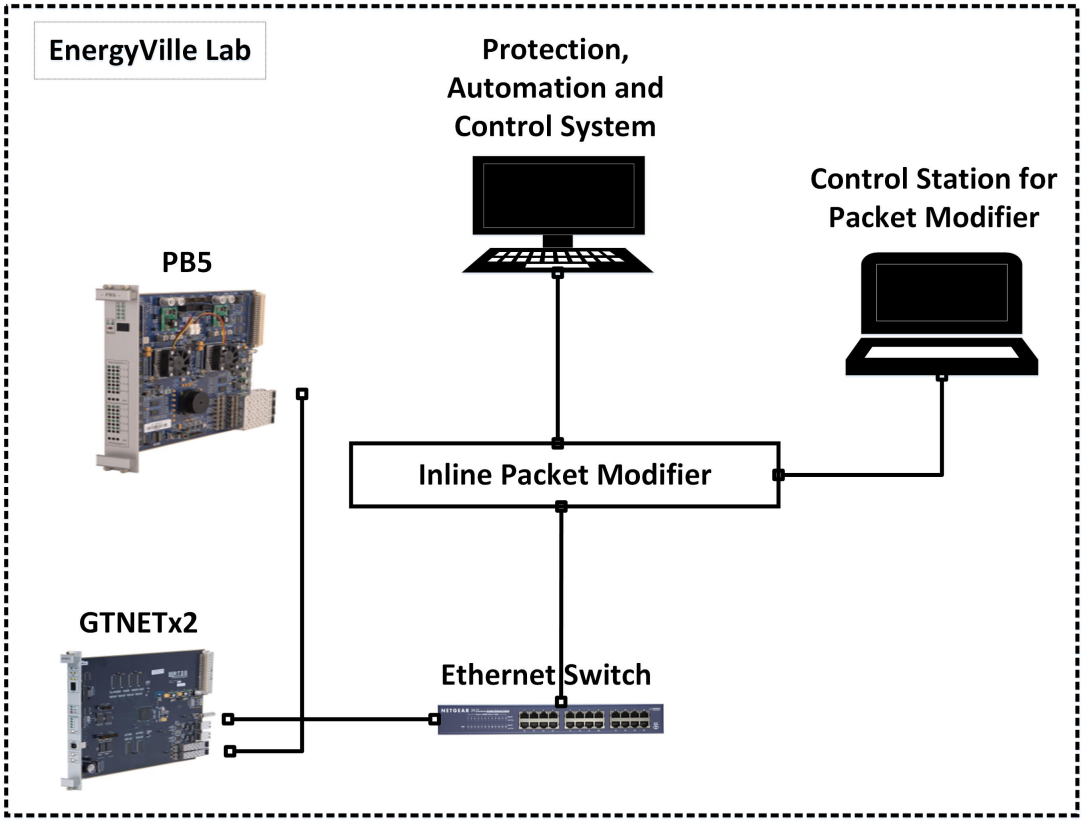

Figure 3.6: Positioning at the Client Side (typical configuration)

