CYBER SECURITY USE CASES IN A SMART POWER DISTRIBUTION CYBER-PHYSICAL TESTBED

A Thesis

by

VINICIUS MAZZILLI BOBATO

Submitted to the Graduate and Professional School of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Dr. Ana Goulart
Committee Members,	Dr. Rainer Fink
	Dr. Katherine Davis
Head of Department,	Dr. Bimal Nepal

December 2024

Major Subject: Engineering Technology

ABSTRACT

Power distribution systems and communication networks are inter-dependent. Remote data collection and control of distributed energy resources (DERs), such as windmills and solar panels, employ not only remote terminal units (RTUs) but also Aggregators, Distributed Service Operators (DSOs), and smart meters. Aggregators are entities that communicate with several DERs in its portfolio. It mediates the communication between the DSO and the DERs.

These devices are located in different places and exchange data through a wide area network, i.e., the cyber sub-system. To help us understand this cyber sub-system and the impact of a cyber attack on a smart power distribution system, this thesis dissertation presents the process to build a real-time cyber-physical testbed as well as two use cases we simulated.

The power distribution system is simulated using Real Time Digital Simulator (RTDS) power simulator, while the communication network is emulated using Common Open Research Emulator (CORE). In our current work, the first use case simulates a peak shaving application, in which the DSO communicates with the Aggregator to find the capacity of four DERs to generate more power. The second use case, utilizes open-source data and a realistic approach to simulate an Advanced Metering Infrastructure (AMI) network. Both use cases contain an intruder that performs a reconnaissance attack, a denial-of-service attack (DoS), and a Man-in-the-Middle (MITM) attack to impact either the system's availability, integrity, and confidentiality.

DEDICATION

To my parents: Your support, boundless love, and endless encouragement have been the foundation of everything I have achieved. You taught me the value of hard work, perseverance, and believing in myself, and for that, I am forever grateful. This thesis is a reflection of your sacrifices. I cannot thank you guys enough.

To my grandmother, uncles, and aunts: Your encouragement and love shaped me in more ways than I can express. Each of you, in your own way, has contributed to the person I am today.

To my beloved fiancé: Your love, patience, and constant belief in me have been my greatest source of strength throughout this journey. You have stood by me through every challenge and celebrated every success. I love having you as my number one fan.

With love and gratitude,

Vinicius.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

These were the main contributors to my work and I am very thankful for their support:

- Kumpanat Thongmai (Pech) – Modeled the distribution system in the proprietery Computer Aided Design (CAD) Software (RSCAD) for the Real Time Digital Simulator (RTDS) for both use cases and worked alongside me.
- Dr. Karen Butler-Purry – Provided the lab (WEB 050), the equipment, and the assistantship in the Fall of 2023.
- Dr. Goulart - Provided guidance through the entirety of the Master's program.
- Dr. Davis and Dr. Fink, my committee members, who provided good feedback on my thesis.
- ESET Program - Teaching assistantship provided through Spring 2024 and Fall 2024.

All other work conducted for the thesis (or) dissertation was completed by the student independently.

Funding Sources

This work was supported in part by the TAMU-PVAMU PRISE (Panther Research & Innovation for Scholarly Excellence) Grant Program.

NOMENCLATURE

AMI	Advanced Metering Infrastructure
ARP	Address Resolution Protocol
BESS	Battery Energy Storage System
CORE	Common Open Research Emulator
DDoS	Distributed Denial-of-Service
DER	Distributed Energy Resource
DMS	Distributed Management System
DNP3	Distributed Network Protocol 3
DoS	Denial-of-Service
DSO	Distributed Systems Operator
DSR	Dynamic Source Routing
EV	Electric Vehicle
ICS	Industrial Control Systems
IP	Internet Protocol
LTE	Long Term Evolution
MAC	Media Access Control
MiTM	Man-in-the-Middle
NAN	Neighborhood Area Network
NS3	Network Simulator 3
PV	Photovoltaic
ReaSE	Realistic Simulation Environment
RF	Radio Frequency

RSCAD	Real-time digital Simulation Computer Aided Design
RTDS	Real Time Digital Simulator
RTU	Remote Terminal Unit
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

The integration of communication technologies into power distribution systems leads to the development of smart distribution systems. Smart infrastructure plays a very important role in the automation of grid processes. They are equipped with advanced network devices and applications, making customers and vendors reliant on their well-functioning. These systems function with various components, incorporating distributed energy resources (DERs) into the grid. DERs are decentralized energy sources, such as solar panels and wind turbines, which can operate on different scales, from large wind and solar farms to residential homes with rooftop solar panels. Some DERs can also function independently of the local power grid. On a smaller scale, DERs are positioned by the customer on the secondary side of the distribution grid, where energy flows bi-directionally, with smart meters tracking both power generation and consumption. Data from these meters is transmitted to an Aggregator, who acts as an intermediary between the DERs and the distributed management system (DMS) or distributed system operator (DSO). The DSO and DMS manage the integration of DERs into the utility's distribution grid by adjusting the power supply from the DERs as needed.

With an interconnected distribution grid, communication can become a challenge and a security issue. These systems carry control messages and confidential information. The complex design and integration of different sensors with the communication infrastructure creates various access paths for a possible intruder to disrupt grid operations. As an example, ransomware cyber attack presented in [2] impacted the operations of large wind turbines owned by a NordEx SE company. The attack affected their corporate network; however, engineers shut down operation of several turbines until the communications network was restored. Another example was the deployment of Industroyer malware [3] on the Ukrainian grid. The malware attacks directly circuit breakers and switchboards by using industry standards Industrial Control Systems (ICS) protocols, such as IEC 60870-5-101 and 104, as well as IEC 61850.

The first use case is to replicate the peak shaving experiment in [4], that specifies how the Real

Time Digital Simulator (RTDS) and Network Simulator 3 (NS-3) were used to run a peak shaving application, with a few differences:

- Instead of using NS-3, the communication network is modelled using Common Open Research Emulator (CORE) [5]. It offers real-time communication with RTDS. This thesis presents details on how we implemented this real-time cyber-physical testbed.
- The communication between RTDS and CORE uses a reliable transport layer protocol – Transmission Control Protocol (TCP) – contrary to the case presented in [4] that uses User Datagram Protocol (UDP). This is more realistic, as TCP is used in the majority of Internet applications and several ICS protocols uses the TCP/IP model.
- The use case presented shows a two-stage intrusion in this network, in which an adversary first finds the software applications that are vulnerable to an attack, then it performs a denial of service (DoS) attack to impact the availability of the Aggregator node in CORE and a man-in-the-middle (MITM) attack to impact the confidentiality and integrity of the data exchanged between aggregator and the DERs.

The techniques and information used on the first use case are applied into a more realistic model simulating an Advanced Metering System (AMI) using Radio Frequency (RF) for communication. AMI technology uses automated meters that periodically read electric usage of a customer in real time through automatic meter reading (AMR) [6] [7]. AMR helps the utility understand high demand times and adjust price accordingly. The automation and integration of smart meters allow the utility to keep track of historical data and balance the electrical flow on the grid by observing generation and distribution data. The aspect of being real-time data flowing through a communication channel brings a lot of challenges to the network side. For the data to readily accessible at the utility headquarters, smart meters and collectors employ an RF mesh network over Internet Protocol (IP).

The second use case is based on SmartDS data set containing distribution network models connecting low voltage customer loads all the way to distribution substations [8]. The model con-

structed is based on the Austin, TX dataset, and uses a specific feeder to model the physical system and the cyber network to accurately describe the area where the feeder is located. The communication network implements real distances between smart meters, collector, and substation. Also have the links bandwidth adjusted to create a realistic and dynamic testbed containing real and accurate delays and packet loss. The testbed will use wireless mesh communication method between smart meters and the collector, and assumes fiber cable connections between collector and utility.

2. LITERATURE REVIEW

2.1 Cyber-Physical Testbeds

Previous studies addressed cybersecurity threats in smart distribution networks. The authors in [4] implemented a cyber-physical testbed that includes a remote terminal unit (RTU), DERs, a DSO and an Aggregator. During a peak shaving application, in which the DSO, Aggregators and DERs exchange data to respond to an increase in load, an intruder performs malicious attacks. They show the impact of these attacks on the distribution system. An important contribution of their work is a detailed description of their cyber-physical testbed, which used the Real-Time Digital Simulator (RTDS) power simulator NS-3 network simulator. This testbed was also documented in a technical report [9].

There have been multiple studies and testbeds developed to aid researchers and professionals to better understand and analyze the impact of different cyber attacks on the power grid. The authors in [10] go into detail on how they used different simulators, such as GridLAB-D to simulate the power grid infrastructure and NS-3 network simulator for the communications portion, to create a realistic model of the power grid system leveraging a Distributed Network Protocol 3 (DNP3), an Industrial Control System (ICS) protocol used by more than 75% of North American utilities in industrial control applications [11]. Their scenario consists of a man-in-the-middle attack where the attacker uses a DNP3 application within NS3 to spread misinformation, or when part of a larger scale attack to take control over a section of the communications network. The authors show the results on different inverters and how their values fluctuate between a real a fake value once the attacker injects false data into the communication.

A proposed dynamic hardware-in-the-loop cyber-physical testbed using the RTDS and QualNet is presented in [12], where the authors investigate a dynamic approach to a Denial-of-Service attack. The dynamic approach has the objective of not using script simulations, so researchers can estimate and analyze a more realistic impact on the power-grid system. The authors use different

hardware and software technologies to create their proposed testbed. The hardware-in-the-loop includes a Protocol Conversion Module (PCM), a piece of hardware used to convert non-standard packets into standard layer 3 IP packets. The proposed DoS attack scenario shows a loss of data where the control measurement packets cannot continuous flow, failing a high requirement of substations to receive several control measurement packets before it can take a course of action.

The authors in [13] developed a hybrid hardware-in-the-loop cyber-physical testbed using OPAL-RT Real-Time Simulator, Photovoltaic (PV) Emulators to simulate the behavior of a PV panel with different environmental conditions adjusted in its software, and hybrid DC-AC inverters to convert the DC power generated by the DERs into AC power for the grid. They ran two cyberattacks, a Distributed Denial-of-Service (DDoS) in the DSO node using EXata network emulator, that overloaded the DSO's CPU and impacted the communication with the DER systems. The second case was demonstrated with a replay attack, where the adversary sent a command to turn off the generation of the PV inverter for 30 seconds. This experiment showed that it took nearly five minutes for the power system to reach normal generation values.

Similarly, the impact analysis framework shown in [14] uses the RTDS simulator to simulate the power system model, and uses a network simulator to carry out the cyberattacks, in their case OPNET simulation. The use case was to perform a Denial of Service attack and a Man-in-the-Middle attack and analyze the data sent between the RTDS and OPNET. The authors showed the delay created in a trip command to a breaker and its impact on the physical system.

2.2 AMI System

Another aspect of smart distribution systems is the integration of smart meters into the communication network, the authors in [6] details how end-to-end communication regarding utilities in smart distribution systems happen. They provide advantages and disadvantages for each communication type, from wireless (radio waves, microwaves, cellular) to fiber optics. They also highlight how different grids and locations might affect the choice of communication used, and that radio frequency mesh is the most suitable option for smart meters and substation communication.

Large number of utilities and AMI systems use wireless mesh network configuration for com-

munication between smart meters and collectors. The authors in [15] detail the communication architecture of RF-mesh systems and discuss the importance of network performance analysis on these networks. A detailed description of delay analysis is also proposed, providing critical information about delays in respect to the network traffic which can be used to detect and mitigate cyber threats in wireless mesh systems.

The authors in [16] present a sophisticated version of a denial-of-service, the Puppet attack, that targets Dynamic Source Routing (DSR), a protocol used in wireless mesh networks. DSR uses request packets (RREQ) and route reply packet (RREP) to find various routes among all nodes in the mesh. For example, node A has path ABCDEF to final destination F, but also discovers an alternate path ABCEF. By setting up a malicious node with a fake route ABCDEX the attacker takes advantage on the structure of the protocol. Nodes in between the path will start sending RREQ packets to find routes to node X, and so will their neighbors, starting a flood of RREQ packets and affecting the node links. The authors show that the packet delivery rate drastically decreases with the Puppet attack.

Cybersecurity issues increased with the integration of smart systems into distribution networks. An impact analysis is performed in [17] where the authors execute a Denial-of-Service (DoS) and a Distributed Denial-of-Service (DDoS) in an AMI realistic grid topology. The authors utilized OMNeT++ simulation environment and INET framework to simulate smart meters, routers, and a utility node. Realistic network traffic and malicious traffic were generated using Realistic Simulation Environments (ReaSE). The authors displays the results of both attacks where 89.7% of smart meters are unable to receive any packets form the utility server.

Similarly, the authors in [18] introduce Long Term Evolution (LTE) cellular network vulnerabilities through different variations of Denial-of-Service attacks to legitimate subscribers in an AMI system. The authors organize the different attacks with explanations and countermeasures and assemble a final table with their findings. Attacks such as Signaling DoS, where the attacker impedes legitimate users of establishing connections with the LTE network by causing overload of the system, and jamming attacks, where the attacker radiates electromagnetic energy with the

intention of reducing the reliability of the communication channel, are used to describe the vulnerabilities in AMI systems using wireless connections for meter communication.

2.3 Common Open Research Emulator: CORE

The motivation to use CORE network emulator is the scalability that it offers and that it is open source. CORE was also used in another cyber-physical testbed - the Resilient Energy System Lab (RESLab) testbed - with PowerWorld power simulator [19]. CORE allows the user to create virtual networks in a Linux environment using container technology. Containers are standalone units of software that have application code and all dependencies. Each container works isolated from the main Operating System (OS) or other containers. Fig. 2.1 depicts the container hierarchy in respect to the entire computer.

