FIREWALL CONFIGURATION AND PATH ANALYSIS FOR SMART GRID NETWORKS

A Thesis

by

NASTASSJA GHISLAINE GAUDET

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

MASTER OF SCIENCE

Chair of Committee,	Ana Goulart
Committee Members,	Katherine Davis
	Behbood Zoghi
Head of Department,	Jorge Leon

December 2020

Major Subject: Engineering Technology

ABSTRACT

The objective of this research is to develop a complete cyber topology model of the Texas 2000-bus synthetic grid, and to study the data flow through utility companies to defend their networks from cyber-attacks. Specifically, this work focuses to create a set of firewall rules and configurations in a model network, optimize them by testing them against various attacks, then translate them to *iptables* to be used in our teams test bed. Cisco Packet Tracer will be used to create and test a network with various protocols allowed and denied at specific nodes in the network. This sample network has a utility control center network, a substation network, and a balancing authority network. Then Network Perceptions NP-View software is used to run and analyze all firewall and router configuration files for a complete path analysis and risk assessment. The final goal is to understand every possible path into and out of each network, who is permitted to use these paths, and where an attacker might exploit the network. Then these possible attacks are simulated, traced, and studied, to allow for a better network topology.

DEDICATION

To my mother, for everything she’s done for me.

ACKNOWLEDGMENTS

I would first like to thank my professor and advisor Dr. Ana Goulart at Texas A&M University, for working with me so closely and providing me with this research opportunity. Dr. Goulart has always provided the advice and wisdom throughout my journey through graduate school. Her willingness to read my work and offer feedback was essential to the success of this project.

This project would not have been possible without Dr. Katherine Davis at Texas A&M University as well, for having the insight and knowledge to further develop the area of cybersecurity in power systems. I want to thank her for being on my committee and providing support. This project would not have been possible without her contributions and experience.

Throughout my entire college career, Dr. Behbood Zoghi at Texas A&M University has been a mentor to me. I thank him for the years of wisdom and support. His door was always open to me when I needed advice or clarity, whether it was about my classes or future career direction. I am grateful to him for serving on my committee.

Next, I want to thank Mr. Edmond Rogers, from the University of Illinois, for all of his expertise and wisdom throughout this project. He has been a wealth of knowledge in the field of cybersecurity and smart grid networks.

I would also like to thank Abhijeet Sahu, another student on the CYPRES team, especially for his knowledge and contributions during this endeavor. He helped to develop the data flow model used in this project, and was always there with a solution or idea when one was needed.

I want to express my gratitude to all of my professors during my time at Texas A&M University who have provided me with the educational guidance and support that has directed me to where I am today.

Finally, I want to thank my mom, for her undying support and encouragement that has guided me along the path towards a successful and happy life.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professors Dr. Ana Goulart and Dr. Behhood Zoghi of the Department of Engineering Technology and Industrial Distribution and Professor Dr. Katherine Davis of the Department of Electrical and Computer Engineering.

All other work conducted for the thesis was completed by the student, under the advisement of Dr. Ana Goulart of the Department of Engineering Technology and Industrial Distribution.

Funding Sources

Graduate study was supported by the US Department of Energy Cybersecurity for Energy Delivery Systems program under award DE-OE0000895.

NOMENCLATURE

SCADA	Supervisory Control and Data Acquisition
UCC	Utility Control Center
BA	Balancing Authority
DMZ	Demilitarized Zone
ACL	Access Control List
IP	Internet Protocol
DNP3	Distributed Network Protocol 3
ICCP	Inter-Control Center Protocol
SQL	Structured Query Language
SSH	Secure Shell
HTTP	HyperText Transfer Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
SNMP	Simple Network Management Protocol
OT	Operational Technology

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

Cybersecurity is an essential and relevant issue to smart grid networks as any cyber-attack could have disastrous consequences. It is therefore important to configure the network to support the power grid and protect it from as many threats as possible. How can a smart grid network be configured to mitigate risk and defend against attacks? How can firewalls be configured to support this critical infrastructure such as the electric grid? Is it possible to audit these firewalls to identify vulnerabilities or assess risk in an electric utility's operations technology (OT) network? What type of data flows and Demilitarized Zones (DMZ's) are typically used in OT networks? To be compliant with the North American Electric Reliability Corporation (NERC) Critical Infrastructure Protection (CIP) NERC-CIP-005 [1], an *electronic system perimeter* is needed. These security perimeters are implemented using firewalls that are carefully configured to protect the utilities' data flows, which are predictable and deterministic, in comparison with traditional data flows.

1.1 Background

The Cyber Physical Resilient Energy Systems (CYPRES) project [2] is working to develop a complete cyber topology model of the Texas 2000-bus synthetic grid [3] [4], and study the data flow through utility companies to defend their networks from cyber-attacks [5]. This project extends the Cyber-Physical Security Assessment (CyPSA) project [6] which provides a common format for future planning and risk assessment of the electric grid. CyPSA models the cyber-physical dependencies within the electric power grid, while adopting a common format using cyber-physical topology language (CPTL) [7]. The model used previously for CyPSA is of the electric and communication systems of an 8-bus substation [8] including communication links such as Ethernet, serial, multi-mode fiber, and a central control center. Each node was then assigned a type, such as a distance relay or a communications device, and an IP addresses. The CyPSA application then performed a complete analysis and risk assessment of the network.

In the research to develop a communications model of the Texas 2000-bus synthetic grid model, the configuration of firewalls in electrical utility companies, from the substation level, to utility control center (UCC), and balancing authority (BA) must first be addressed. This work extends the communications model presented in [5] to include firewall configurations. These allow for risk assessments, simulate different firewall configurations, and perform training to the existing and future workforce. Our ultimate goal is to optimize the firewall rules to reduce vulnerabilities in a utilities' OT network, following the approach used in [9] in which an attack tree model is used to quantify the vulnerabilities in supervisory control and data acquisition (SCADA) systems.

The network policies for the model use the concept of least-privileges in that access to resources is limited to single hosts on a predetermined Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) port-based [10] on the requirements of the operation of the power system. Only ports and services necessary for operation are allowed. The control system is used as a central point. The control system initiates connections out to field devices, processes data in real time, then pushes results in real time out to servers in DMZ zones toward corporate and partner operations. Connections towards the control network are not allowed and this helps to reduce the network attack surface of the control system.

1.2 Literature Review

This research stemmed from a need to reinforce and secure the power grid networks in the United States. In December 2015, there was a major attack on a power grid in Ukraine, which was the first known successful power grid attack [11][12]. About 225,000 people were without power for hours. This attack began with phishing attacks to gain access to the virtual private network (VPN) for the control center, and then with denial-of-service (DoS) attacks. This allowed the hackers to gain access to the control center network from outside and control the substation remotely while masked as an insider. The control center then had to manually operate their devices and instruments in the substation while they worked to secure the network. In a paper about this cyber-attack, the authors assume that the attacker will have knowledge about the power system as well as the network topology and operation information in order to operate the system [11]. They further recommend regular cyber security training for all staff and the capability for both automatic and manual operating modes for the operational technology.

In addition to power grid attack, other cyber physical systems were subject to major cyber-attacks. As an example, one type of malware was able to infect many industrial sites in Iran [13]. This computer worm as referred to as Stuxnet, and was able to infiltrate a network, once installed, possibly through a phishing attack. This attack had three stages: one that targeted Windows computers to replicate itself across the entire network, a second that targeted Windows-based software that controls industrial equipment, and a third that targeted programmable logic controllers (PLC's) for the industrial equipment. A major target was centrifuges, as the worm was able to first over stress the rotors to cause the machinery to breakdown, allowing the malware to be undetected initially. Later, the malware increased the speed of the rotors, allowing the attackers to gain full control of the industrial sites. Attacks such as these have catastrophic consequences.

The use of firewalls is one method of securing a network and mitigating risk. Firewalls are network devices capable of inspecting packets and further allowing them into the network or discarding them. They match the source and destination addresses to configured rules to determine what to do with each packet. Port numbers are also matched to determine if the specific application used should be allowed or denied between specific source and destination nodes. If a port is left open and not specified, there are many opportunities for attacks since there are many possible paths into the network [14]. In this paper, the effect of various cyber attacks onto a smart grid network was studied, to understand the integration between the cyber network and the power systems.

Research was further conducted into the previous methods of analyzing and configuring firewalls. Some researchers have been able to quantify error in configuration files by defining error as deviation from industry regulations and best-practices, which I will be using as well. One author said that the hardest part is defining and quantifying what entails a configuration error in a firewall file [15]. The author was able to obtain and study configuration files from multiple resources and industries to show that it is not just one industry with network vulnerabilities. He studied the operating system, the firewall version, and mostly the firewall rule complexity. To measure the complexity of the configurations, he added the amount of object groups supporting the rule set to the number of rules in the set to the number of interfaces multiplied by the number of interfaces minus 1, and divided by 2. This gave the total combinations of rule sets per interface and took into account the number of object groups. He then determined the twelve greatest configuration errors and evaluated them. This will be helpful to expand upon.