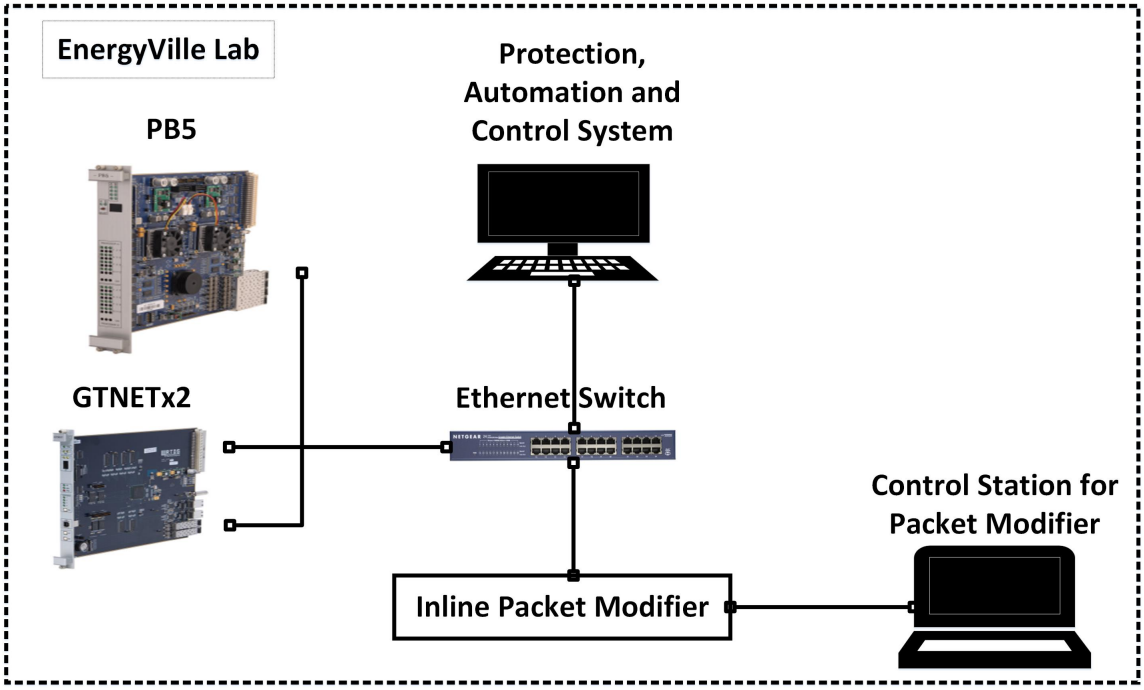

Figure 3.7: Positioning for Ethernet Multicast Protocols (typical configuration)

4

Risks and mitigation

In order to identify possible problems in the development and execution of the real-time cyber-physical co-simulator, and methods to mitigate these problems, the following section identifies possible risks to this implementation, and mitigation methods in order to ensure that the project can continue even considering foreseen risks.

4.1 Electrical simulation

Risk 1 (R1): Choice of case study network is too large to be simulated in real-time with the available resources.
Mitigation: A choice of case studies are being selected for use across the CYPRESS project. Most of the case studies will be able to be simulated in real-time in their entirety, however it is possible that the largest case study will not be possible to fully simulate in real-time. It it expected that not all case study networks will need to be evaluated in real-time simulation, which focuses on the physical hardware in the lab. However, if the largest case study system is required for a particular analysis and the full network cannot be simulated, a network reduction will be performed such that the relevant features for the lab hardware are retained. The reduced order electrical network response will be validated against the full network in order to ensure results are comparable for phenomena of interest.

Risk 2 (R2): Electrical simulation cannot represent all phenomena under study
Mitigation: State-of-the-art EMT simulation will be used, which is capable of representing all relevant steady state and dynamic phenomena. The case study systems have been developed in a specific task within the CYPRESS project to ensure that suitable test systems are chosen that are suitable for the full range of studies required.

4.2 Cyber simulation

Risk 3 (R3): FdNetDevice class in NS-3 is able to read and write traffic using a file descriptor provided by the user. To avoid overwhelming the scheduler (that schedules the reception of the frame by the device upon the reception of the frame) when there is a high incoming data rate, a counter is used for the number of frames scheduled to be received. If the counter reaches the RxQueueSize (this vale limits the number of packets that have been read from the network into a memory buffer but have not yet been processed by the simulator) attribute in the device, the new frame will be dropped.

Mitigation: When sending data towards NS-3, The number of packets should not exceed the maximum size of the read queue (RxQueueSize).

Risk 4 (R4): Out of the supervisory control and data acquisition (SCADA)/industrial control systems (ICS) protocols, SNORT only supports MODBUS and DNP3. Those may not be sufficient to perform the data modifications that we need.
Mitigation: Dynamic preprocessors for the MMS, IEC104 and PMU protocols and preprocessors for the GOOSE and the SV protocols should be added and then modified manually.

4.3 Physical lab hardware

Risk 5 (R5): Lab hardware damaged by penetration testing and not possible to perform repetitive or reproducible tests.
Mitigation: Experienced cyber security team are able to perform non-destructive penetration testing on in-lab devices. Segregated networks will be used to ensure that cyber attacks only affect equipment under use in the project and do not inadvertently disrupt or damage other EnergyVille systems. Redundant devices are available in the exceptional case that targeted devices are damaged.

Risk 6 (R6): Limited available equipment could limit scope of lab testing.
Mitigation: Penetration testing will first focus on examining vulnerabilities in particular devices, before work moves towards full cyber simulation and testing of cyber attacks. A wide range of equipment is available in the lab, however, it will of course not be possible to represent a full power system in hardware. Hardware under test will be focused on, e.g. in one substation or adjacent substations according to the requirements of the test. It will be possible to represent the rest of the elements in the system at an appropriate level of detail in the cyber simulation and/or the electrical simulation.

4.4 System-level configuration and connection/interfacing (system integration)

Risk 7 (R7): Synchronisation of time-domain simulation (electrical) and event-based simulation (cyber) is not possible at the full scale of the planned networks.
Mitigation: Some technologies already provide the functionality to run event-based cyber simulations in a mode synchronised with real-time. If the computational and interface requirements of the fully detailed networks are not possible to fully replicate in a lab environment real-time co-simulation, reduced order representations (e.g. reduction of number of interfaces, reduction of computational complexity or cyber simulation) will be developed and validated against full scale simulations.