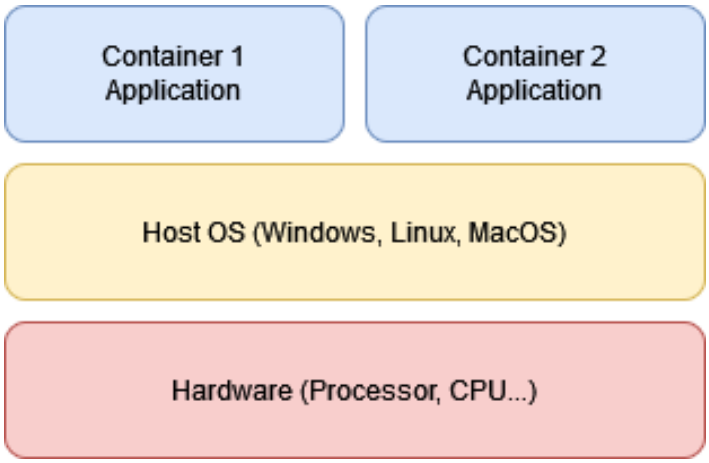

Figure 2.1: Container architecture.

CORE allows us to create our own applications using socket programming in Python. These applications represent nodes in a smart distribution system, such as an RTU, an Aggregator, or a DSO. CORE runs in real time, similarly to the RTDS power simulator. The network emulation can be integrated with external devices through the use of a TAP interface, which creates a data link between two devices. This interface allows us to connect CORE to physical devices, such as a

computer or the RTDS power simulator. The use of containers and RTDS has also been described in [20].

3. OBJECTIVES AND SCOPE

The development of cyber-physical testbeds is essential for simulating cyber attacks on power distribution systems without affecting the integrity and the availability of real-world systems. These testbeds can replicate the operational environment of a distribution grid, integrating both physical components and cyber elements, such as the Real Time Digital Simulator (RTDS) to simulate the physical portion of a power distribution grid, and a network emulator to serve as the communication system of the distribution power network. By creating a controlled and realistic environment, we can conduct various types of analyses of potential cyber threats and attack vectors for a more resilient grid. The scope of this work includes designing and implementing a testbed infrastructure, configuring it to mirror operational characteristics of real-world distribution grids and its applications, and ensuring it supports various attack scenarios for a full evaluation of their impact on these real systems.

The primary objective of establishing a cyber-physical testbed is to enhance the resilience and security of power distribution systems by enabling risk-free testing of cyber attack scenarios. The key objectives include:

- To replicate and test different cyber attack scenarios and their impact on a power distribution grid, including potential disruptions and system responses without affecting any real infrastructure.
- To provide a practical training environment for cybersecurity researchers, enhancing their ability to recognize, respond to, and manage cyber threats in a controlled setting.
- To facilitate the development and testing of new defensive technologies, strategies, and protocols in a realistic environment, ensuring they are robust before deployment in real-world systems.
- To advance research in the field of cyber-physical systems by exploring new attack tech-

niques, defensive mechanisms, and system designs that can improve the security and reliability of power distribution grids.

By achieving these objectives, our cyber-physical testbed can strengthen the cybersecurity of power distribution systems, ensuring their operational resilience against existing and new cyber threats.

4. CYBER-PHYSICAL TESTBED ARCHITECTURE

4.1 Hardware Setup

A diagram displaying the setup of the testbed is shown in Fig 4.1 and a figure of the laboratory setup is displayed Fig. 4.2. The proposed testbed contains five physical parts:

- RTDS Hardware: Where all the power system simulation occurs.
- GTNETx2 Board: Interfaces the RTDS simulation with external devices through two Ethernet ports.
- Ethernet Switch: Connects all the physical equipment to the same Local-Area-Network (LAN).
- Computer 1: Windows 11 machine that contains the RSCAD software used to build and model the power system.
- Computer 2: Linux machine with Ubuntu 20.04 used to install CORE and emulate the cyber network portion of the testbed.

Next I provide more details of the communication system configuration to allow packet to flow from RTDS to the CORE network and vice-versa.

4.2 Communication Network

The network portion of the testbed is built in CORE. Every system modeled in RSCAD (DER, DSO, Aggregators, smart meters, and collectors) is represented by a CORE container in the cyber network, each one with its own IP address working independently from each other. To allow the communication to be extended outside of the CORE network, a TAP interface was implemented. Represented by the RJ45 icon available in CORE, the TAP interface bridges the communication from CORE's network to the real physical network in the lab. Ethernet 2 port in the Dell computer running Ubuntu was configured to be the TAP interface. Fig 4.3 shows a visual representation how

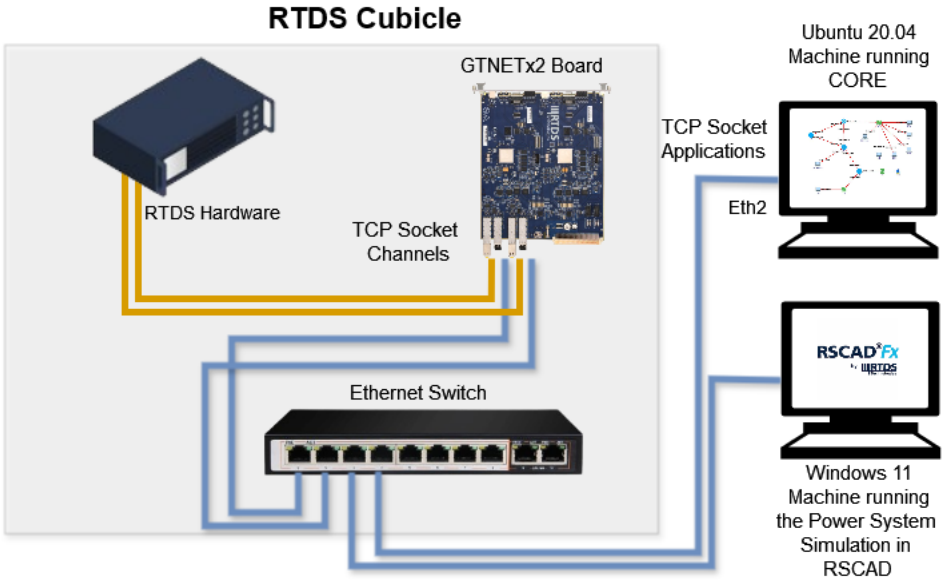

Figure 4.1: Testbed diagram.

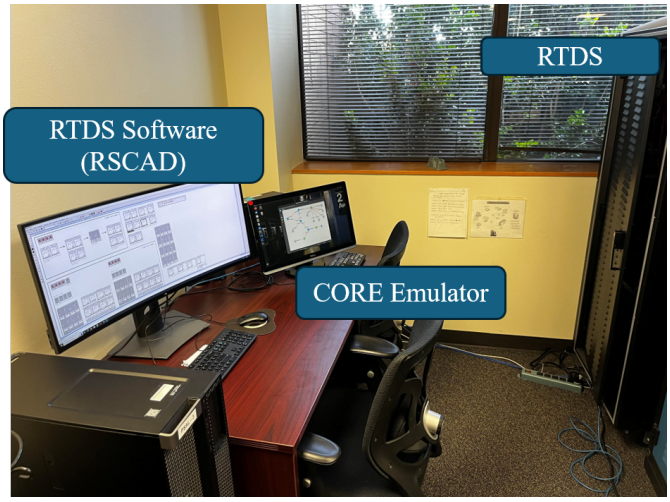

Figure 4.2: Testbed computers and RTDS in the lab.

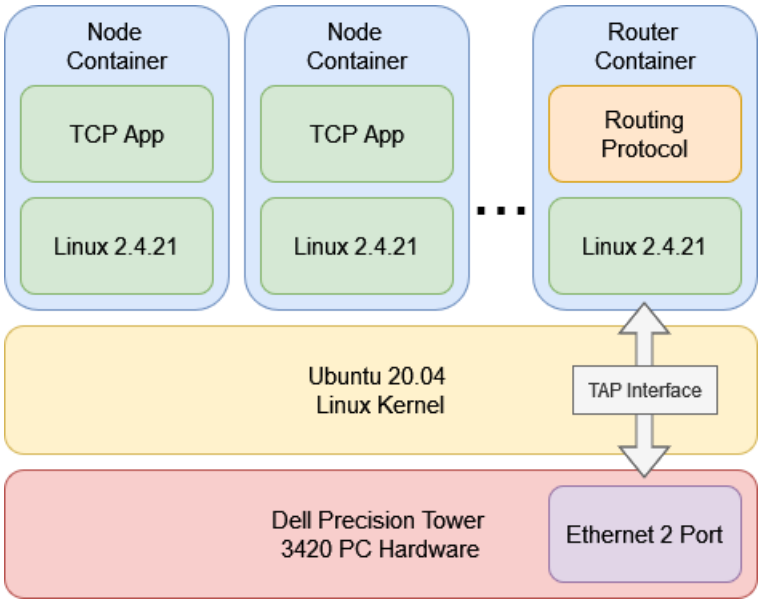

Figure 4.3: CORE uses containers to emulate hosts and routers.

the TAP Interface is able to connect CORE to the outside network and how CORE containers are layered in the Ubuntu machine.

To facilitate the communication from CORE to outside devices and ensure correct routing, the interface in the router container connected to the TAP interface must assume an IP address in the physical outside network. In both use cases this IP address was 10.125.184.186. This IP must be the default gateway configured in the RTDS. This configuration allows the RTDS to reach every node in CORE. The router directly connected to the TAP interface also needed a change to its routing table to ensure the lab's LAN, 10.125.184.0/23, is reachable via 10.125.184.186, as shown in the first row of the routing table presented in Fig. 4.4.

The emulated nodes communicate with RTDS through the GTNETx2 board. The board offers real-time communication to run the test cases with the capability of implementing applications with different ICS communication protocols. GTNETx2 board is equipped with two Ethernet modules, both modules are configured to use the GTNET-SKT network protocol, which communicate to other equipment in a Local or Wide Area Network using TCP or UDP sockets, exchanging data points through IEEE 754 floating-point standard [21].

```
root@R4:/tmp/pycore.1/R4.conf# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.125.184.0    0.0.0.0         255.255.254.0   U     0      0        0 eth0
172.24.9.0      172.24.9.53     255.255.255.240 UG    3      0        0 eth2
172.24.9.16     0.0.0.0         255.255.255.240 U     0      0        0 eth1
172.24.9.32     172.24.9.53     255.255.255.248 UG    2      0        0 eth2
172.24.9.40     172.24.9.53     255.255.255.252 UG    3      0        0 eth2
172.24.9.44     172.24.9.53     255.255.255.252 UG    2      0        0 eth2
172.24.9.48     172.24.9.53     255.255.255.252 UG    2      0        0 eth2
172.24.9.52     0.0.0.0         255.255.255.252 U     0      0        0 eth2
```

Figure 4.4: Routing table for a router connected to the TAP interface.

Each host in the CORE network uses TCP/IP sockets to communicate with RTDS machine. A socket is a communication channel between two applications. For both use cases, we used TCP port number 15,001 for all data transfers within CORE and port 7,001 for data transfer between CORE nodes and the RTDS. The configuration of all communication channels are done through TCP client and server applications, where each node in CORE connects to a TCP server, attached to a specific channel in the RTDS configuration, corresponding to its counterpart node. For example, a smart meter in CORE connects to a TCP server in the RTDS through the GTNETx2 board attached to the smart meter with an IP address inside 10.125.184.0/23. All TCP client/server applications in CORE are written using Python programming language. The Python libraries to implement TCP sockets was *Sockets* and the library used to encode and decode IEEE 754 floating point data was *struct*.

5. USE CASE 1 - PEAK-SHAVING APPLICATION

The first use case presented has the goal of analyzing the impact on a smart distribution system under three different cyber-attacks. The cyber network was modeled according to the power system that is configured to perform a peak shaving application.

5.1 Peak Shaving Application

The authors in [22] summarize peak shaving with five different involved agents, each serving a particular purpose on the smart distribution system and actively handling important data. The authors also breakdown the application into nine different steps that are used in this use case. These steps are numbered and are used to show which steps are affected during the cyberattacks.

1. The SCADA system reads the load measurement data from distribution transformer through the RTU and sends this data to the DMS.
2. The DMS calculates power flow estimates for all grid systems. If any of them has a load above their limit, the DMS calculates the inverted difference as a reference signal.
3. The DMS sends a reference signal to the aggregator that is providing the service to different DERs. The case study has only one aggregator.
4. The aggregator requests flexibility information from all DER units in its portfolio.
5. Each DER unit responds with a flexibility prognosis with its maximum flexibility.
6. The aggregator performs an internal optimization to be able to deliver the service in a cheap and optimal way.
7. The aggregator sends set-points to all DER units and requests flexibility updates.
8. The DER units respond with an updated flexibility prognosis.
9. DER smart meters provide current measurements to the DSO.

5.2 IP and Channel Configuration

All the acting agents described in the peak shaving application are modeled into the cyber network. DERs (1-4), the RTU, DSO/DMS, and the Aggregator have nodes in the CORE network, they all assume IP addresses in the 172.24.9.0/24 network and are configured according to Fig. 5.1. Each of these nodes are connected to its counterpart node in the power system simulation through different channels, separated into *Multi1* and *Multi2* differentiated by configuration groups in RSCAD. Tables 5.1 and 5.2 show the configuration for each channel, what node they are attached to, as well as the two IP addresses used for the communication in each channel. All of the channels in the RTDS are configured as TCP servers and the hosts in CORE are TCP clients. The channels also are specified if the RTDS sends data, receives data, or both. The aggregator has five channels, one for each DER. For example, Multi 2 channel 5 sends and receives data from or to DER 1. The last channels are for the Aggregator node itself.