Network Perceptions NP-View was developed to determine if a company's current or proposed network security is optimal and meets both regulatory standards and best practices [16]. It is a tool that has been tied to this research for over a decade finding its roots in the Trustworthy Cyber Infrastructure for the Power grid (TCIP) research program, and then the CyPSA project. In addition, NP-View is the standard tool used by the North American Electric Reliability Corporations (NERC) Critical Infrastructure Protection (CIP) auditors to measure network attack surface for Critical Assets in CIP 005. The software allows the user to upload a set of firewall configurations and automatically creates a model of the network topology which is then used to complete a risk assessment. It was designed to be scalable to real-world networks.

The goal of this project will be to configure a model of a smart grid network to be as secure as possible, following industry standards and best practices. The model network will be secured using firewalls to control both incoming and outgoing traffic, by configuring them with rules to limit traffic to specific hosts or nodes on needed application ports and in specified directions. These rules will first be created and configured on Cisco Adaptive Security Appliances (ASA) routers [8], and then converted to Linux-based *iptables*. Our test bed for the CYPRES project uses several Linux virtual machines as firewalls, therefore *iptables* are more practical for our research. This project will incorporate CyPSA as well, which is currently used in our test bed with the PowerWorld power simulator [17], to create a real-time version that could function with NP-View.

1.3 Tools for Modelling Firewalls

There are several tools to model computer networks and firewall configurations. For instance, Cisco Packet Tracer is an educational network simulation tool that allows users to simulate and test network architectures that they create [18]. The model presented in this paper began with a network in Cisco Packet Tracer to show the basic topology including a UCC, substation, BA, and a few DMZ's.

Next, we used NP-View [16] to analyze several firewall and router configuration files and run a path analysis for a risk assessment. NP-View was developed to determine if a company's current or proposed network security is optimal and meets both regulatory standards and best practices [19]. It is a tool that has been tied to this research for over a decade finding its roots in the Trustworthy Cyber Infrastructure for the Power grid (TCIP) research program, and then the Cyber-Physical Security Assessment (CyPSA) project [6]. In addition, NP-View is the standard tool used by NERC CIP auditors to measure network attack surface for Critical Assets in CIP 005 [1].

NP-View uses a path analysis approach to determine a full risk assessment based on the firewall and router configurations submitted to the software. It reads the IP addresses set on each interface and the object-groups [20] created in the file to determine the total number of networks known to each device. NP-View is able to determine every path into and out of a network, and give a warning for the risk level of that particular path. This path includes both the source and destination addresses, as well as the protocol and port used to access the network, which was useful in optimizing our configurations.

2. DATA FLOWS AND NETWORK TOPOLOGY FOR SMART GRID NETWORKS[*]

The goal of this research is to produce a set of firewall rules and configurations to be implemented in a smart grid network for maximum security. The first step in designing these rules is to define exactly what traffic should be allowed into and out of the network. It is important to know what should be permitted, before securing the network by denying traffic.

2.1 Definition of Data Flows

To successfully configure a set of firewalls for a smart grid network, a model network first has to be drawn to show the most important data flows. These controlled flows should be the only flows permitted into and out of the power grid network. All other traffic will be denied or discarded to secure the network. To accomplish this, these data flows were listed:

- telemetry data requests from control center to substation,
- data from control center to corporate and other DMZ's,
- data from control center to balancing authority.

Figure 2.1 shows an overview of each of the data flows, color-coded by the protocol used. The firewalls are configured so that only certain traffic is allowed between specified devices with all other applications and users blocked for security. The model was created with five main firewalls protecting a utility control center (UCC), substation, and balancing authority (BA). Using Figure 2.1 as a reference, the data flows are explained next beginning from the substation at the bottom of the figure.

2.1.1 DNP3 Protocol

Distributed Network Protocol 3 (DNP3) is a protocol used to control a remote network from a central network. This is mainly used in utility networks with SCADA systems. In industry,

[*] Reprinted with permission from Firewall Configuration and Path Analysis by Authors Names, 2020. IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR), pp. 1-6, l' 2020 IEEE

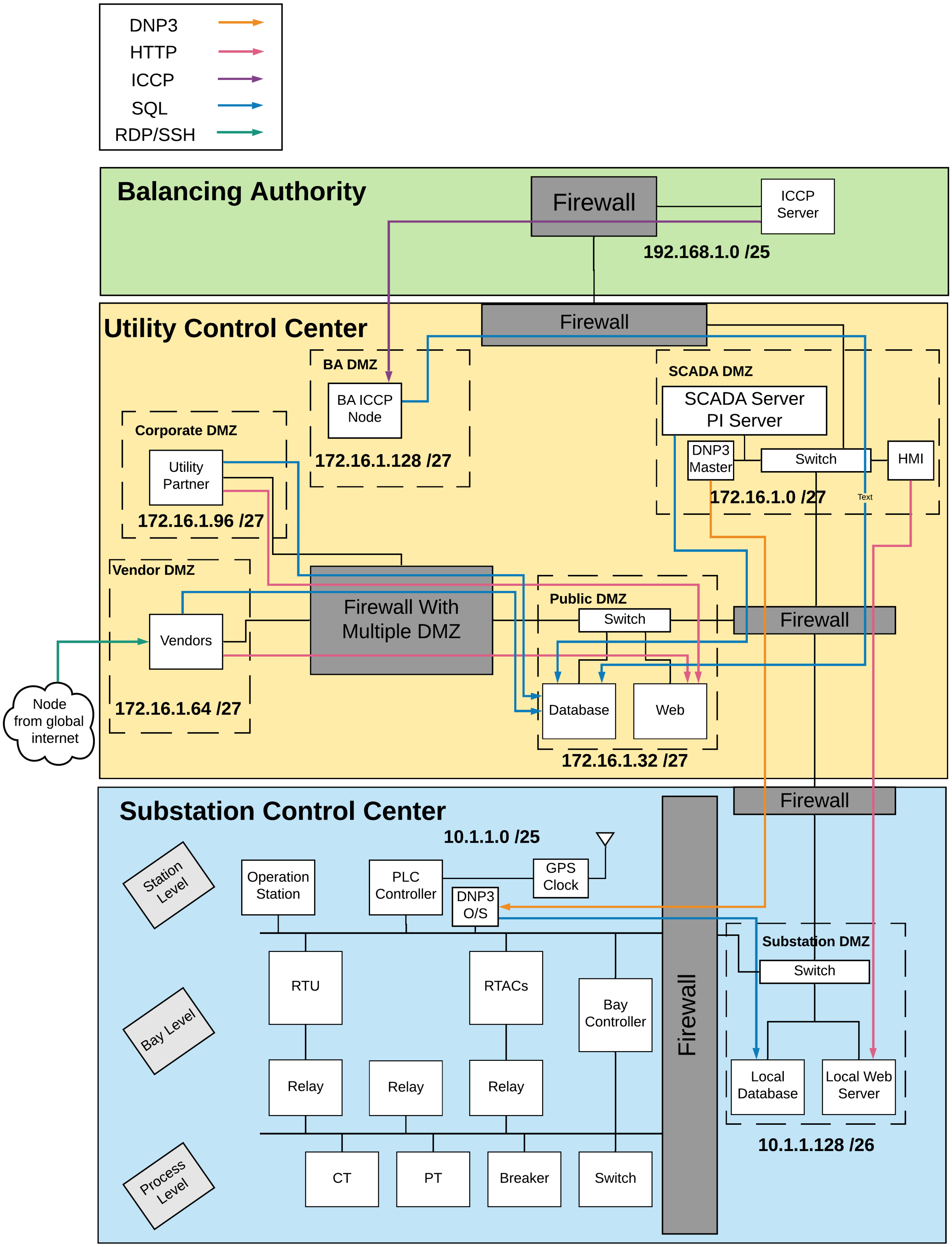


Figure 2.1: Data Flows Within Substation, Utility Control Center and Balancing Authority [24]

port 20000 is used over TCP for DNP3. This creates the SCADA object-group, which is used between the DNP3 Master in the UCC and the relays in the bay level at the substation. This allows the SCADA server to control the relays and get telemetry data at any given time. The DNP3 Master in the UCC will initiate the connection and access the DNP3 Outstation (DNP3 O/S) in the substation. The DNP3 O/S will have the current data sent from the Remote Terminal Unit (RTU) that the UCC needs to monitor. A Remote Terminal Access Controller (RTAC) is also found at the substation, which can control the relays locally. The relays can for instance trip a circuit breaker to isolate a faulty circuit. Other devices shown at the substation process level are transformers and switches.

2.1.2 Web-based Protocols

The next data flow between the substation and UCC is derived from Web-based protocols. They are used to access data in the smart grid network over HyperText Transfer Protocol (HTTP) and HTTP over Transport Layer Security (HTTPS). These two protocols run over TCP, and use ports 80 or 8080 for HTTP, and ports 443 or 8443 for HTTPS. In the UCC, these protocols are used by the Human Machine Interface (HMI) node to access the local substation data in the local web server. Within the UCC, vendors will also need to access the web server located in the public DMZ from their own vendor DMZ.