5

Conclusion

To allow testing of cyber-physical security of real power system components, a specification for a real-time cyber-physical co-simulator has been developed. This specification will be used to develop the laboratory setup in order to meet the needs of the real-time studies within the CYPRESS project.

This document has identified the key distinct parts of the cyber-physical real-time co-simulation platform; physical (electrical power system) simulation, cyber simulation, interfacing, and industrial Protection, Automation and Control System (PACS) hardware in the laboratory.

A specification determining the requirements of each of these elements has been developed, including specific requirements on which the success of the system, when implemented, can be evaluated.

A implementation plan for each element of the co-simulation system has been developed, along with a plan for integrating the elements together into the overall co-simulation platform.

Any laboratory implementation involving hardware and interfacing different environments carries some risk that tasks will overrun or it will not be possible to complete all activities. In order to manage these risks, foreseen challenges have been identified and suitable mitigation has been proposed such that the project outcomes are not adversely affected. In this manner, it is suggested that reasonable outcomes will be gained even in case of adverse events.

Following the development of the real-time cyber-physical co-simulation, and the associated laboratory setup, a wide range of prospective tests will be possible. In addition to those detailed in this document, future plans will include penetration testing of the Protection, Automation and Control Systems (PACS) in the laboratory. Foreseen steps include: asset discovery, scanning and enumeration, vulnerability analysis, exploitation of vulnerabilities, privilege escalation, lateral movement and reporting.

In summary, specification and planning has been developed for the implementation of the cyber-physical real-time co-simulation environment. This document will be used as a framework for the next phases of the CYPRESS project - implementation and testing within the cyber-physical co-simulation

laboratory environment.

Bibliography

[1] M. Welsch, "Europe's energy transition," in *Europe's Energy Transition*, Elsevier, 2017, pp. 3–6.

[2] S. Potrč, L. Čuček, M. Martin, and Z. Kravanja, "Sustainable renewable energy supply networks optimization–the gradual transition to a renewable energy system within the european union by 2050," *Renewable and Sustainable Energy Reviews*, vol. 146, p. 111 186, 2021.

[3] S. Tagliapietra, G. Zachmann, O. Edenhofer, J.-M. Glachant, P. Linares, and A. Loeschel, "The european union energy transition: Key priorities for the next five years," *Energy Policy*, vol. 132, pp. 950–954, 2019.

[4] D. Wilson, J. Wright, S. Richards, and P. Mohapatra, "Functionality, design and benefit analysis of end-to-end digital substation," 2016.

[5] P. Pinchukov and S. Makasheva, "Estimation of digital substation reliability indices," in *International Scientific Siberian Transport Forum*, Springer, 2019, pp. 3–14.

[6] A. Khodabakhsh, S. Y. Yayilgan, S. H. Houmb, N. Hurzuk, J. Foros, and M. Istad, "Cyber-security gaps in a digital substation: From sensors to scada," in *2020 9th Mediterranean Conference on Embedded Computing (MECO)*, IEEE, 2020, pp. 1–4.

[7] P. Hehenberger, B. Vogel-Heuser, D. Bradley, B. Eynard, T. Tomiyama, and S. Achiche, "Design, modelling, simulation and integration of cyber physical systems: Methods and applications," *Computers in Industry*, vol. 82, pp. 273–289, 2016.

[8] R. V. Yohanandhan, R. M. Elavarasan, P. Manoharan, and L. Mihet-Popa, "Cyber-physical power system (cpps): A review on modeling, simulation, and analysis with cyber security applications," *IEEE Access*, vol. 8, pp. 151 019–151 064, 2020.

[9] D. S. Kushner, "The real story of stuxnet," *IEEE Spectrum*, vol. 50, pp. 48–53, 2013.

[10] M. Z. Gunduz and R. Das, "Analysis of cyber-attacks on smart grid applications," in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, 2018, pp. 1–5. DOI: 10.1109/IDAP.2018.8620728.

[11] S. Tufail, I. Parvez, S. Batool, and A. Sarwat, "A survey on cybersecurity challenges, detection, and mitigation techniques for the smart grid," *Energies*, vol. 14, no. 18, p. 5894, 2021.

[12] A. Godfraind, "Report d1.3 describing the benchmarks," 2022.

[13] *Ns-3 tutorial*, https://www.nsnam.org/docs/tutorial/html/, 2022.

[14] RTDS Technologies, *Gtnetx2: The rtds simulator's network interface card*, https://knowledge.rtds.com/hc/en-us/articles/360034788593-GTNETx2-The-RTDS-Simulator-s-Network-Interface-Card, 2022.

[15] NS-3 Manual, *Real time*, `https://www.nsnam.org/docs/manual/html/realtime.html`, 2022.

This project is supported by the Belgian Energy Transition Funds