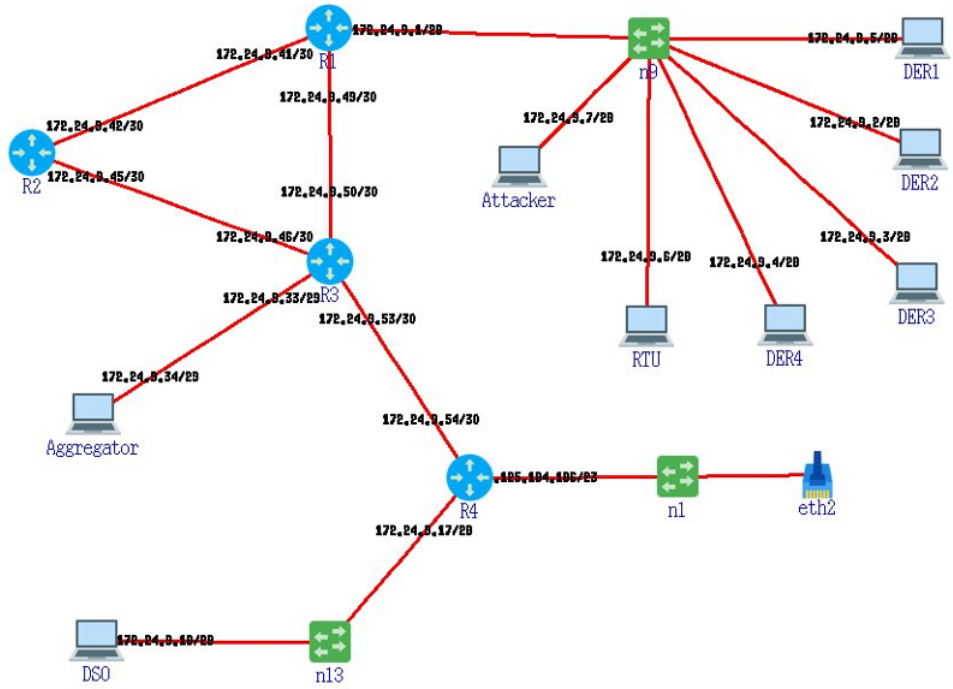

Figure 5.1: Use case 1 communication network emulated using CORE.

Table 5.1: Multi 1 Channel Configuration - Use Case 1

Channel	Node	IP Address RTDS	IP Address CORE	Function
1	RTU	10.125.184.183	172.24.0.6	Send
2	DSO	10.125.184.180	172.24.0.18	Send and Receive

Table 5.2: Multi 2 Channel Configuration - Use Case 1

Channel	Node	IP Address RTDS	IP Address CORE	Function
1	DER 1	10.125.184.179	172.24.0.5	Send and Receive
2	DER 2	10.125.184.171	172.24.0.2	Send and Receive
3	DER 3	10.125.184.172	172.24.0.3	Send and Receive
4	DER 4	10.125.184.173	172.24.0.4	Send and Receive
5	Aggregator	10.125.184.174	172.24.0.34	Send and Receive
6	Aggregator	10.125.184.175	172.24.0.34	Send and Receive
7	Aggregator	10.125.184.176	172.24.0.34	Send and Receive
8	Aggregator	10.125.184.177	172.24.0.34	Send and Receive
9	Aggregator	10.125.184.178	172.24.0.34	Receive

5.3 Cyber Threat Use Cases

The testbed was used to run three different use cases. Each one of them was a different cyber attack performed in CORE that affected the power system in the RTDS. The first use case was a reconnaissance attack that was used to find vital information regarding the communication system emulated in CORE. The second use case was a DoS attack where the aggregator node was targeted to affect its availability. The last use case was a man-in-the-middle attack, where the adversary is able to affect the integrity of the data exchanged between the DERs and the Aggregator.

5.3.1 Reconnaissance

In the context of cybersecurity and information technology, reconnaissance attack is an initial and necessary step in the attempt to find vulnerabilities in a communication network. Malicious actors may use different digital applications and social engineering techniques in this stage of the

attack life-cycle shown in [23]. The primary goal of reconnaissance is to gather information rather than exploiting or causing harm. It identifies vulnerabilities, weaknesses, and points-of-entry in the targeted network or machine. The attackers may use various applications that employ different techniques such as port scanning, network enumeration, and packet sniffing.

This use case presented in this paper focuses on the use of a software tool called *nmap*, an open-source network mapper application. To gather information regarding the emulated CORE network through port scanning and service detection, the adversary node in CORE runs the command *nmap -oN nmap-results.out -A -T4 -p 0-20000 172.24.9.34*, and discovers a server running on the Aggregator node using port 15001. The options used with the command to find information regarding the aggregator node are explained below:

- -A: Intense scan. Performs port scanning, service detection, OS detection, and performs a *traceroute* command to find the path to the target.
- -oN: Outputs the scan in text format.
- -T4: Sets a timing template; it can be used with values between 0-5.
- -p: Scans the ports, which in this case is a range of port numbers from 0 to 20,000.

5.3.2 Denial of Service

A Denial of Service (DoS) attack, in simple terms, is the exhaustion of network and computational resources by flooding the target's machine with network traffic. There are multiple variations of this attack. In this use case, we present a TCP SYN flood attack. On a TCP connection, the client (adversary) sends a SYN packet, which represents the intention of connection. The server (target) responds and allocates memory and processing power to this new connection. In theory, the client would respond with the last acknowledgement, which is the last step of a TCP three-way handshake. However, when running a DoS attack, the client never sends the last message, which leads the server to keep these resources allocated for the intended connection. This process is repeated thousands of times for every SYN packet that is received. This eventually slows down the machine an shuts down the TCP application.

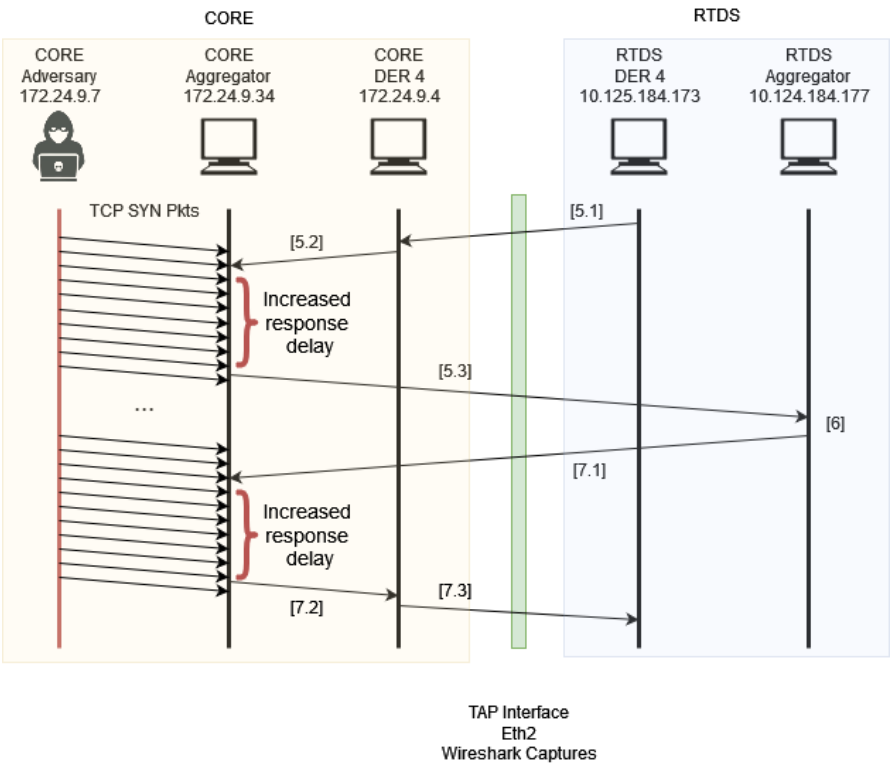

Figure 5.2: Timing diagram of data flow between CORE and RTDS during the DoS attack.

The attack was all done through the CORE containers. The adversary node in CORE uses *hping3*, similarly to the authors in [24], to target the Aggregator node (172.24.9.34) during a peak shaving application. The timing diagram in Fig. 5.2 depicts the data-flow of the application and where the DoS attack was implemented. The diagram uses DER 4 as an example and bases the steps from the communication steps found in [22] for the peak shaving application.

- Step 5.1: Each DER in RTDS forwards flexibility prognosis, or its maximum capacity to the DER container in CORE.
- Step 5.2 DER container forwards the prognosis to the aggregator container.
- Step 5.3 Aggregator container sends the data received by all DERs to the Aggregator channel in the RTDS.
- Step 6 Aggregator in the RTDS performs calculations in order to deliver the service needed

in the cheapest and most effective way and forwards the DER set-points to the aggregator container.

- Step 7.1 Aggregator in the RTDS sends the calculated set-points and the updated flexibility request to Aggregator container in CORE.
- Step 7.2 Aggregator container in CORE forwards the set-points and the request to each DER node in CORE.
- Step 7.3 Each DER container sends the set-points and update flexibility request to each DER node in the RTDS.

With the attack the aggregator node in CORE is unable to process the data coming from the DERs and the aggregator node in the RTDS, which creates additional delays between steps 5.2 and 5.3 as well as steps 7.1 and 7.2

The cyberattack is executed with a single command: *hping3 -S -p 15001 -d 90 –flood –rand-source 172.24.9.34*. All different arguments in the command are specified to generate the desired communication with the target, as follows:

- -S: The machine's IP address to send TCP SYN packets.
- -p: The targeted port number; in this case 15001.
- -d: Size of packet: 90 bytes + 40 bytes of default headers.
- –flood: Tells Hping3 to flood the targeted IP.
- –rand-source: Randomize the source IP address.

5.3.3 Man-in-the-Middle

A Man-in-the-Middle (MiTM) attack is a cybersecurity threat in which an attacker intercepts, monitors, and alters the communication between two parties without their knowledge. This type

Figure 5.3: Effective vs Actual link during MITM Attack Use Case 1.

of attack typically occurs on a local network, meaning all participants are part of the same subnetwork. By exploiting vulnerabilities of network configurations, such as through Address Resolution Protocol (ARP) spoofing, the attacker can divert the data flow through their own system. Once in position, the attacker can eavesdrop on sensitive information, such as login credentials or personal data, manipulate the transmitted data to alter its content, or inject malicious payloads into the communication.

This attack, similarly to the DoS attack, was all done through CORE, where the adversary uses Scapy, a Python tool used to create and manipulate network packets that can be used in a Python script or on a command line interface. The adversary places himself in between the DERs and the gateway by sending ARP packets to the gateway of the subnet (172.24.9.1) and all of the DERs, thus updating the ARP table of each device. Fig. 5.3 shows the effective links while the attack is being executed. ARP protocol maps the IP addresses of the devices in a local area network to their Media Access Control (MAC) addresses. This technique allows the Adversary to receive all intended traffic to and from the DERs. Fig. 5.4 depicts a timing diagram of the attack with the relevant steps affected in the peak shaving application with DER 4 used as example.

The attack is executed in a series of events, where several tools were used to make it successful. Below is a detailed list on how the adversary successfully executed the attack:

- The adversary container had to be configured to allow the automatic forwarding of packets received during the attack by running *sysctl net.ipv4.conf.eth0.send_redirects=0* and *sysctl net.ipv4.ip_forward=1* commands.

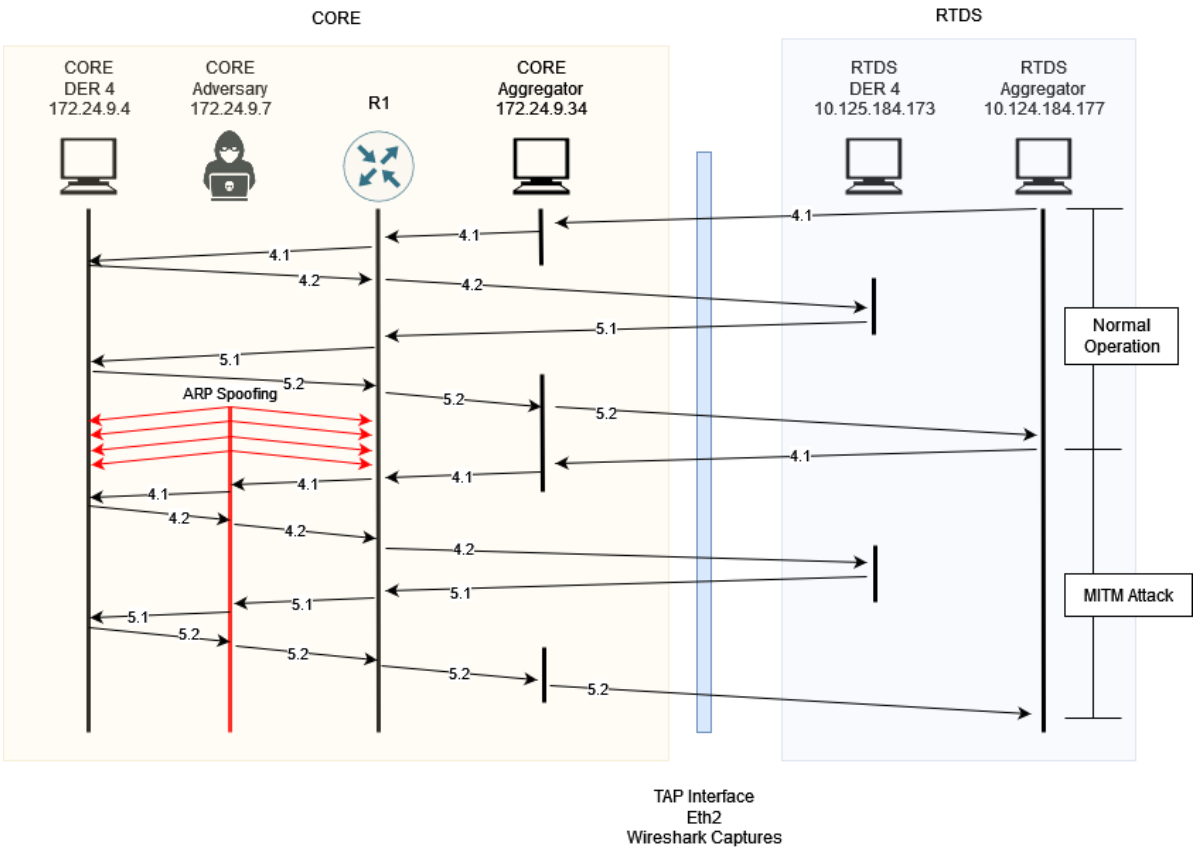

Figure 5.4: Timing diagram of data flow between CORE and RTDS during the MiTM attack.