2.1.3 Remote Access Protocols

Contractors or vendors access the UCC from an outside node in the global Internet to the vendor DMZ, which includes a dedicated machine for vendors to access. Secure Shell (SSH) and Remote Desktop (RDP) protocols are used for vendors to remotely access the Vendor DMZ, and from there obtain information from the database and web servers in the public DMZ.

2.1.4 Database Protocol

In this model, Structured Query Language (SQL) is the database protocol used by the UCC to upload or retrieve data from PI servers [21], also known as historian servers, which archive organized data from the substations and allow operators to perform data analytics. In the substation, the DNP3 O/S pushes the current data received from the RTU to a local database in the substation. The purpose of this is to store a backup of the recent data, which can be accessed by the UCC to see a history of data from the relays. The main database is located in the UCC, as the PI server in the SCADA DMZ. A copy of this database is also available at the Public DMZ, for vendors and corporate users to access.

2.1.5 ICCP Protocol

At the top of Figure 2.1, is the BA which oversees several utilities connected to the power grid and manages the deregulated energy market. The Inter-Control Center Protocol (ICCP) is a protocol that has been developed internationally for use in energy networks to transfer various types of data including both historical and current data. The BA uses ICCP to communicate with the UCC via an ICCP server in the BA that accesses an ICCP node in the BA DMZ to pull data about that utility.

2.2 Basic Network Topology

The network topology used for this project is shown in Figure 2.2, which was designed based off of the data flow diagram in Figure 2.1. This network diagram was created using Cisco Packet Tracer, to test if the network included all necessary components for the smart grid, and to test the firewall rules that Cisco Packet Tracer is able to test. This allows for testing of the HTTP protocol by trying to access a web server, and for testing that other protocols such as Internet Control Message Protocol (ICMP) are blocked by trying to ping a device. It includes the model of the UCC connected to the BA through a serial link, and the UCC connected to the substation through another serial link.

Beginning at the bottom of the network topology, the substation network includes one firewall which divides the substation into two subnets, both with a high security level. One subnet is for the relay network which sends all power information back to the UCC, and the other is for the substation DMZ, which includes a local database and web server, that the UCC can access to pull substation data history.

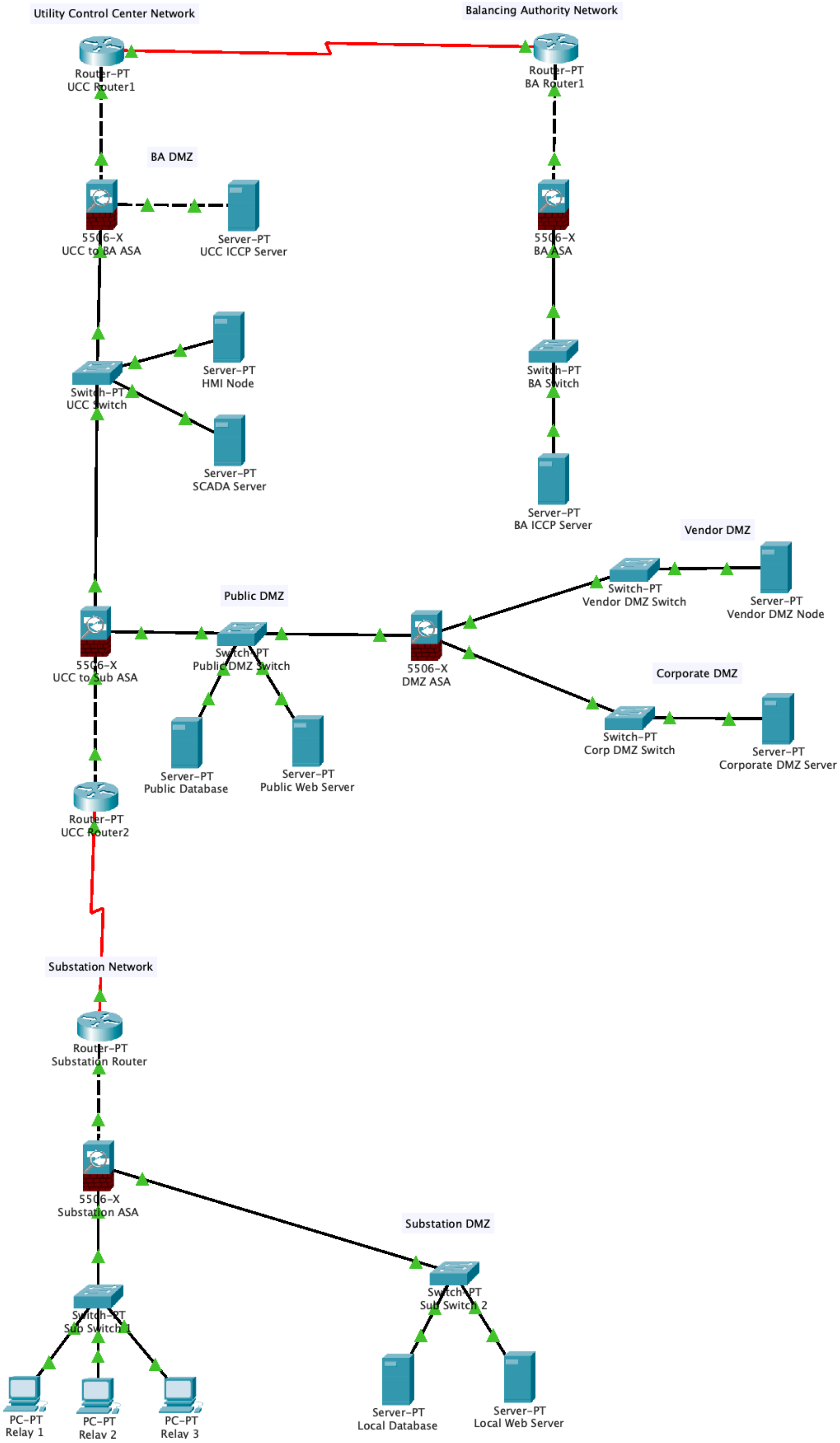


Figure 2.2: Comprehensive Network Topology of a Smart Grid Network [24]

Moving up there is the UCC Network, which includes three firewalls and two routers. Following the data flows in Figure 2.1, the UCC houses five DMZ's for the:

1. SCADA network
2. Corporate Network
3. Balancing Authority (BA)
4. Public network
5. Vendors to access the public network

First, there is a DMZ for the BA to access and control the UCC using ICCP. The ICCP server in the UCC is protected behind one interface of a firewall. The other two interfaces guard the inside and outside of the control center network respectfully. Next, the SCADA server and HMI node are protected between the inside interface of the previously mentioned firewall, and another firewall. The corporate and vendor DMZ's are protected by two different interfaces of a shared firewall, and that same firewall is connected to one end of the public DMZ. This ensures that the main control center can access the public DMZ from one side and both vendors and corporate can access it from the other side. On the other side of the public DMZ is a firewall also connected to the inside of UCC and the substation.

At the top right of the network topology in Figure 2.2 is the BA network. This network only has one firewall, which protects the BA's ICCP Server.

The IP addresses assigned to this model network are shown below in Tables 2.1, 2.2, and 2.3. To be more concise, the point-to-point links in the UCC are not shown, only the major subnets. For the Texas 2000 model, each substation will have the address space of 10.1.X.0, X representing the substation number. Since there are about 1250 substations in the Texas 2000 model, some addresses will begin with 10.5.X.0. Each UCC will have the address space of 172.16.X.0, with X representing the UCC number. Finally, each BA will have the address space of 192.168.X.0, with

Table 2.1: IP Allocation in Substation [24]

Description	Subnet	Subnet Mask	Available IP addresses
Relay Network	10.1.1.0	/25	126
Substation DMZ Network	10.1.1.128	/26	62
Firewall to Router Link	10.1.1.192	/30	62

Table 2.2: IP Allocation in Utility Control Center [24]

Description	Subnet	Subnet Mask	Available IP addresses
Control Center Servers	172.16.1.0	/27	30
Public DMZ	172.16.1.32	/27	30
Vendor DMZ	172.16.1.64	/27	30
Corporate DMZ	172.16.1.96	/27	30
ICCP DMZ for BA	172.16.1.128	/27	30

X representing the UCC it is connected to. Although in Texas there is only one BA (e.g., ERCOT, or Electric Reliability Council of Texas), other locations may have more than one BA.

In summary, the essential components of the smart grid (substation, utility control center, and balancing authority) with its main data flows and protocols was described in this chapter. Additionally, a simulation model was created with Cisco Packet Tracer to evaluate this model and configure the firewall rules, which will be described in detail in the next chapter.

Table 2.3: IP Allocation in Balancing Authority [24]

Description	Subnet	Subnet Mask	Available IP addresses
ICCP Server Network	192.168.1.0	/25	126
Firewall to Router Link	192.168.1.128	/30	2
BA Router to UCC Router Link	192.168.1.132	/30	2

3. FIREWALL CONFIGURATION[*]

Firewalls are network devices that monitor and inspect incoming and outgoing traffic, therefore providing a layer of defense between networks. They are Open Systems Interconnection (OSI) Layer 3 devices capable of using IP addresses and routing commands. This allows the firewall to forward or discard IP packets after matching them against the established rules configured onto that firewall. For this research, Cisco Adaptive Security Appliances (ASA’s) were used, which have a special configuration called a security level [20]. This means that each interface can be set to have a numerical level from 0-100 that determines its security hierarchy in the network. Incoming packets cannot pass from lower security zones to higher security zones without special rules. Typically, the inside of a network is set to a security level of 100, and the outside is set to a level of security level of 0. Furthermore, DMZ’s can be created with a typical security level of 50. This means that the main or inside network can access the DMZ to push or pull data from a server inside that DMZ for an outside node to access. That outside node would not be able to gain entry to the inside network however, since they would be in a lower security zone.

3.1 Cisco ASA Configuration - Substation Firewall

To implement and test this model, firewalls and routers were first configured in Cisco Packet Tracer then tested in NP-View. The model, which is based on the network topology and IP address configuration discussed in the previous chapter (Fig. 2.2), has five firewalls and four routers. A typical firewall configuration includes three main components: interface configuration, object-groups, and access control lists. To illustrate each component, the configuration shown next is the configuration of the firewall in the substation (*asaSub*), followed by an explanation of each component.

```
ASA Version 9.6(1)
```

[*] Reprinted with permission from Firewall Configuration and Path Analysis by Authors Names, 2020. IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR), pp. 1-6, l’ 2020 IEEE

```
!
hostname asaSub
names
!
interface GigabitEthernet1/1
nameif inside
security-level 100
ip address 10.1.1.1 255.255.255.128
!
interface GigabitEthernet1/2
nameif outside
security-level 0
ip address 10.1.1.194 255.255.255.252
!
interface GigabitEthernet1/3
nameif dmz
security-level 50
ip address 10.1.1.129 255.255.255.192
!
!
object-group service HTTP tcp
port-object eq www
port-object eq 443
port-object eq 8080
port-object eq 8443
!
object-group service ICCP tcp
port-object eq 102
!
```

```
object-group service SCADA tcp

port-object eq 20000

!

object-group service SQL tcp

port-object eq 1433

!

object-group service remote_access tcp

port-object eq 22

port-object eq 3389

!

object-group service SNMP udp

port-object eq 161

port-object eq 162

!

object-group network Local_Database

network-object host 10.1.1.130

!

object-group network Local_Web_Server

network-object host 10.1.1.131

!

object-group network Relays

network-object host 10.1.1.2

network-object host 10.1.1.3

network-object host 10.1.1.4

!

object-group network SCADA_Server

network-object host 172.16.1.3

!

object-group network HMI_Node
```

```
network-object host 172.16.1.4
!
object-group network UCC_ICCP_Server
network-object host 172.16.1.130
!
object-group network BA_ICCP_Server
network-object host 192.168.1.2
!
object-group network Public_Database
network-object host 172.16.1.35
!
object-group network Public_Web_Server
network-object host 172.16.1.36
!
object-group network Vendor_Node
network-object host 172.16.1.66
!
object-group network Corp_Server
network-object host 172.16.1.97
!
!
!
route outside 0.0.0.0 0.0.0.0 10.1.1.193 1
!
!
access-list from_inside remark ********* from_inside ACL *********
access-list from_inside extended permit tcp object-group Relays object-group Local_Database object-group SQL
access-list from_inside extended deny ip any any
```

```
!
access-list from_outside remark ********* from_outside ACL *********
access-list from_outside extended permit tcp object-group SCADA_Server object-group Relays object-group SCADA
access-list from_outside extended permit tcp object-group HMI_Node object-group Local_Web_Server object-group HTTP
access-list from_outside extended deny ip any any
!
access-list from_dmz remark ********* from_dmz ACL *********
access-list from_dmz extended deny ip any any
!
!
access-group from_inside in interface inside
access-group from_outside in interface outside
access-group from_dmz in interface dmz
!
!
!
telnet timeout 5
ssh timeout 5
```

3.1.1 Interface Configuration

To begin the firewall configuration, IP addresses were first configured on each device, including all servers and substation relays, according to the IP addresses assigned in Figure 2.1. Then, the security-levels were configured on each interface of each firewall. The interfaces protecting the inside network of the firewall were assigned a level of 100, the outside interfaces were assigned a level of 0, and the DMZ interfaces were assigned a level of 50.

In the firewall configuration for the substation firewall (*asaSub*) in Section 3.1, interface GigabitEthernet1/1 is part of the subnet with the relays and therefore is the inside network, with a security level of 100. The IP address is defined as the first address in the 10.1.1.0 subnet with a subnet mask of 255.255.255.128 (/25). The address assigned is 10.1.1.1. Interface GigabitEthernet1/2 represents the outside network, which is connected to the control center network, and therefore outside of the substation network. This security-level is set to 0, so that traffic is permitted to leave the network, but not to enter the network. The IP address for this interface is set to follow the model and represents a point-to-point link between the firewall and router leaving the substation network. Lastly, the GigabitEthernet1/3 interface represents the DMZ in the substation, including the local web server and database. This interface is set to a security level of 50 so that data can pass into it from the relays, but cannot be accessed from outside of the substation. The IP address is also set to be the first address in this subnet. The subnet address is 10.1.1.128 with a subnet mask of 255.255.255.192 (/26).

3.1.2 Object-Groups

Next, object-groups were created and copied into each firewall configuration in order to save processing time and error by grouping nodes or ports together that would be used for the same application. Object-groups can be created to include each network host that needed to be accessed at the same time, and for each application and protocol that would need to be used at the same time as well. The former are network object-groups which allow one specific host or set of hosts to be called by a specified keyword when creating the rules for network access. Service object-groups allow specific port numbers to be grouped and named for easier access [22]. For instance, an "HTTP" service object-group was created which included ports 80,443, 8080, and 8443. This allowed four configuration lines to be condensed into just one rule. Furthermore if those ports needed to be accessed by two different nodes for instance, it would have been eight configuration lines instead of just two lines with the use of object-groups.

For this model, eleven network-objects were created, and six service-object groups were created for the different data flows defined in Figure 2.1. The firewall configuration for the substation firewall in Section 3.1 shows these seventeen total object-groups. These object-groups were then copied over onto each firewall for easier access and referencing.

The model in Figure 2.1 shows five data flow types, which are the first five service object-groups. The first is for HTTP, which includes ports 80, 443, 8080, and 8443, over TCP. Port 80 is represented by the "www" keyboard, by the Cisco firewall. Next, an object-group was created for the ICCP service. This group only includes one port number: port 102, which is used over TCP. The third service group created was for SCADA, which consists of port 20000, which is the port used by industry for DNP3 over TCP. Next, SQL was defined as port 1433, also used over TCP. The fifth group created for the last data flow was for remote access groups. This object-group includes ports 22 and 3389, which are the ports used for SSH and RDP, respectively. These ports also use TCP.

A sixth service object-group was then defined for the Simple Network Management Protocol (SNMP). This data flow was not defined by the model, but would be important in industry. This application is used by the control center to communicate with devices in the substation such as the relays, to check on their status. This application sends Protocol Data Units (PDU's) and waits for a reply, to make sure each device is running and stable. Table 3.1 shows a list of each object group along with the application port numbers and protocol used.

Table 3.1: Object Groups and Port Numbers

Object Group	**Protocol**	**Port Number(s)**
HTTP	TCP	80, 443, 8080, 8443
ICCP	TCP	102
SCADA	TCP	20000
SQL	TCP	1433
Remote Access	TCP	22, 3389
SNMP	UDP	161, 162

3.1.3 Access-Control Lists

The next step was to configure the rulesets to define who can access each device in the network and how, using access-control lists (ACL's) [23]. There are two main types of ACL's, standard and extended. Standard ACL's check the source address of the packet only, whereas extended ACL's match both the source and destination addresses on the packet, as well as the protocol and application port number. The ACL is also configured onto one interface, and directed to match either incoming or outgoing packets. There can only be one ACL per interface, per direction. For this research, a set of extended ACL's were configured onto each firewall in the model. At the end of each ACL, a deny all statement must be written according to NERC's CIP-005 requirement [1] to ensure that all other packets attempting to access the network are discarded. The ACL's are created using applications over TCP, and so are state-aware ASA's. This means that each rule only needs to be defined on one interface, and in one direction, as the firewall will allow the return packet to pass back through the device without another rule on the opposite interface. This allows for a connection to only be initiated in one direction, but still allow for a reply. Lastly, static routes are also written onto the firewalls, which are formatted to direct any packet out of a certain interface, to be matched against the configured ACL's as a default path.