- Using Scapy command line tool, the adversary begins the ARP spoofing by running the *arp_mitm("172.24.9.1", "172.24.9.X")* command. Where the first IP address is the default gateway's IP address, and the second is the IP of each DER. This command is executed four times, once for each DER. (At this stage the adversary is already in between the communication of the Aggregator and the DERs, by running a simple Python program the attacker can eavesdrop on all packets intended for each party.)

- To inject malicious TCP payloads into the packets and alter the communication between the DERs and the Aggregator, a Linux kernel queue feature called Netfilterqueue (NFQueue) is used to place the packets before sending them out to their final destination. NFQueue is a packet filter/firewall that offers tools for networking related tasks, such as packet inspection and modification [25]. To enable packets to be sent to the queue prior to be redirected, the

following rule must be configured *iptables -o eth0 -t filter -A FORWARD -p tcp –tcp-flags PSH,ACK PSH,ACK -j NFQUEUE –queue-num 1 –queue-bypass*. All the arguments used in the command are as follows:

- -o: Only packets on interface *eth0*.
- -A: Rule added only for forwarded packets.
- -p: Select protocol for rule to be applied to be TCP.
- –tcp-flags: Configures rule to be applied only to TCP packets with flags PSH/ACK.
- -j: Set the target of the packet to queue 1.
- –queue-bypass: Do not send packets to queue if the application has not started yet (avoids race condition).

A Python algorithm using the Scapy module and the NFQueue Application Programming Interface (API) is then used to access the packets placed in the queue and modify their data to show that only DER 1 has flexibility, therefore being paid unfairly compared to the remaining DERs.

5.4 Results

This section compiles the results for the three cyberattacks explained in the previous section. Their data sets were acquired through Wireshark and a data acquisition tool from the RTDS. This section provides insights on the impact of the cyberattacks on the communication side of the testbed, as well as patterns found in the malicious data.

5.4.1 Reconnaissance

The reconnaissance attack showed to the adversary that the Aggregator's server uses port 15001 to communicate with the DERs and DSO, as seen in Fig. 5.5. Other information, such as OS version and route information were collected. This helps the adversary understand the network topology and services running during the peak shaving application. This information can later be used in a more elaborated attack.

5.4.2 Denial of Service

With detailed information regarding the network, the adversary now is able to impact the availability of the server on the Aggregator node by slowing down the data exchange the Aggregator has with the DSO and the DERs. The impact on the distribution transformer's load and reactive power can be seen in Figures 5.6 and 5.7. During normal operation, the total runtime of the peak shaving application in RTDS is roughly 6 seconds. However, when the DoS attack begins, the duration of the peak shaving application increases to 30 seconds.

The data packets captured using Wireshark packet sniffing tool are analyzed in Fig. 5.8. In green, it shows traffic within CORE, and between CORE and RTDS. In red, it shows the TCP SYN flood traffic generated by the adversary. There are over one million packets sent in the first wave of traffic. The packet sizes generated during this wave are bigger than normal packets. After this initial flood, the graph shows a big halt of packet flow, with no traffic for about 40 seconds between 100 and 150 seconds. This corresponds to the increased delays depicted in the time diagram in Fig. 5.2.

5.4.3 Man-in-the-Middle

A comparative analysis is done to observe the impact of the MITM attack on the communication network by showing results from normal and affected operations. Two different sets of data

```
Nmap scan report for 172.24.9.34
Host is up (0.00013s latency).
Not shown: 20000 closed ports
PORT      STATE SERVICE    VERSION
15001/tcp open  tcpwrapped
Device type: general purpose
Running: Linux 2.4.X
OS CPE: cpe:/o:linux:linux_kernel:2.4.21
OS details: Linux 2.4.21
Network Distance: 3 hops

TRACEROUTE (using port 143/tcp)
HOP RTT     ADDRESS
1   0.05 ms 172.24.9.1
2   0.07 ms 172.24.9.50
3   0.15 ms 172.24.9.34
```

Figure 5.5: NMAP results displaying port 15001 open on the Aggregator node.

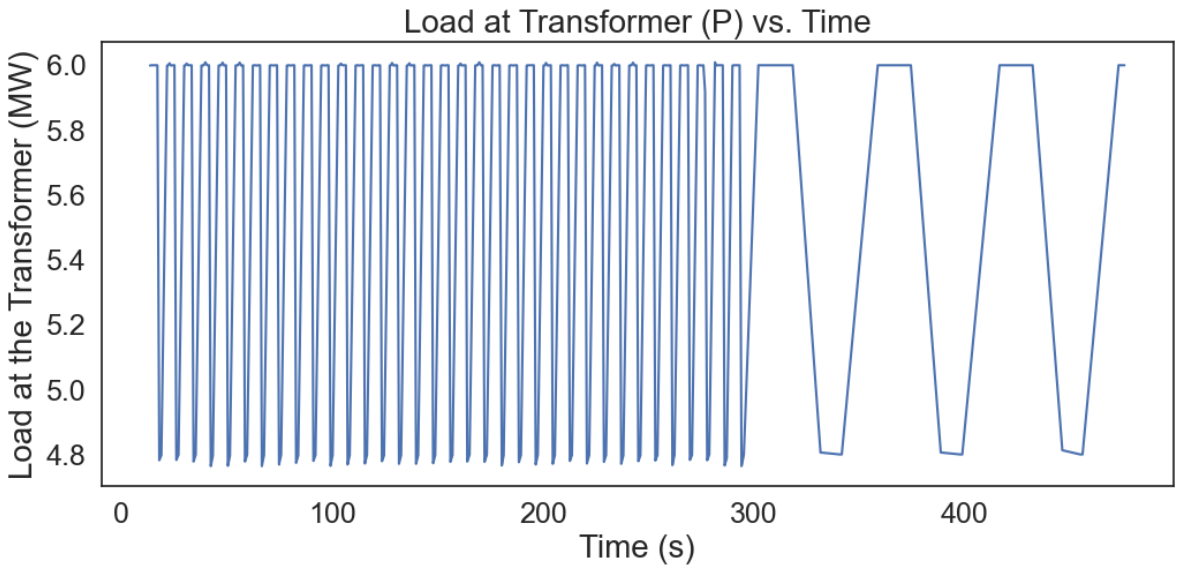

Figure 5.6: DoS Attack - Impact on the transformer's load in MWatts.

were captured using Wireshark. Fig. 5.9 shows the observed round-trip times (RTT) from when the flexibility prognosis is sent to the aggregator in step 5.2 with a *PUSH/ACK* TCP flag to when a acknowledgement *ACK* TCP flag is received by DER during normal operation of the peak shaving and with MITM attack. The normal RTT ranges from the lowest data point, around 0.02 ms, to the highest data point, a little over 0.06 ms. The MITM attack displays the delay times during the attack increased to approximately 0.5 ms to a maximum of 2.25 ms. The impact of the delays can be observed side by side on a box plot showing the discrepancies of the two datasets in Fig. 5.10. The normal operation data set has a median delay of 0.000055532 seconds, or 55.532 μs and the MITM data set has a median delay of 0.001453 seconds, or 1.453 ms.

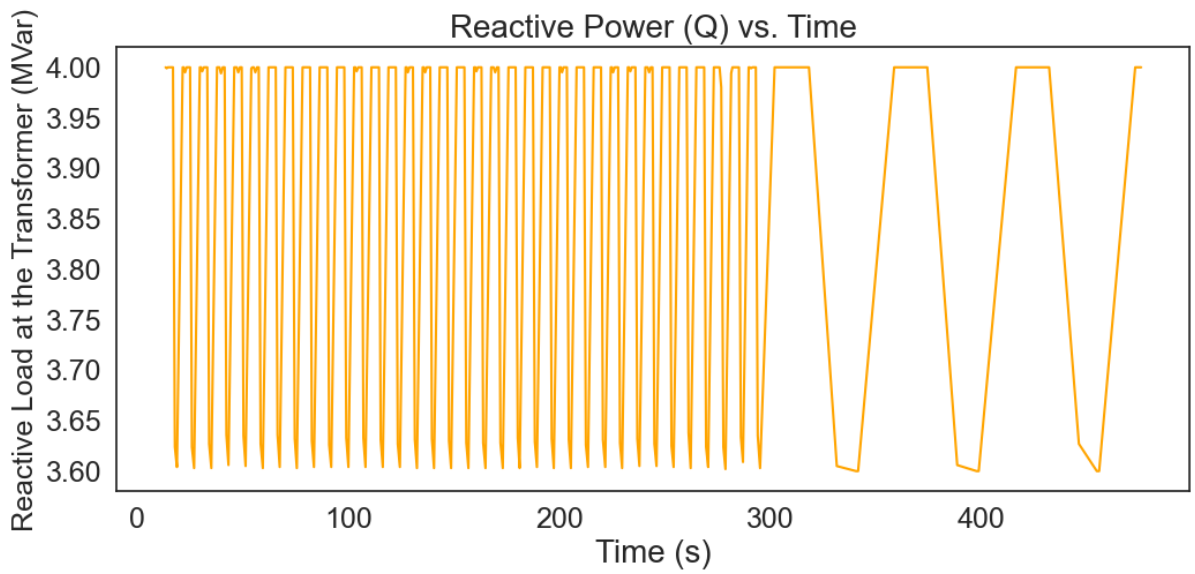

Figure 5.7: DoS Attack - Impact on the transformer's reactive power in MVar.

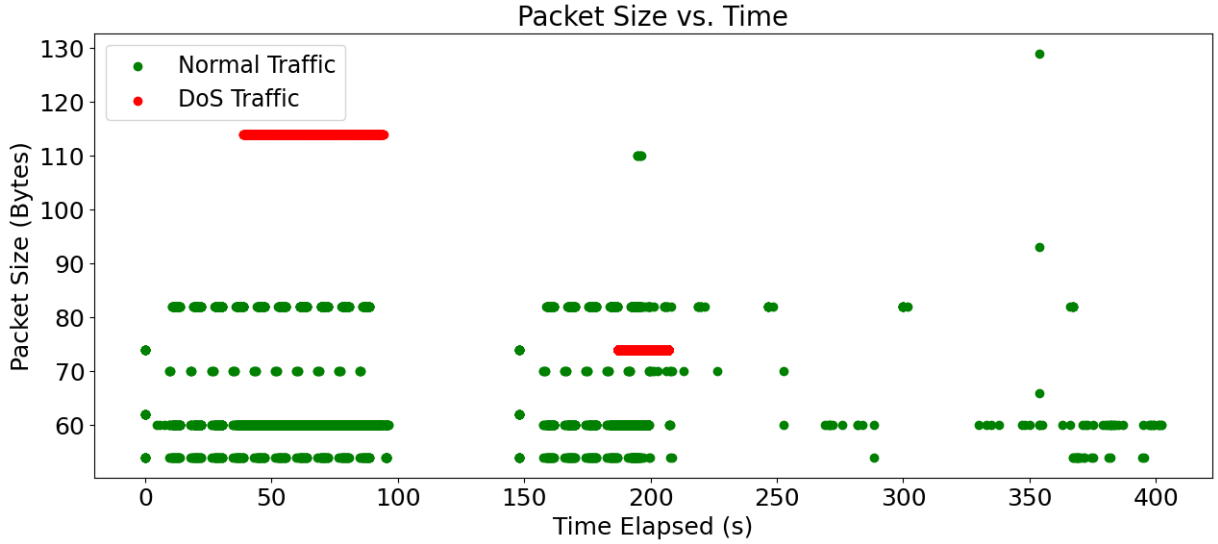

Figure 5.8: Data packet sizes observed during the experiment.

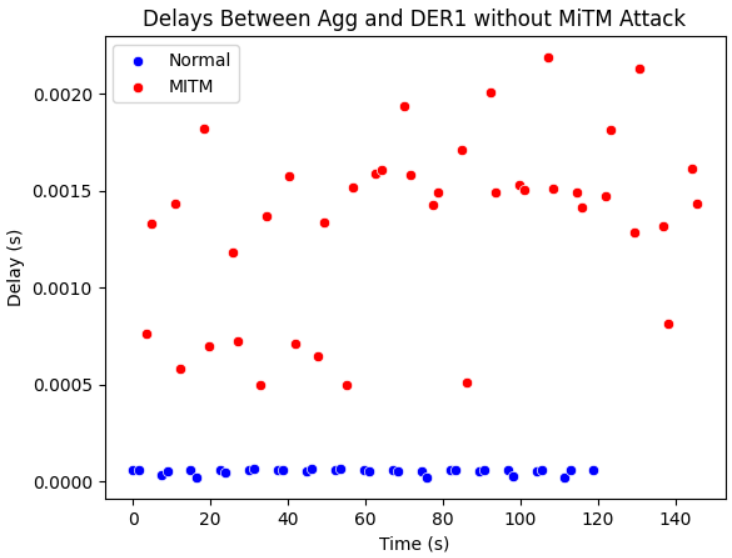

Figure 5.9: RTT for normal operation and while running MITM attack.