Beginning with the firewall configuration in the substation, there is a ruleset of ACL's configured onto each of the three interfaces. The first is the inside interface, which secures the main network housing the relays and other important elements of the substation network. According to Figure 2.1, there is only one data flow that needs to be permitted out the inside network, which is for the information from the relays to be moved out into the substation DMZ, and stored on a local database. This traffic would be carried over SQL, and so the ACL is written so that the "Relays" object-group is allowed to access the "Local_Database" object-group, over the "SQL object-group. Next, the outside interface maintains two data flows that need to be permitted into the substation. One is for the SCADA server to access the relays via the DNP3 O/S over DNP3, and the other is for the HMI node in the control center to access the local web server in the substation DMZ over HTTP. These rules were also written with the appropriate network and service object-groups.

The last interface protects the substation DMZ, and does not need any ACL's except for a deny all statement, since now data flows should be permitted out of the substation DMZ and into the rest of the network. This allows an outside node to possibly access that DMZ, but not breach the rest of the network or control center.

After defining each of the rules that make up the ACL, the ACL must be bound to an interface of the firewall. In these configurations for the model, the ACL's named "from_inside" are bound to the inside interface with an inbound direction. This allows all packets coming into the firewall at the inside interface to be matched against the defined rules. All ACL's named "from_outside" are bound to the outside interface of the firewall with an inbound direction as well. This allows all packets coming into the network through the outside interface to be examined and filtered. Furthermore, all ACL's named "from_DMZ" are bound to the interface protecting the DMZ and are also defined in an inbound direction.

Lastly, the static route configured onto this firewall forces all traffic going outside of the network to the outside interface by specifying the next hop, in this case, the IP address of the neighboring router. First the interface name is specified, then the destination address and mask, and lastly the next hop IP address. The static route for the substation firewall is configured as:

```
route outside 0.0.0.0 0.0.0.0 10.1.1.193 1
```

3.2 Firewalls in the Utility Control Center

In the Utility Control Center there are three firewalls, one at each entrance to the network from the substation and balancing authority respectively, and one for the main DMZ's in the network.

3.2.1 Control Center and Substation Firewall

The first firewall at the UCC controls all access points between the control center and the substation. This firewall also has three interfaces configured to have interfaces, one for inside the control center, one for outside the control center connecting the substation, and one for the public DMZ network. The following configuration shows the configuration for the ACL's on the firewall between the UCC and the substation. Here only the static routes and ACL's are shown.

```
route inside 172.16.1.130 255.255.255.224 172.16.1.1 1
route outside 0.0.0.0 0.0.0.0 172.16.1.166 1
!
!
access-list from_inside remark ********* from_inside ACL *********
access-list from_inside extended permit tcp object-group SCADA_Server object-group Relays object-group SCADA
access-list from_inside extended permit tcp object-group HMI_Node object-group Local_Web_Server object-group HTTP
access-list from_inside extended permit tcp object-group UCC_ICCP_Server object-group Public_Database object-group SQL
access-list from_inside extended permit tcp object-group SCADA_Server object-group Public_Database object-group SQL
access-list from_inside extended deny ip any any
!
access-list from_dmz remark ********* from_dmz ACL *********
access-list from_dmz extended deny ip any any
!
access-list from_outside remark ********* from_outside ACL *********
access-list from_outside extended deny ip any any
!
!
access-group from_inside in interface inside
access-group from_dmz in interface dmz
access-group from_outside in interface outside
```

In order to provide the most security, there should not be any nodes from the substation or public DMZ allowed to initiate a connection with any node inside the control center. To enforce this, ACL's are configured onto the inside interface of the firewall, to allow outgoing connections only. The first flow allowed is for the SCADA server to pull data from the relays over DNP3. Next, the HMI Node in the control center must be able to access the local web server in the substation. Finally, both the UCC's ICCP Server and the SCADA server would also be able to pull data from the database in the Public DMZ over SQL. The other two interfaces have only one ACL written to each, which is a statement to deny all packets that are incoming to either interface. Lastly, two static routes are configured, one to direct traffic inside the network from a specific source address, and the other is to direct all remaining traffic outside the network as a default. The static route directing traffic inside the network has a source ip address of the address space for the substation.

3.2.2 Control Center and DMZ's Firewall

The second firewall in the UCC is to separate the DMZ's. The ACL configuration for this firewall is shown next.

```
   access-list from_public_dmz remark ********* from_public_dmz ACL *********
   access-list from_public_dmz extended deny ip any any
   !
   access-list from_vendor_dmz remark ********* from_vendor_dmz ACL *********
   access-list from_vendor_dmz extended permit tcp object-group Vendor_Node
object-group Public_Database object-group SQL
   access-list from_vendor_dmz extended permit tcp object-group Vendor_Node
object-group Public_Web_Server object-group HTTP
   access-list from_vendor_dmz extended deny ip any any
   !
   access-list from_corp_dmz remark ********* from_corp_dmz ACL *********
   access-list from_corp_dmz extended permit tcp object-group Corp_Server
object-group Public_Database object-group SQL
   access-list from_corp_dmz extended permit tcp object-group Corp_Server
object-group Public_Web_Server object-group HTTP
   access-list from_corp_dmz extended deny ip any any
   !
   !
   access-group from_public_dmz in interface public_dmz
   access-group from_vendor_dmz in interface vendor_dmz
   access-group from_corp_dmz in interface corp_dmz
```

The configuration for this firewall has a ruleset for the public DMZ, vendor DMZ, and the corporate DMZ. The ruleset for the interface of the public DMZ has only one ACL, which is to block all incoming packets. This is because no node should be able to go through the public DMZ and gain access to the main network, or any other DMZ. Next, the interface connecting the vendor DMZ has two allowed data flows. One is for a node from the vendor DMZ to be able to access the public database in the public DMZ, over SQL. The other is also for a node from the vendor DMZ to be able to access the public web server in the public DMZ, over HTTP. The third interface is to separate the corporate DMZ from the rest of the control center. This firewall permits the same two dataflows to enter from the corporate DMZ as it did from the vendor DMZ. One is for the corporate server to access the public database in the public DMZ over SQL, and one is for the corporate server to access the public web server over HTTP.

3.2.3 Control Center and BA Firewall

The last firewall in the UCC separates the control center from the BA. This firewall configuration is shown next, beginning with static routes and followed by the ACL configuration.

```
   route outside 192.168.1.2 255.255.255.128 172.16.1.161 1
   route inside 172.16.1.35 255.255.255.224 172.16.1.2 1
   !
   !
   access-list from_dmz remark ********* from_dmz ACL *********
   access-list from_dmz extended permit tcp object-group UCC_ICCP_Server
object-group Public_Database object-group SQL
   access-list from_dmz extended deny ip any any
   !
   access-list from_inside remark ********* from_inside ACL *********
   access-list from_inside extended deny ip any any
   !
   access-list from_outside remark ********* from_outside ACL *********
   access-list from_outside extended permit tcp object-group BA_ICCP_Server
object-group UCC_ICCP_Server object-group ICCP
   access-list from_outside extended deny ip any any
   !
   !
   access-group from_dmz in interface dmz
   access-group from_inside in interface dmz
   access-group from_outside in interface outside
```

This firewall also has three interfaces, one to protect the inside of the control center and main SCADA server, one to defend from nodes outside of the network, and one to protect the ICCP Server inside of a DMZ. The interface for the this DMZ has one ACL to allow the ICCP server in the UCC to communicate with the public database in the public DMZ over SQL. All other flows are blocked that are initiated from this DMZ. The inside interface including the SCADA server has one statement to block all traffic, since no traffic is allowed to be initiated from this DMZ and forwarded towards the BA. Lastly, the outside interface allows one data flow, which is for the BA's ICCP server to initiate connections with the UCC's ICCP Server. Lastly the two static routes are configured onto the firewall to route traffic for the inside interface and for the outside interface. All traffic with a source address from the BA's subnet is routed outside the network, and all traffic with a source address from the UCC's subnet is routed inside the network.

3.2.4 BA Firewall

The last firewall presented in this model is the firewall in the BA, which secures the BA's ICCP server. The configuration code for this ASA is shown next.

```
  route outside 0.0.0.0 0.0.0.0 192.168.1.129 1
  !
  access-list from_inside remark ********* from_inside ACL *********
  access-list from_inside extended permit tcp object-group BA_ICCP_Server
object-group UCC_ICCP_Server object-group ICCP
  access-list from_inside extended deny ip any any
  !
  access-list from_outside remark ********* from_outside ACL *********
  access-list from_outside extended deny ip any any
  !
  !
  access-group from_inside in interface inside
  access-group from_outside in interface outside
```

The firewall in the BA only has two interfaces, one to protect the inside of the BA and one to defend the outside of the BA network. The inside interface has one rule to allow the ICCP server inside the BA network to initiate a connection with the ICCP server in the UCC, and all other traffic is blocked. The other interface has only one rule configured to block all traffic through that interface. Lastly one static route is configured to route all traffic outside of the network by specifying the next hop towards the control center.

3.3 Router Configurations

After all five firewalls were configured according to the data flow model, four routers were configured. For each router IP addresses were configured onto the appropriate interfaces for point-to-point links, and static routes were implemented to forward packets automatically to the appropriate firewall.

3.4 Quantitative analysis of firewall configurations

One way to optimize firewall configurations is to compare quantifiable data about each ruleset. The number of used object-groups in each firewall are shown in Table 3.2, broken down by the object-group type. This number was minimized as much as possible, because less possible access-points creates a more secure network.

Table 3.2: Firewall Object-Groups

Firewall Location	Network Object-Groups	Service Object-Groups
Substation ASA	5	3
UCC to Sub ASA	6	3
UCC DMZ ASA	4	2
UCC to BA ASA	3	2
BA ASA	2	1

Table 3.3 shows the properties of the ACL's configured on the five ASA firewalls. This shows a summary of the number of rules required on each firewall to allow the specified data flows and block all other traffic in order to secure the network.

Table 3.3: Firewall ACL Properties [24]

Firewall Location	Number of Rules	Number of Interfaces
Substation ASA	6	3
UCC to Sub ASA	7	3
UCC DMZ ASA	7	3
UCC to BA ASA	5	3
BA ASA	3	2

These firewall configurations were created using Cisco Packet Tracer, and tested in the following chapter with the NP-View tool.

4. PATH ANALYSIS AND RISK ASSESSMENT

One tool for analyzing a network and ensuring the lowest number of paths into the network is Network Perception's NP-View software [16]. NP-View allows for a static path analysis by reading a number of firewall or router files, and automatically builds a network topology based on the provided IP addresses and subnets. It can accept configuration files for any OSI Layer 3 network device, and learns how each IP address is connected based on configured interfaces and object-groups.

4.1 NP-View Analysis

After creating the firewall configurations for this model as described in Chapter 3, the Cisco ASA configuration files were input into a project in NP-View where a path analysis was completed. Figure 4.1 shows the network model that NP-View built based on only these configuration files. The lower half beneath the router named “routerSub" represents the substation network, the upper part of the network directly connected underneath the router named “routerBA" represents the BA, and the rest of the network represents the UCC. The cloud shapes in the figure that are named “inside," “outside," or “dmz," represent the border gateway interface for each firewall in Figure 4.1. The nodes are then shown within each of the interfaces represented by the gray dots in the model. For instance, the three nodes inside of the inside interface on “asaSub" named “.2," “.3," and “.4," represent the three relays of the substation. This model is beneficial to see how each node can be accessed, and through which interfaces on each firewall.

4.1.1 Path Analysis

A path analysis was then run on the network, and the NP-View software determined every possible way to access every node in the network, and the criticality level of each path. In this model, fifty-six possible paths were found first, before the next stage of optimizing. These paths were then reviewed carefully to see if each was necessary and this number was gradually reduced to limit the permissible access points as much as possible. It was noted that some of these paths were bi-directional at this time. This meant that while the SCADA server could initiate a connection with the relays, the relays could also initiate a connection with the SCADA server, which was a big vulnerability. These insecure paths were eliminated, and finally only thirty-three possible paths were found, which were then reviewed and marked to be okay, low-risk, or high-risk. Figure 4.2 shows the highlighted network in NP-View after the path analysis was completed. Any blue node can initiate a connection, and any red node can receive a connection. If the node is half red and half blue, then it can both initiate and receive connections.

The path analysis can also be filtered to only show every incoming or outgoing traffic to a specific node in the network. For instance, the SCADA server shows only two outgoing paths which are: to the public database in the public DMZ over SQL, and to the relays in the substation over DNP3. There are no incoming paths; therefore, this verifies that the rule sets in the model network are secure, as no one from outside the control center can initiate a connection with the SCADA server. Figure 4.3 shows a filtered path analysis from NP-View of the incoming and outgoing paths to the SCADA server. The SCADA server is still blue, meaning that it can only initiate connections. It can establish a connection with the three relays in the substation, and the public database in the public DMZ.

4.1.2 Rule Audit

Next, a rule audit was run on the network which shows a table of the complete ruleset for each firewall, for the finalized firewall configurations. NP-View parses each firewall configuration and separates the ACL's from the rest of the configuration, to form the rule audit. The rule audit then shows the device where the rule is configured, the line number from the file, the name of the ACL where the rule is set, and the details of the rule itself. This includes the source address, destination address, application port number, and whether the packet should be permitted or denied. Next there is a column for parsing error if one occurred. Finally, the rule audit shows any risk alerts that the tool has found. In this model there were only four criticality risk alerts on the fully optimized firewall files. This alert occurred because the HTTP object group had four different port numbers: 80, 443, 8080, and 8443. However, these four ports are necessary for both HTTP and HTTPS connections, so this was a low-risk alert. Table 4.1 shows these four criticality alerts. Figurs 4.4 and 4.5 show the results of the rule audit in NP-View.

Table 4.1: Criticality Report

Risk	**Criticality**	**Description**
Risk	Low	[asaDMZ] line 132: Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt
Risk	Low	[asaDMZ] line 127: Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt
Risk	Low	[asaSub] line 131: Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt
Risk	Low	[asaUCCtoSub] line 123: Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt

The rule audit was useful in eliminating extra paths that could be exploited by an attacker. It also flags all rules that use open ports or allows any IP address for the source or destination addresses. These paths are not specific for a permitted data flow, and are therefore a vulnerability.

NP-View is a very beneficial software tool for analyzing many network types across many industries. It was necessary to run this tool over many iterations in order to minimize the open paths and ports in the network. The path analysis conducted on this model was published in a paper for the IEEE Communications, Quality, and Reliability (CQR) conference in May 2020 [24].

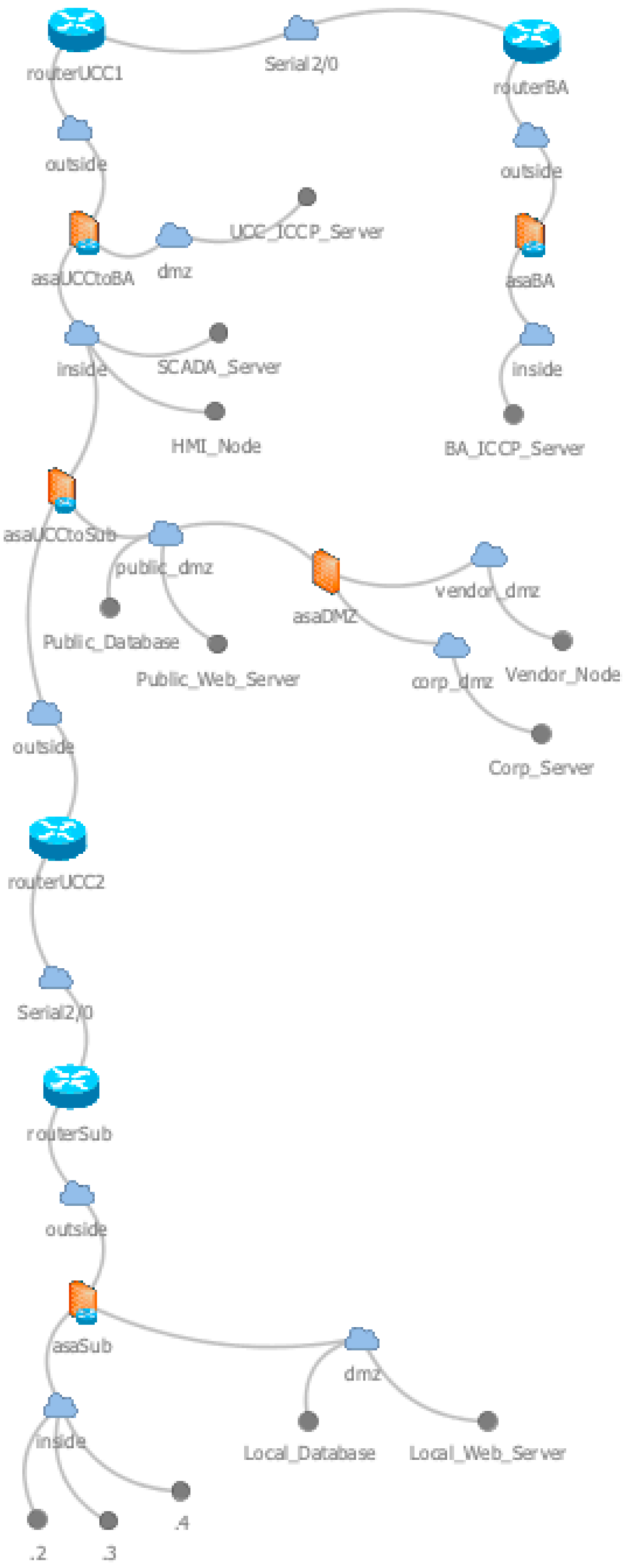


Figure 4.1: NP-View diagram.