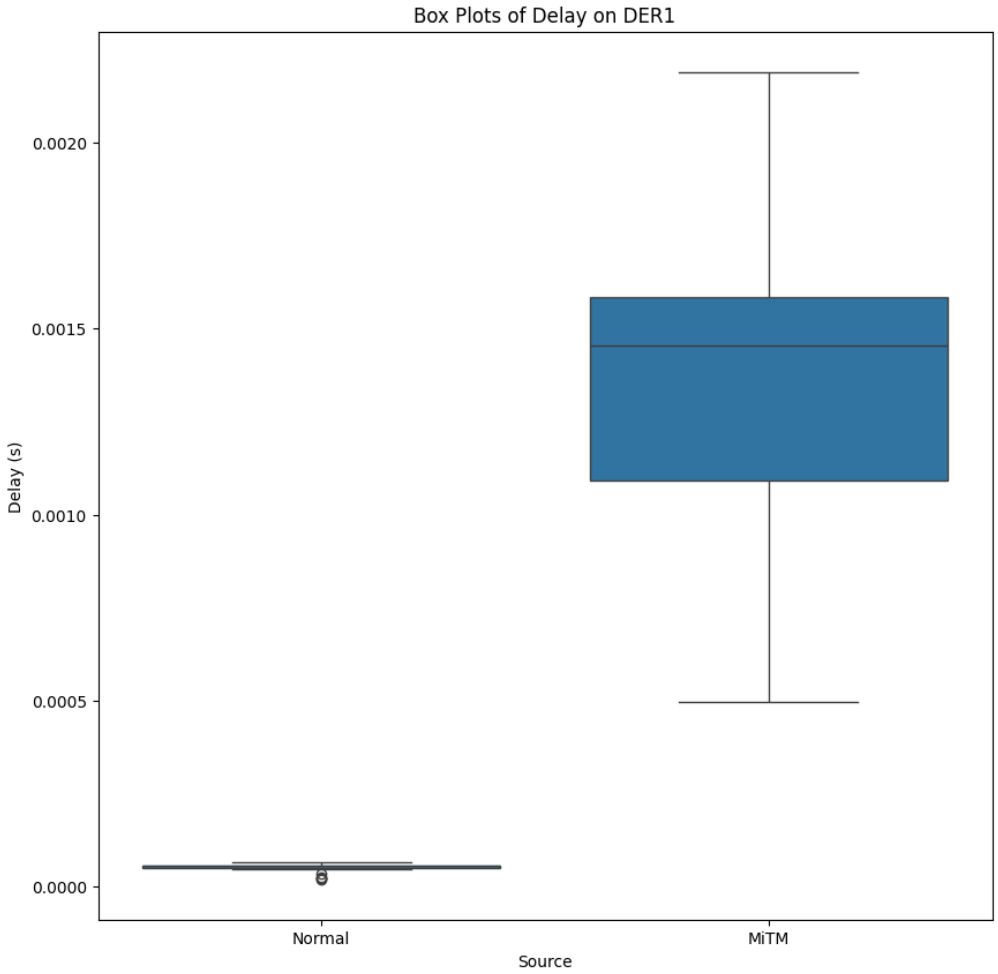

Figure 5.10: Boxplot for RTT with and without MiTM.

6. USE CASE 2 - AMI MESH NETWORK

Feeder P1UDT18962 was chosen from the Smart-DS dataset and simulated using CYME software [26] by another student (Kumpanat Thongmai). The feeder is separated in 7 zones displayed in Fig. 6.1 This model was simplified to reduce the number of nodes in the RTDS simulation. Zone 7 was chosen to be replicated in CORE, with residential and commercial loads containing models for Electric Vehicle (EV) charger loads, Photovoltaic (PV) loads, and Battery Energy Storage System (BESS) loads. The placement in the map for each meter, routers, and collector are shown in Fig. 6.2. Smart meter communication in the AMI system is based on an RF mesh metering system [1].

This system utilizes a layered architecture, with smart meters interconnected at the base layer, forming a Neighborhood Area Network (NAN). An intermediate layer of routers connects to the meter and act as repeaters, directing network traffic to the collector node. The collector node is connected to a wide area network layer, where traffic is sent from the remote collector to the utility's head-end system. The diagram in Fig. 6.3 shows how smart meters, collectors, and utility are connected in an RF mesh network.

The CORE network models a realistic real-time network that incorporate delays that represent the geographical distances between various systems within the feeder model. Bandwidth is adjusted to real values, explained in the next section, to emulate the real link as well as signal loss due to the RF nature of the system. Finally, studies were conducted in the testbed to evaluate how cyberattacks affect the cyber-physical system.

In June 2024, our research team went to visit Oncor Electric, a delivery and distribution electrical company in Dallas, TX. We discussed about AMI, how meters communicate, their real distances, and got to see the equipment working in real-time. We learned that the distance meter to meter is between 0.1 to 0.2 miles, and repeaters cover a two-mile radius. Collectors are usually placed in a substation or distribution pole. The communication between meter and collector is encrypted and each meter has its own key or fingerprint. Oncor uses equipment for frequency

hopping in the 900MHz Industrial Scientific and Medical (ISM) band.

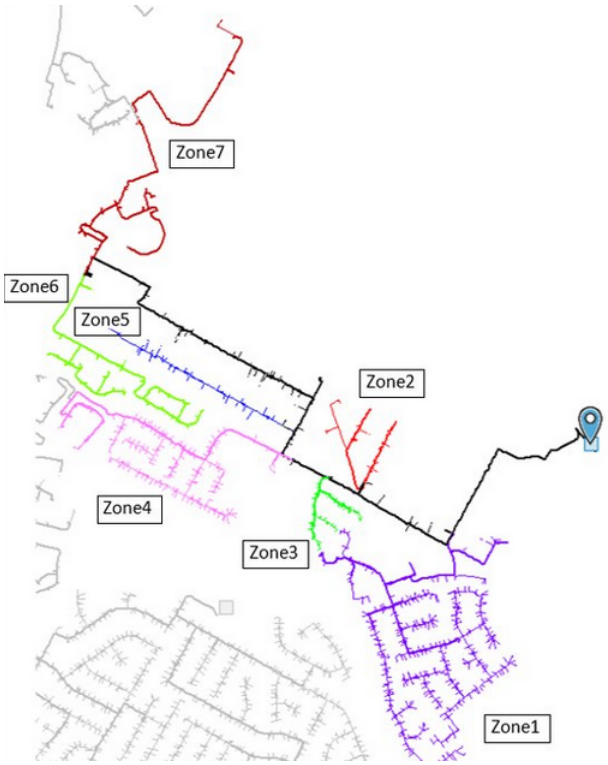

Figure 6.1: Zones for feeder P1UDT18962 in Austin, TX.

6.1 Network and Channel Configuration

All smart meters, repeaters, collector, and utility have nodes in CORE They all assume IP addresses in the 10.0.0.0/20 network. Fig. 6.4 shows the final network design for Zone 7. Each of these nodes are connected to its counterpart node in the power system simulation through two different Ethernet channels provided by the GTNETx2 board, separated into *Multi1* and *Multi2* configurations. Tables 6.1 and 6.2 show the configuration for each channel, and what nodes the channel connects from the RTDS to CORE, as well as the two IP addresses used for the communication. All of the channels in the RTDS are configured as TCP servers and the hosts in CORE are TCP clients, and similarly to Use Case 1, uses port 7001 for the data transfers between the RTDS and CORE. The smart meters are incorporated with computer nodes offered by CORE, and are

Figure 6.2: Equipment placement in Zone 7 for cyber network.

named RES for residential and COMM for commercial.

CORE offers the Wireless Local Area Network (WLAN) node to implement Wireless connections, a total of five WLAN nodes are used. Since repeaters have a two-mile radius, WLANs between routers are configured to have wider range compared to the WLANs connecting smart meters and they are positioned as far apart as the real distances in Zone 7 found using QGIS software. All WLAN nodes are configured to emulate real bandwidth of the links in use today. Authors in [1] describe the link operates at 9.6 kbps between smart meters and routers. Collectors and routers can communicate using 19.2 kbps or 9.6 kbps links.

In the model, we assume the collector is at the substation. Also we do not use encryption for the smart meter data. This use case simulates normal operation of the AMI system, and is further explained in the following section.

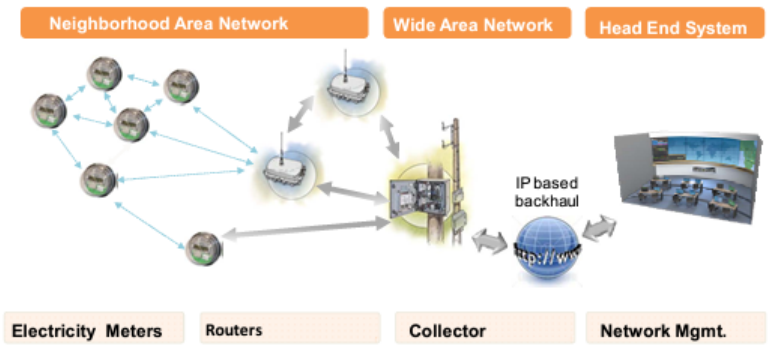

Figure 6.3: RF Mesh network diagram. Reprinted from [1].

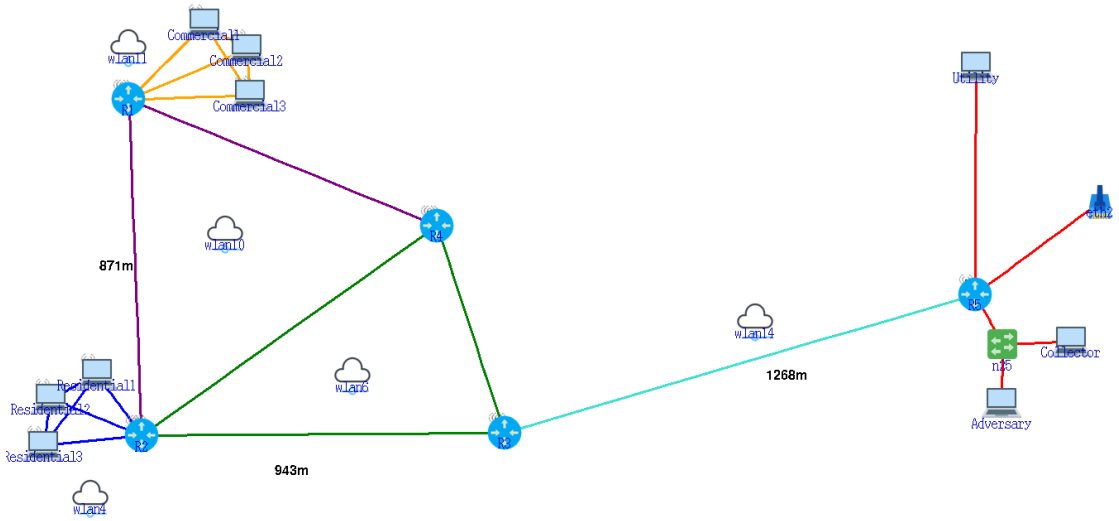

Figure 6.4: Use case 2 communication network emulated using CORE.

6.2 AMI Scenario

During normal operation of the AMI system, smart meters collect data (loads, PV, EV, BESS) from residential and commercial buildings through automatic readings and sends the data to the collector in a set interval of time, usually between fifteen minutes to one hour. The collector then forwards this data to the utility where the data is interpreted and set-points are calculated aiming to meet the demand for power. Fig. 6.5 shows a flowchart of the data-flow in this scenario, each step is explained below:

Table 6.1: Multi 1 Channel Configuration - Use Case 2

Channel	Node	IP Address RTDS	IP Address CORE	Function
1	Smart Meter RES1	10.125.184.172	10.0.2.20	Send
2	Smart Meter RES2	10.125.184.173	10.0.2.21	Send
3	Smart Meter COMM1	10.125.184.178	10.0.3.20	Send

Table 6.2: Multi 2 Channel Configuration - Use Case 2

Channel	Node	IP Address RTDS	IP Address CORE	Function
1	Utility	10.125.184.179	10.0.6.20	Send and Receive
2	Collector	10.125.184.180	10.0.5.20	Send and Receive

1. Load, PV, EV, BESS for each smart meter is sent from the RTDS to the smart meter nodes in CORE.

2. All smart meters in CORE forwards the data to a wireless mesh of routers or NAN.

3. Data from each smart meter arrives at the collector through the mesh.

4. Collector sends data pertaining to all meters to the utility node in CORE to an upper Wide Area Network (WAN), using fiber link.

5. Data from all meters are sent from the utility node in CORE to the utility in the RTDS.

Two different simulations are used for this scenario. *Simulation 1* is a transient non-real-time simulation, RTDS sends 24 data sets to each smart meter to simulate hourly updates throughout a whole day. The RTDS waits to receive an update from CORE utility node before sending the next meter update to the smart meters in CORE and runs for 200 seconds. *Simulation 2* runs in real-time, with the RTDS sending meter updates in an interval of 15 minutes for 2 hours (7200 seconds). The meter data models the behavior of the systems (Loads P and Q, PV, EV, BESS) from from 1PM to 3PM. Table 6.3 summarizes both experiments done for Use Case 2.

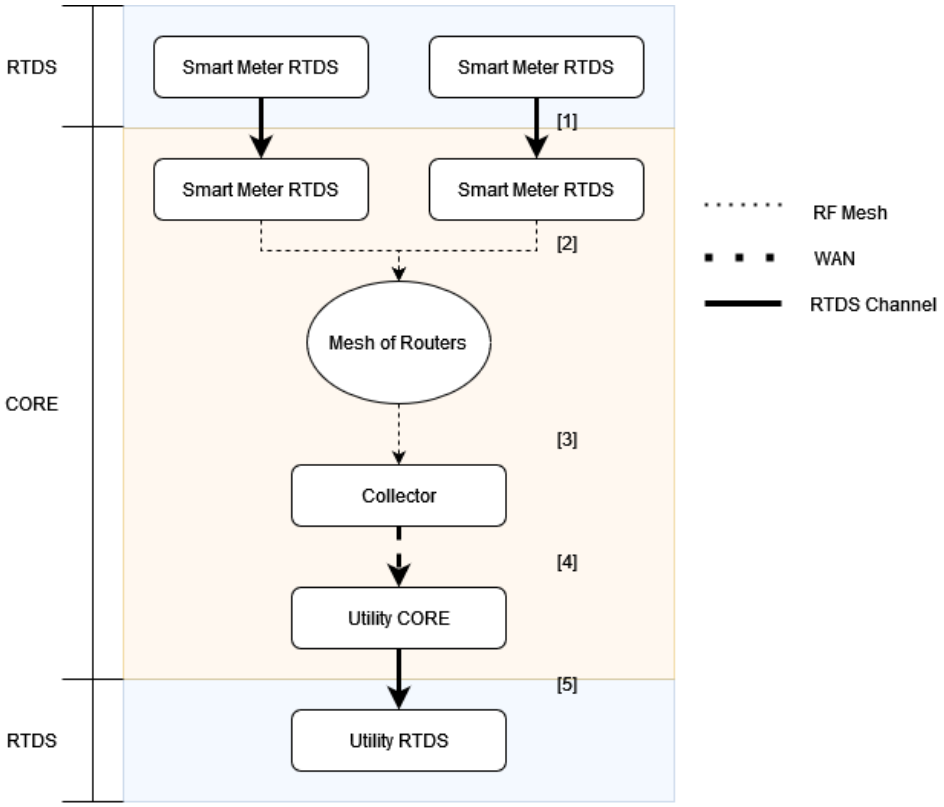

Figure 6.5: Scenario 1 flowchart.