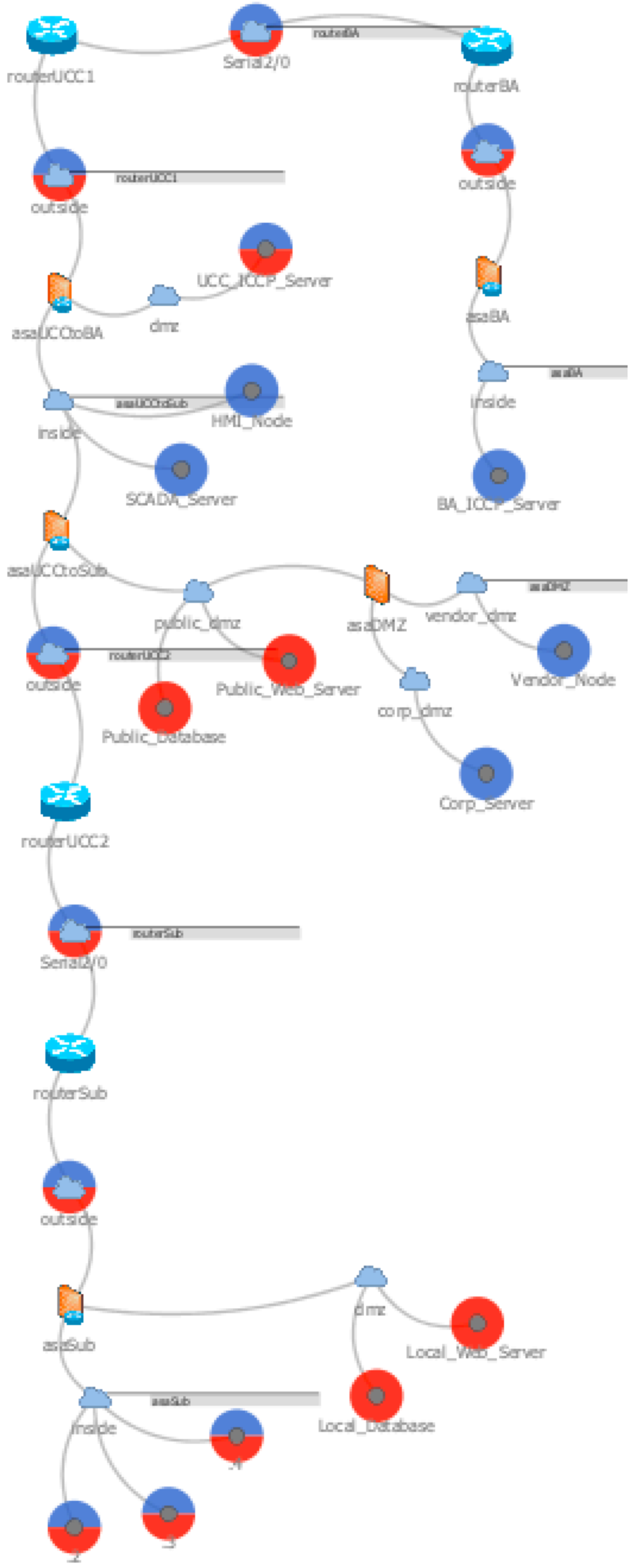


Figure 4.2: NP-View path analysis highlighting source and destination nodes.

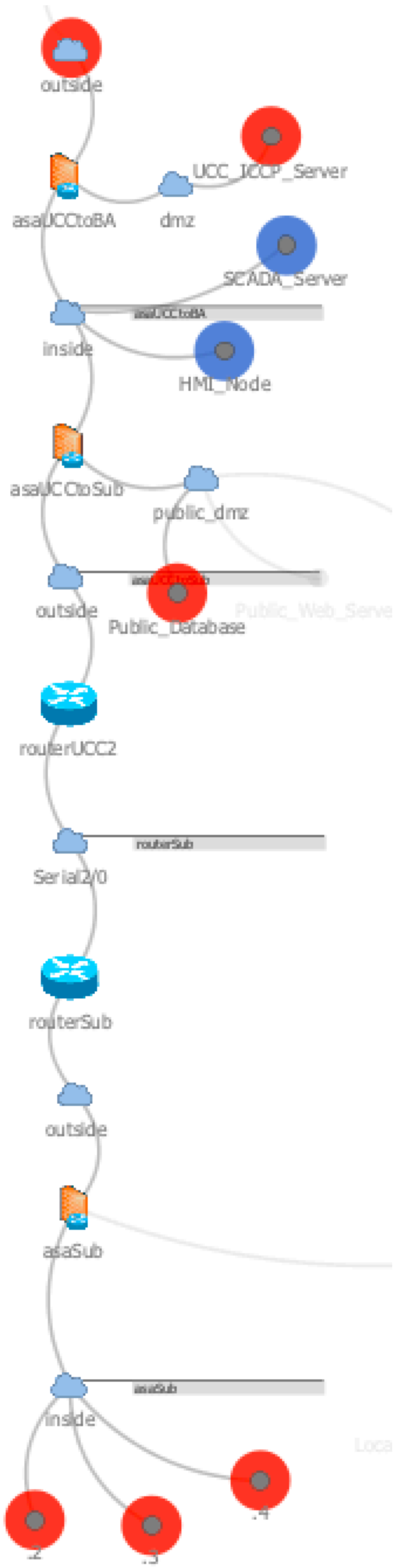
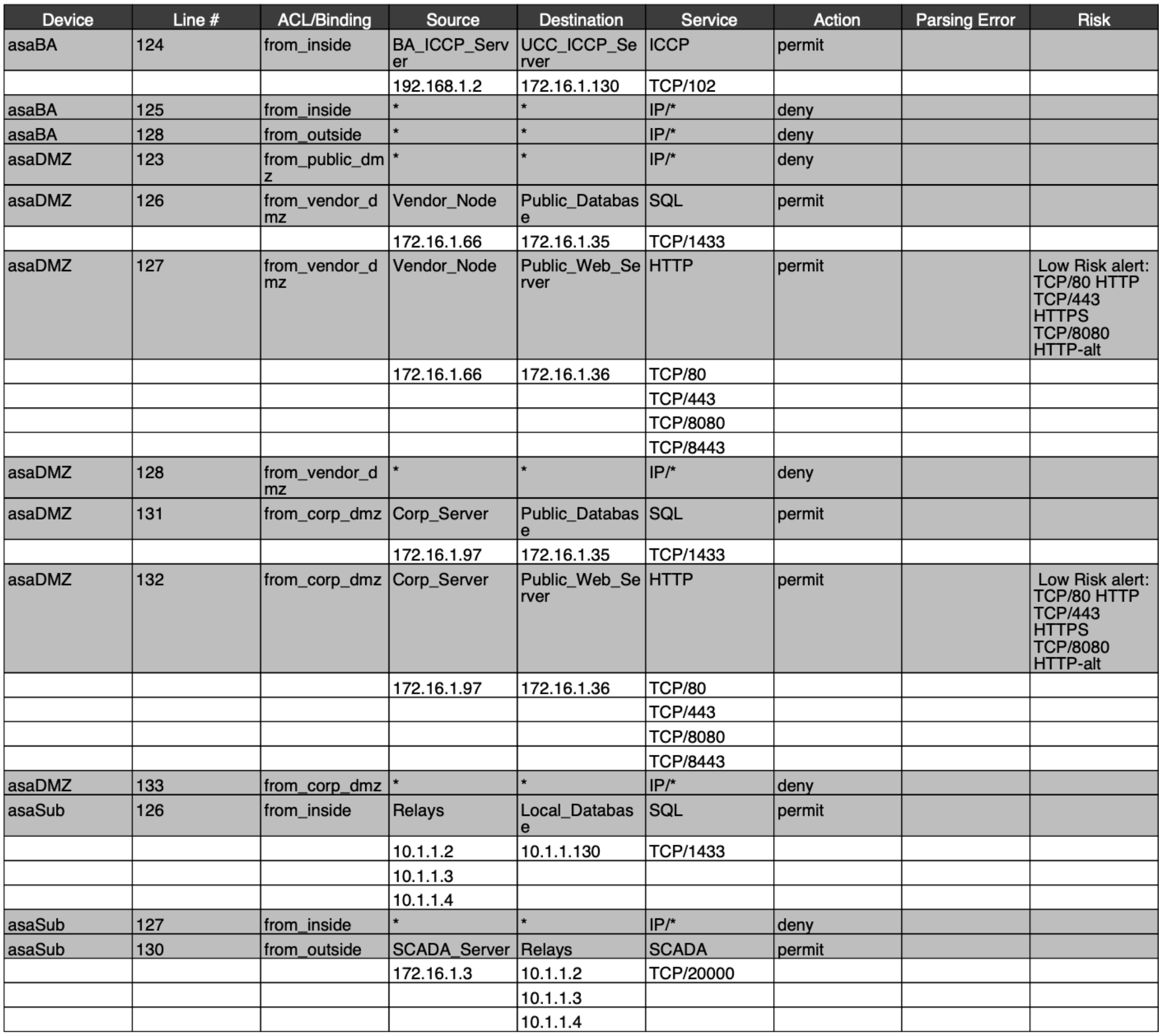
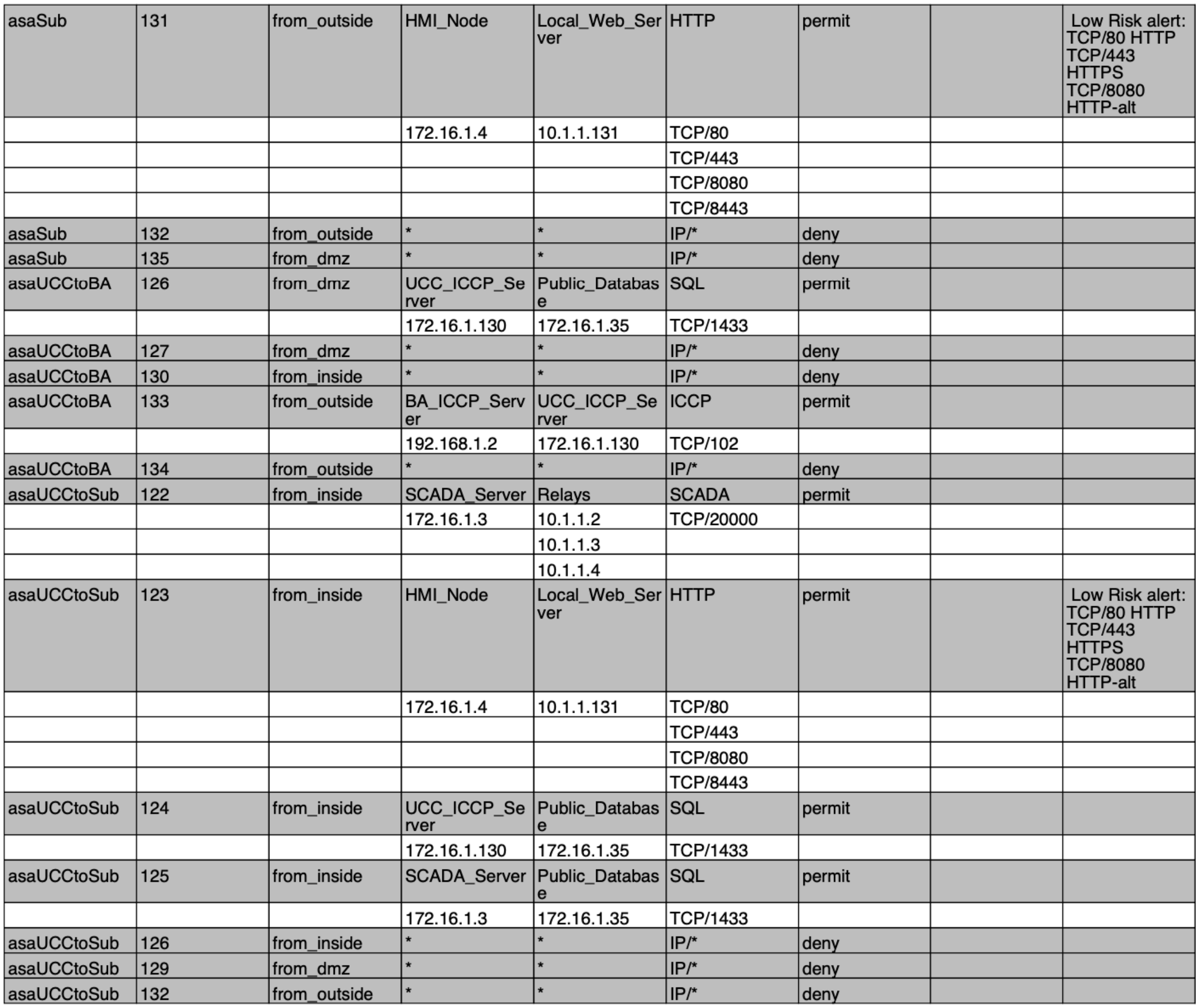