Table 6.3: Simulations for Use Case 2.

Simulation	Type	Number of Meter Updates	Meter Update Interval	Simulation time
1	Transient	24	1 hr	$\sim$ 200 sec
2	Real-time	8	15 min	2 hrs (7200 sec)

Table 6.4: Cyber threats and simulation used to perform them.

Attack	Simulation used
Reconnaissance	*Simulation 1*
DoS	*Simulation 1*
MITM	*Simulation 1 and 2*

6.3 Cyber Threat Use Cases

The testbed was used to run three different cyberattacks among Simulation 1 and Simulation 2 experiemnts. Each attack is performed in CORE with the goal of impacting the simulation in the RTDS. The first use case is a reconnaissance attack that is used to find important data regarding the cyber network and servers running in the application. The second attack is a DoS attack where the utility node is targeted to affect its availability of processing and communicating data. The last use case is a man-in-the-middle attack, where the adversary is able to affect the integrity of the data exchanged between the collector and the utility. In summary, Table 6.4 displays what use cases were used for each simulation.

Figures 6.6 and 6.7 shows the normal operation of Commercial 1 meter data (loads, PV, EV) for Simulation 1. Figures 6.8 and 6.9 shows the normal operation of Commercial 1 meter data (loads, PV, EV) for Simulation 2.

6.3.1 Denial of Service

As explained in the previous use case, a Denial-of-Service attack is the exhaustion on networking and computational resources at one of the nodes has the objective of slowing down or impeding traffic through a link, as a result impacting the availability of the server. The DoS attack executed during the second use case also used *hping3* command line tool to create and send TCP SYN packets to the utility node. A timing diagram of the attack is shown in Fig. 6.10, where a delay is created between steps 4 and 5 where the utility node in CORE processes the data to send it to the utility node in the RTDS.

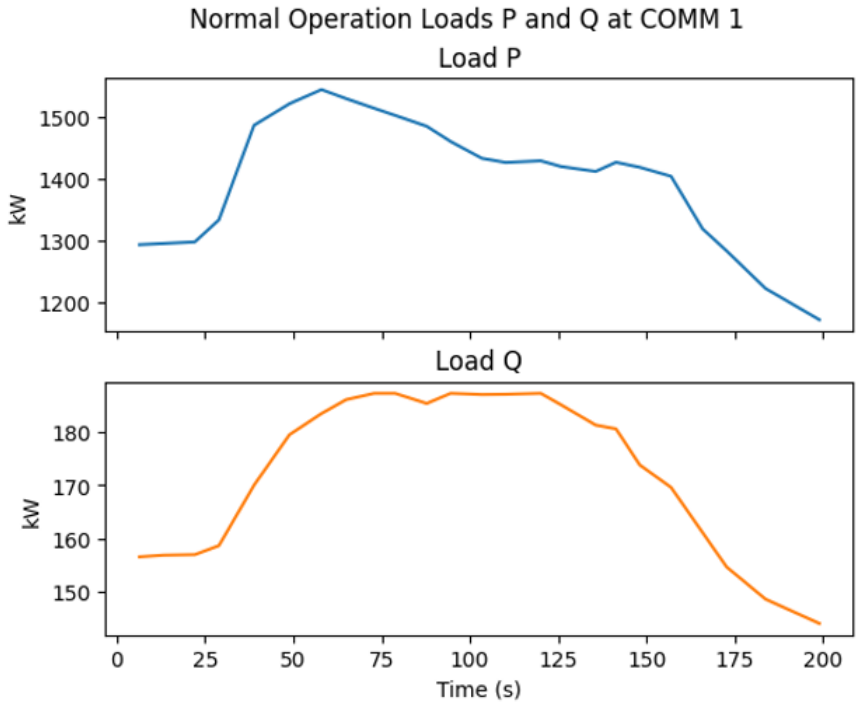

Figure 6.6: Normal operation of meter COMM1 Loads P and Q – Simulation 1.

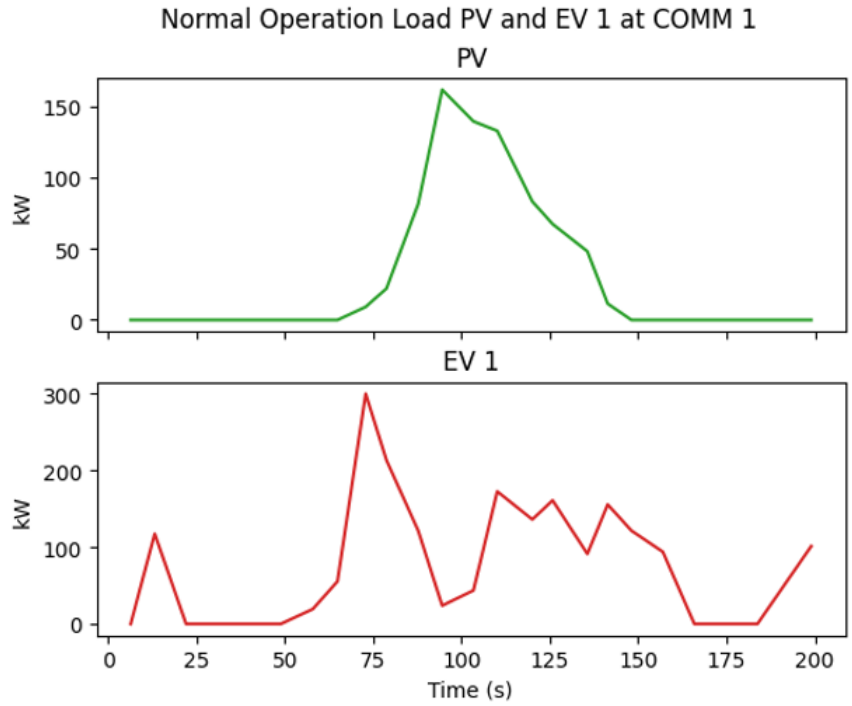

Figure 6.7: Normal operation of meter COMM1 PV and EV – Simulation 1.

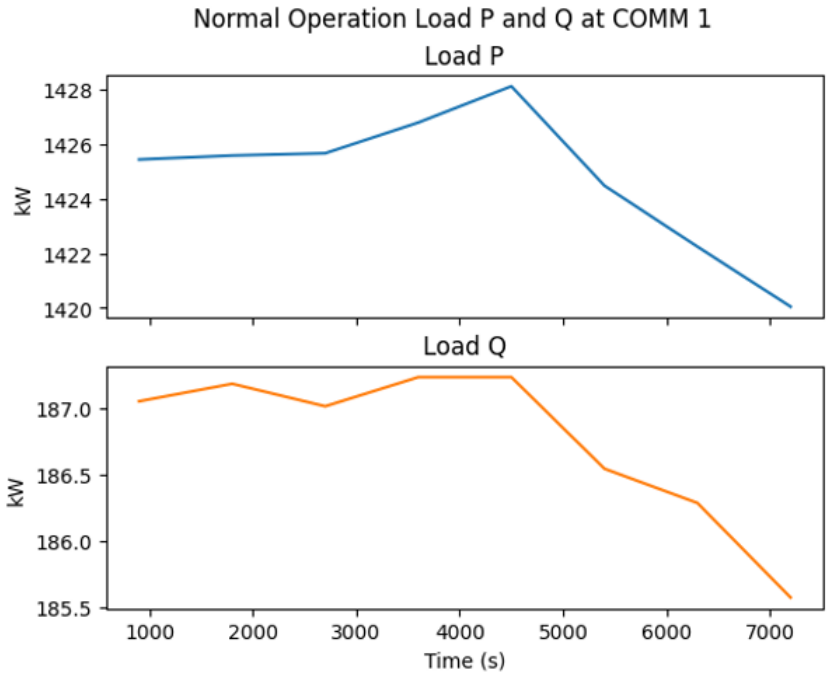

Figure 6.8: Normal operation of meter COMM1 Loads P and Q – Simulation 2.

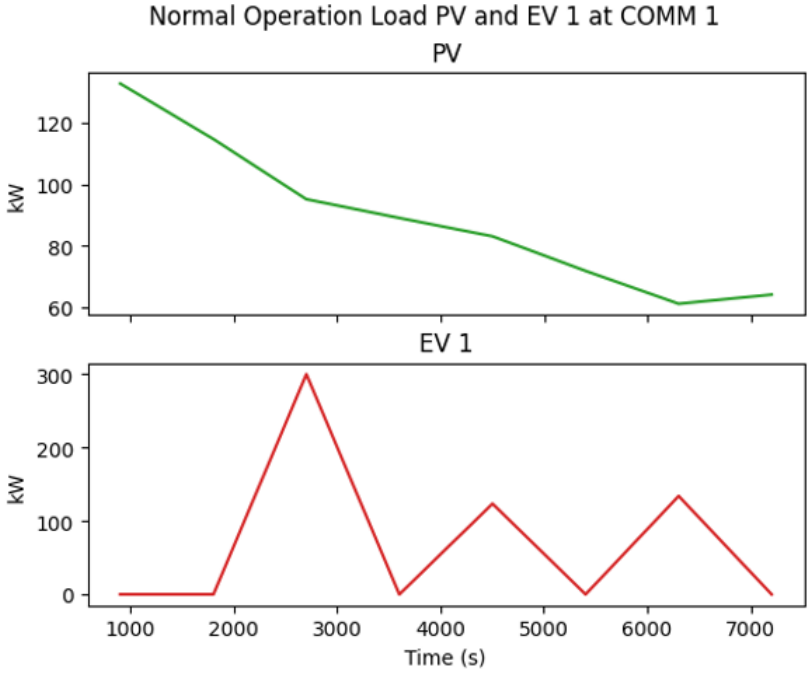

Figure 6.9: Normal operation of meter COMM1 PV and EV – Simulation 2.

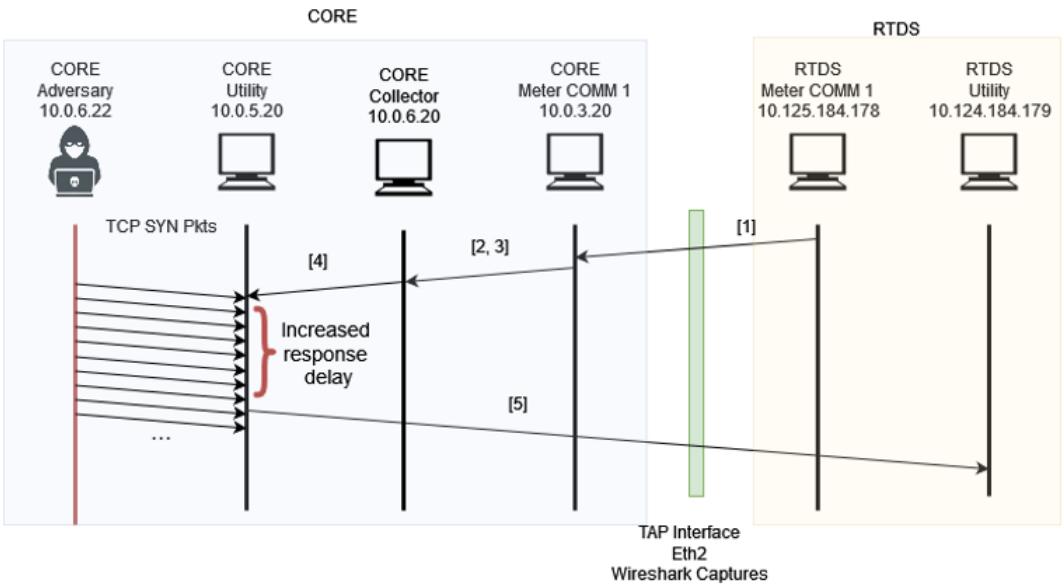

Figure 6.10: Timing diagram of data flow between CORE and RTDS during the DoS attack.

6.3.2 Man-in-the-Middle

The MITM attack uses the same steps and techniques used in the previous use case. First step is to enable automatic forwarding of packets the adversary will receive throughout the cyber attack. The second step is to perform the ARP Spoofing to manipulate the ARP tables of gateway and COMM 1 meter using Scapy command line tool. When the ARP tables are updated the adversary is already in between the communication, Fig. 6.11 shows the effective links of the communication vs the real link. A Bash script was built to automatically add the NFQueue rule to the Adversary's IP table while the packet modifier Python script is running. It implements two conditions: the first condition being the MITM is set to modify the payloads so packets are sent to NFQ, and the second condition is the MITM is not ready to modify the payloads so packets are automatically redirected without going to the queue set by NFQ. This automates the process of sending the packets to NFQ only when the adversary wants to inject malicious TCP payloads into the communication between COMM 1 and the collector, and fixes the IP tables rules to not use NFQ while the adversary does not want to change the payloads of the packets. If the Bash script is not running, the adversary is

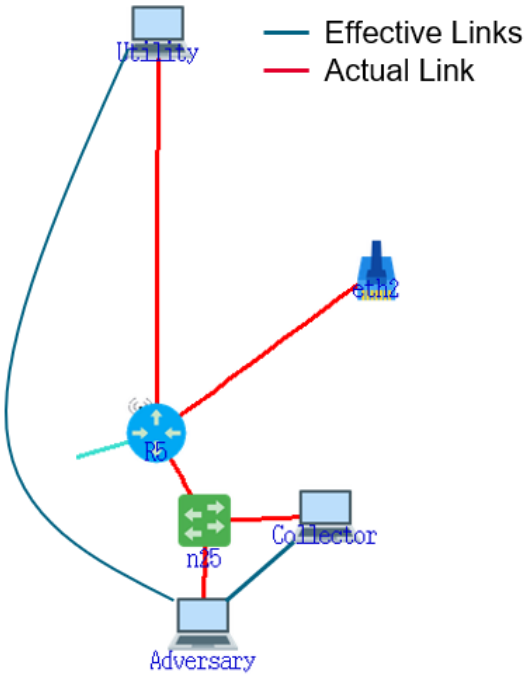

Figure 6.11: Effective vs Actual link during MITM Attack Use Case 2.

only able to eavesdrop on the communication since the packets are automatically forwarded to the final destination.

The packet modifier Python script looks for a packet containing the data from COMM 1 smart meter node. It modifies the data for the load readings (P and Q), as well as the PV and the EV readings. The modified values are strategically chosen to benefit COMM 1 node financially. The loads P, Q and EV readings are decreased to make it seem they are using less power and PV reading is increased to financially benefit COMM 1 node with false values of power generated by their solar panels.

6.4 Results

This section compiles the results for the different cyber threats explained in the previous section. Their data sets were acquired through Wireshark, a historian database programmed for each CORE node that keeps track of every packet received and the time, and a data acquisition tool from

```
Nmap scan report for 10.0.5.20
Host is up (0.00018s latency).

PORT      STATE SERVICE VERSION
15001/tcp open  unknown
Warning: OSScan results may be unreliable because we could not find at least 1 o
pen and 1 closed port
Aggressive OS guesses: Linux 2.6.32 (96%), Linux 3.2 - 4.9 (96%), Linux 2.6.32 -
 3.10 (96%), Linux 3.4 - 3.10 (95%), Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A
 or 211 Network Camera (Linux 2.6.17) (94%), Synology DiskStation Manager 5.2-56
44 (94%), Netgear RAIDiator 4.2.28 (94%), Linux 2.6.32 - 2.6.35 (94%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops

TRACEROUTE (using port 15001/tcp)
HOP RTT     ADDRESS
1   0.03 ms 10.0.6.1
2   0.08 ms 10.0.5.20
```

Figure 6.12: NMAP results displaying TCP port 15001 open on the Utility.

the RTDS. This section provides insights on the impact of the cyberattacks on the communication side of the testbed, as well as patterns found in the malicious data.

6.4.1 Reconnaissance

The reconnaissance showed the utility's TCP server running in port 15001 used in the communication with the collector. Fig. 6.12 shows the results of the reconnaissance attack. Other information, such as OS version and route information were collected. This helps the adversary understand the network topology and services used in an advanced metering infrastructure.

6.4.2 Denial of Service

The attacker upon executing a reconnaissance attack is able to impact the availability of the server on the utility node by slowing down the data exchange the utility has with the collector. The impact of the DoS attack can be seen in Figure 6.13 and 6.14 which displays the load values (P and Q), as well as PV, and EV data points. The area marked by the red squares represent time intervals where the DoS attack stopped or slowed down the communication of the AMI system. During normal operation, the total runtime of Simulation 1 in RTDS is about 200 seconds. However, when the DoS attack begins, the duration of the simulation increases to over 250 seconds, where constant values are perceived by the utility while the collector tries to send updated smart meter data..

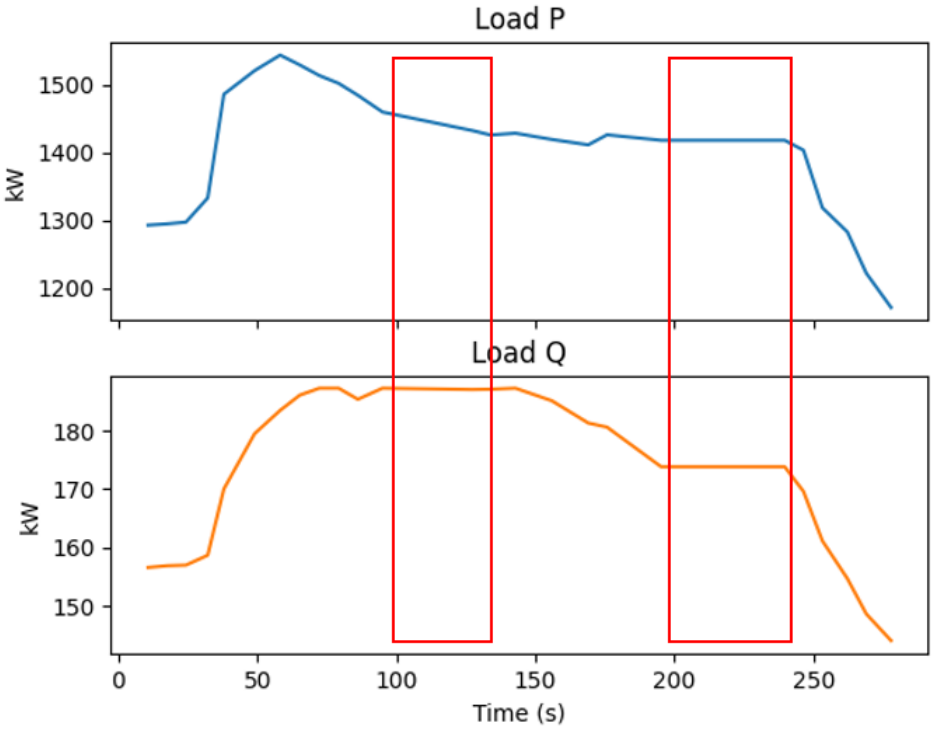

Figure 6.13: DoS Impact on distribution assets – Simulation 1.

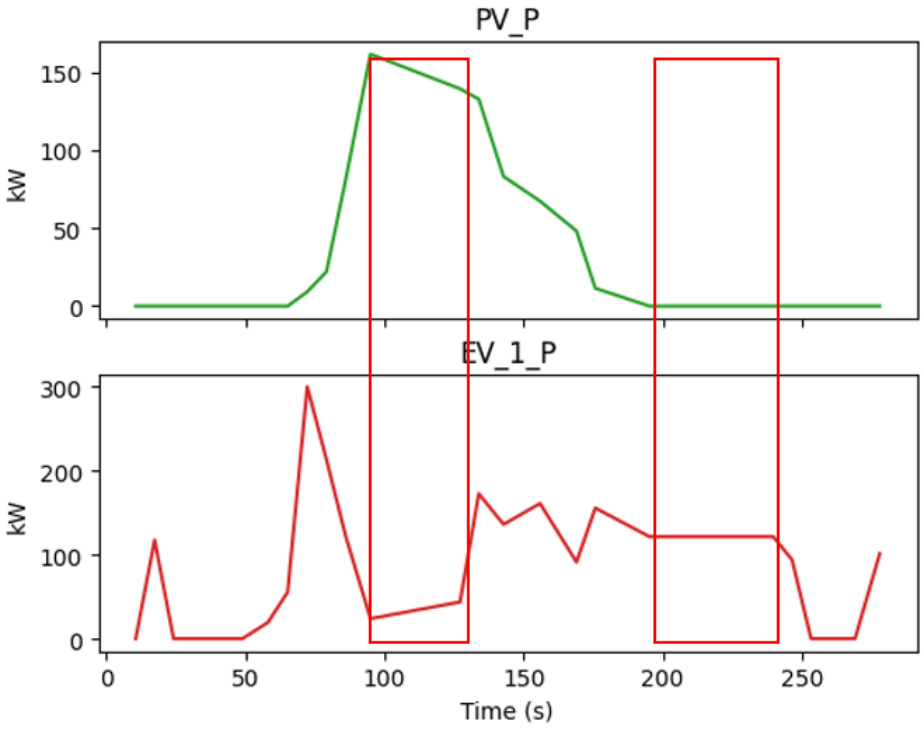

Figure 6.14: DoS Impact on distribution assets – Simulation 1.

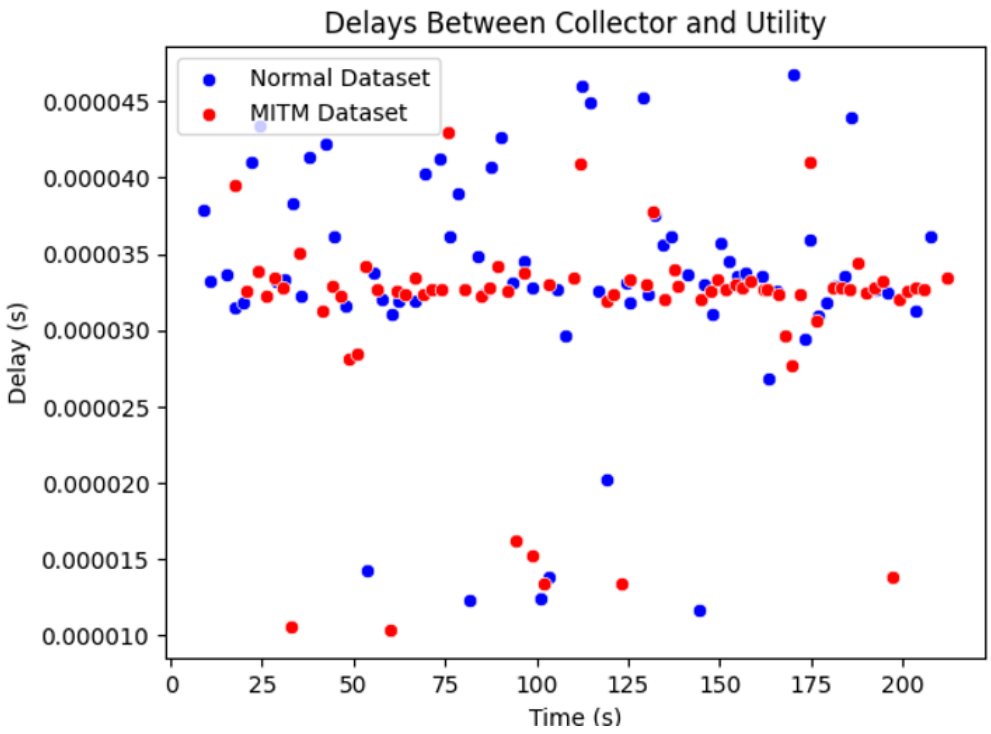

Figure 6.15: MITM impact on communication delays – Simulation 1.

6.4.3 Man-in-the-Middle

A comparative analysis is done to observe the impact of the MITM attack on the communication network by showing results from normal and affected operations. Two different sets of data were captured using Wireshark. Fig. 6.15 shows the observed round-trip times (RTT) from when the smart meter data is sent from the collector to the utility with a *PUSH/ACK* TCP flag to when a acknowledgement, *ACK* TCP flag, is received by collector during normal operation of the distribution system and with MITM attack.

Differently from the results from Use Case 1, where there was a clear disparity between the normal and MITM delays, the delay for Use Case 2 seems to come from the same distribution, with no meaningful difference between the two. A T-Test assuming unequal variances between the samples was utilized to see if there is a statistical difference, with the null hypothesis stating the two means are the same, and the alternate hypothesis stating the population means are different.

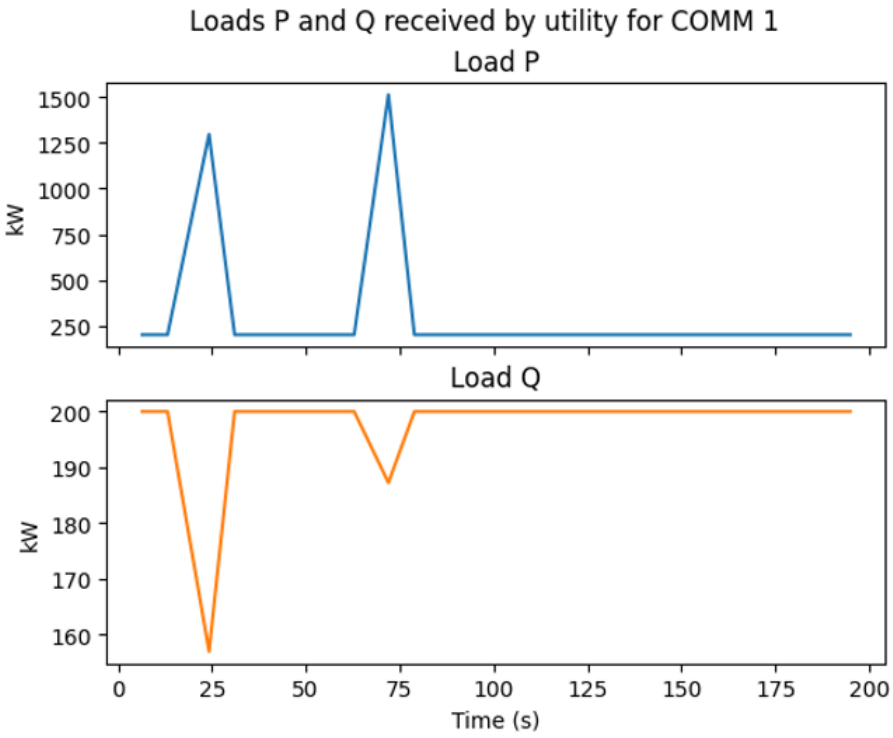

Figure 6.16: MITM Impact on COMM1 meter data for loads P and Q – Simulation 1.

The test yielded a T-statistic of 1.9622 and a P-value of 0.0513, leading me to not reject the null hypothesis that the mean delays between normal operation and during the MITM attack have no meaningful statistical difference.

Although a difference in the delays is not observable, the attack was still successful for both simulations. Figures 6.16 and 6.17 displays the data for COMM 1 seen by the utility. The values modified were chosen strategically to give COMM 1 node a financial advantage over the other nodes, by decreasing its consumed power through load P, Q, and EV and increasing its generated power through PV. A value of 250 kW is set for load P and a value of 200 kW is set for load Q. The value of PV is increased to 165 kW and EV is decreased to 0 kW for simulation 1. The results for this simulation has two peaks, which are real smart meter data from COMM 1 passing through the MITM attack with no modification. Simulation 2 results had one hundred percent of its packets modified by the attacker (seen by the constant values for each data-point), with the difference being load P was set to 200 kW.

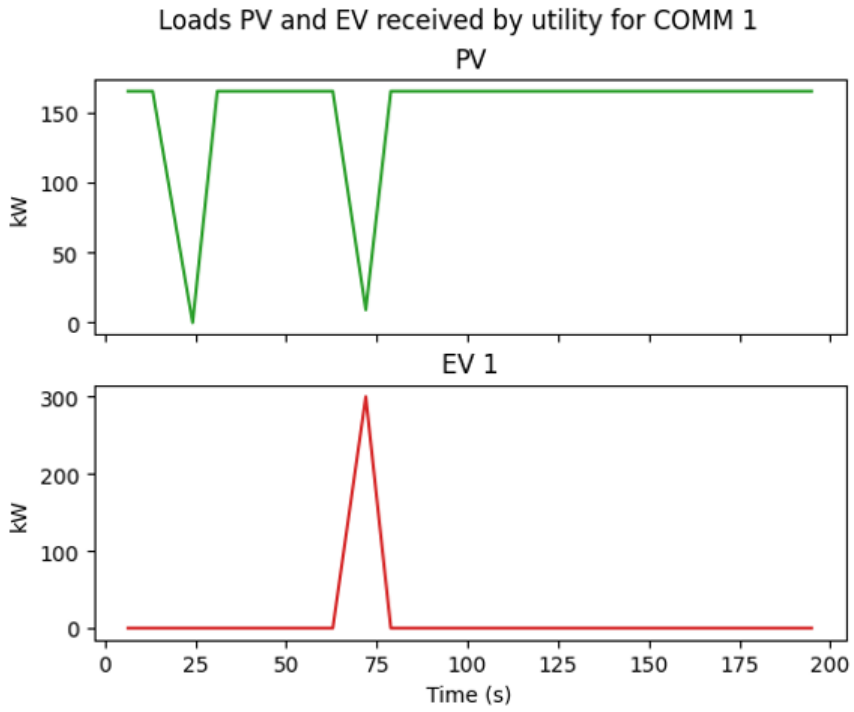

Figure 6.17: MITM Impact on COMM1 meter data for PV and EV – Simulation 1.

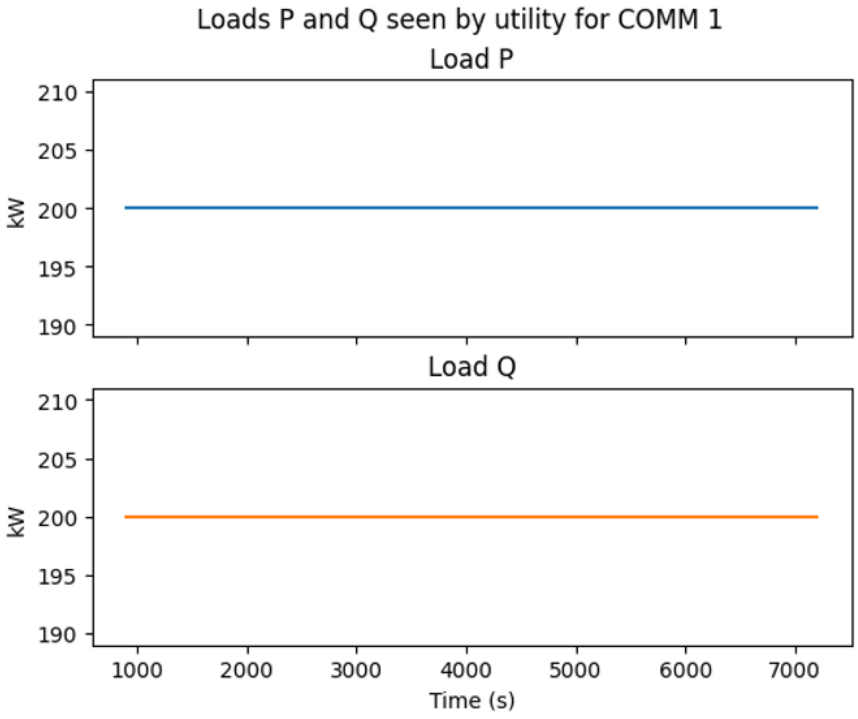

Figure 6.18: MITM Impact on COMM1 meter data for loads P and Q – Simulation 2.

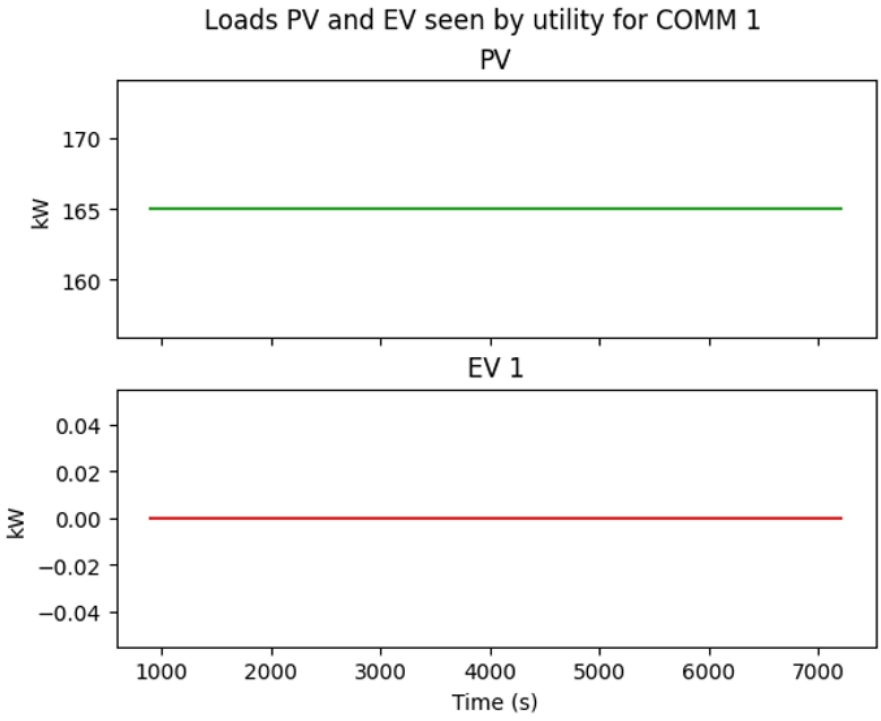

Figure 6.19: MITM Impact on COMM1 meter data for PV and EV – Simulation 2.

7. CONCLUSION

Smart power distribution systems have a critical role in the distribution of power consumers to meet the required demand. The automation of these systems by incorporating communication technology poses a great advance in how power is distributed across end consumers and DERs. On the other hand, the use of communication technology brings the risks of cyber threats. This thesis analyzed the impact of different cyber threats in a smart distribution system scenarios through the construction of a dynamic cyber-physical testbed using the RTDS machine with a GTNETx2 board and CORE network emulator.

The research provided insights into how cyber threats can affect a distribution system in different ways. A reconnaissance attack, although simple in this use case, provides important information regarding the communication configuration (IP addresses, open ports and services, and operating systems) that is used to plan and coordinate big and elaborated attacks. The Denial of Service attack results provide insights on the availability impact of critical nodes in the infrastructure. By interrupting the communication, the attacker is able to delay time critical information from getting to its destination, i.e. meter data to the utility in order to balance supply and demand of power. The Man-in-the-Middle attack shows the impact on both subsystems, cyber and physical. By modifying packet values, the attacker is able to access critical information, affect proper billing from the utility, and provide false values to be used in set-point calculations for other nodes in the distribution system.

The findings and results for the experiment contribute to a deeper understanding in the creation of dynamic CPS testbeds to simulate cyber threats in a realistic scenario. It shows the importance of creating controlled environments to run and test the cyber resilience of new software, protocols, and network configurations for critical infrastructure. Despite the valuable insights given by the results, the study contain limitations on computational power from both computers and the RTDS. The RTDS has limitation on the number of channels it can use in a simulation, with a maximum 30 data points per packet while using the GTNET-SKT configuration. The computer running Linux

may not have enough memory and processing power for large use cases with several DERs and/or smart meter containers. Future use cases could explore the integration of more components into the distribution system, DERs, smart meters, or the integration of hardware-in-the-loop (HIL) systems to further study and visualize the impact of emerging cyber threats.

Ultimately, this research provides a foundation for further exploration into realistic cyber-physical testbeds for distribution systems, contributing to a broader understanding of possible cyber threats related to smart distribution systems and its potential impact on the functionality and operation of such systems.

REFERENCES

[1] B. Lichtensteiger, B. Bjelajac, C. Müller, and C. Wietfeld, "RF Mesh Systems for Smart Metering: System Architecture and Performance," in *2010 First IEEE International Conference on Smart Grid Communications*, pp. 379–384, 2010.

[2] C. Stupp, "European Wind-Energy Sector Hit in Wave of Hacks," April 2022. https://www.wsj.com/articles/european-wind-energy-sector-hit-in-wave-of-hacks-11650879000.

[3] P. Kozak, I. Klaban, and T. Šlajs, "Industroyer cyber-attacks on Ukraine's critical infrastructure," in *2023 International Conference on Military Technologies (ICMT)*, pp. 1–6, 2023.

[4] C. Devanarayana, Y. Zhang, and R. Kuffel, "Testing Cyber Security of Power Systems on a Real Time Digital Simulator," in *2019 IEEE 8th International Conference on Advanced Power System Automation and Protection (APAP)*, pp. 1166–1170, 2019.

[5] J. Ahrenholz, C. Danilov, T. R. Henderson, and J. H. Kim, "CORE: A real-time network emulator," in *MILCOM 2008 - 2008 IEEE Military Communications Conference*, pp. 1–7, 2008.

[6] C. W. Asbery, X. Jiao, and Y. Liao, "Implementation guidance of smart grid communication," in *2016 North American Power Symposium (NAPS)*, pp. 1–6, 2016.

[7] "Assesment of Demand Response & Advanced Metering ," in *Federal Agency Regulation Commission Staff Report*, December 2012.

[8] B. Palmintier, T. Elgindy, C. Mateo, F. Postigo, T. Gómez, F. de Cuadra, and P. D. Martinez, "Experiences developing large-scale synthetic U.S.-style distribution test systems," *Electric Power Systems Research*, vol. 190, p. 106665, 2021.

[9] "Using GTNET and NS-3 for cyber-physical simulation," tech. rep., RTDS Technologies Inc., October 2019.

[10] O. Bel, J. Kim, W. Hofer, M. Maharjan, S. Purohit, and S. Niddodi, "Co-Simulation Framework For Network Attack Generation and Monitoring," June 2023.

[11] S. East, J. Butts, M. Papa, and S. Shenoi, "A Taxonomy of Attacks on the DNP3 Protocol," vol. 311, 03 2009.

[12] Z. Zhou, H. Yang, H. Li, J. Zhang, S. Li, X. Gao, and P. Gong, "A Dynamic Cyber-attack Approach for Real-time Hardware-in-the-loop Simulation of Power Grid," in *2022 24th International Conference on Advanced Communication Technology (ICACT)*, pp. 212–217, 2022.

[13] A. S. Musleh, J. Ahmed, N. Ahmed, H. Xu, G. Chen, J. Hu, and S. Jha, "Development of a Collaborative Hybrid Cyber-Physical Testbed for Analysing Cybersecurity Issues of DER Systems," in *2023 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pp. 1–6, 2023.

[14] B. Chen, K. L. Butler-Purry, A. Goulart, and D. Kundur, "Implementing a real-time cyber-physical system test bed in RTDS and OPNET," in *2014 North American Power Symposium (NAPS)*, pp. 1–6, 2014.

[15] F. Malandra and B. Sansò, "Performance analysis of large scale RF-mesh networks for smart cities and IoT," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 13–18, 2017.

[16] P. Yi, T. Zhu, Q. Zhang, Y. Wu, and J. Li, "A denial of service attack in advanced metering infrastructure network," in *2014 IEEE International Conference on Communications (ICC)*, pp. 1029–1034, 2014.

[17] K. I. Sgouras, A. D. Birda, and D. P. Labridis, "Cyber attack impact on critical smart grid infrastructures," in *Innovative Smart Grid Technologies (ISGT) 2014*, pp. 1–5, 2014.

[18] K. Pedramnia and M. Rahmani, "Survey of DoS Attacks on LTE infrastructure used in AMI System and Countermeasures," in *2018 Smart Grid Conference (SGC)*, pp. 1–6, 2018.

[19] A. Sahu, P. Wlazlo, Z. Mao, H. Huang, A. Goulart, K. Davis, and S. Zonouz, "Design and evaluation of a cyber-physical testbed for improving attack resilience of power systems," *IET Cyber-Physical Systems: Theory & Applications*, 2021.

[20] S. Hossain-McKenzie, N. Jacobs, A. Summers, B. Kolaczkowski, C. Goes, and R. Fasano, "Harmonized Automatic Relay Mitigation of Nefarious Intentional Events (HARMONIE) – Special Protection Scheme (SPS)," 2022.

[21] "IEEE Standard for Floating-Point Arithmetic," *IEEE Std 754-2019 (Revision of IEEE 754-2008)*, pp. 1–84, 2019.

[22] M. Korman, M. Ekstedt, O. Gehrke, and A. Kosek, "Deliverable 1.1 Smart grid scenario: Project: Cyber-phySicAl security for Low-VoltAGE grids (SALVAGE)," , KTH - Royal Institute of Technology, DTU - Technical University of Denmark, PWR - Wroclaw Institute of Technology, 2015.

[23] I. Odun-Ayo, "A review of common tools and techniques for reconnaissance attacks," in *28th iSTEAMS Multidisciplinary Reserach Conference*, pp. 141–157, October 2021.

[24] E. Liu, Y. Li, and L. Du, "Denial of Service Cyberattacks to Naval Software Defined-Networking-Enabled SCADA Network," in *2022 IEEE Transportation Electrification Conference Expo (ITEC)*, pp. 986–990, 2022.

[25] M.-H. Wu, C.-Y. Chiu, J.-Z. Wu, J.-H. Huang, J.-X. Chen, and H.-J. Wang, "Mechanisms for Enhancing Network Services by Live Migration in the Network," in *2023 IEEE 6th Eurasian Conference on Educational Innovation (ECEI)*, pp. 10–13, 2023.

[26] "CYME power engineering software solutions," September 2024. https://www.eaton.com/us/en-us/digital/brightlayer/brightlayer-utilities-suite/cyme-power-engineering-software-solutions.html.