Figure 4.3: Path Analysis Filtering Incoming and Outgoing Paths for SCADA Server



Device	Line #	ACL/Binding	Source	Destination	Service	Action	Parsing Error	Risk
asaBA	124	from_inside	BA_ICCP_Server	UCC_ICCP_Server	ICCP	permit		
			192.168.1.2	172.16.1.130	TCP/102			
asaBA	125	from_inside	*	*	IP/*	deny		
asaBA	128	from_outside	*	*	IP/*	deny		
asaDMZ	123	from_public_dmz	*	*	IP/*	deny		
asaDMZ	126	from_vendor_dmz	Vendor_Node	Public_Database	SQL	permit		
			172.16.1.66	172.16.1.35	TCP/1433			
asaDMZ	127	from_vendor_dmz	Vendor_Node	Public_Web_Server	HTTP	permit		Low Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt
			172.16.1.66	172.16.1.36	TCP/80			
					TCP/443			
					TCP/8080			
					TCP/8443			
asaDMZ	128	from_vendor_dmz	*	*	IP/*	deny		
asaDMZ	131	from_corp_dmz	Corp_Server	Public_Database	SQL	permit		
			172.16.1.97	172.16.1.35	TCP/1433			
asaDMZ	132	from_corp_dmz	Corp_Server	Public_Web_Server	HTTP	permit		Low Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt
			172.16.1.97	172.16.1.36	TCP/80			
					TCP/443			
					TCP/8080			
					TCP/8443			
asaDMZ	133	from_corp_dmz	*	*	IP/*	deny		
asaSub	126	from_inside	Relays	Local_Database	SQL	permit		
			10.1.1.2	10.1.1.130	TCP/1433			
			10.1.1.3					
			10.1.1.4					
asaSub	127	from_inside	*	*	IP/*	deny		
asaSub	130	from_outside	SCADA_Server	Relays	SCADA	permit		
			172.16.1.3	10.1.1.2	TCP/20000			
				10.1.1.3				
				10.1.1.4				

Figure 4.4: Rule Audit Part 1



asaSub	131	from_outside	HMI_Node	Local_Web_Server	HTTP	permit		Low Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt
			172.16.1.4	10.1.1.131	TCP/80			
					TCP/443			
					TCP/8080			
					TCP/8443			
asaSub	132	from_outside	*	*	IP/*	deny		
asaSub	135	from_dmz	*	*	IP/*	deny		
asaUCCtoBA	126	from_dmz	UCC_ICCP_Server	Public_Database	SQL	permit		
			172.16.1.130	172.16.1.35	TCP/1433			
asaUCCtoBA	127	from_dmz	*	*	IP/*	deny		
asaUCCtoBA	130	from_inside	*	*	IP/*	deny		
asaUCCtoBA	133	from_outside	BA_ICCP_Server	UCC_ICCP_Server	ICCP	permit		
			192.168.1.2	172.16.1.130	TCP/102			
asaUCCtoBA	134	from_outside	*	*	IP/*	deny		
asaUCCtoSub	122	from_inside	SCADA_Server	Relays	SCADA	permit		
			172.16.1.3	10.1.1.2	TCP/20000			
				10.1.1.3				
				10.1.1.4				
asaUCCtoSub	123	from_inside	HMI_Node	Local_Web_Server	HTTP	permit		Low Risk alert: TCP/80 HTTP TCP/443 HTTPS TCP/8080 HTTP-alt
			172.16.1.4	10.1.1.131	TCP/80			
					TCP/443			
					TCP/8080			
					TCP/8443			
asaUCCtoSub	124	from_inside	UCC_ICCP_Server	Public_Database	SQL	permit		
			172.16.1.130	172.16.1.35	TCP/1433			
asaUCCtoSub	125	from_inside	SCADA_Server	Public_Database	SQL	permit		
			172.16.1.3	172.16.1.35	TCP/1433			
asaUCCtoSub	126	from_inside	*	*	IP/*	deny		
asaUCCtoSub	129	from_dmz	*	*	IP/*	deny		
asaUCCtoSub	132	from_outside	*	*	IP/*	deny		

Figure 4.5: Rule Audit Part 2

5. TEST BED IMPLEMENTATION

After the smart grid network model was configured and tested to optimize the rule sets on the firewalls for greater security, the next step was to incorporate this model into the cyber-physical CYPRES test bed. In the previous chapters, the network was established using Cisco Packet Tracer and tested using NP-View. In this chapter, the cyber network is combined with the power system. Using the test bed, a power simulator is implemented to test real DNP3 packets that can be viewed using a packet sniffer program.

To accomplish this, the Cisco ASA configurations were first converted into Linux *iptables* to test the model with the existing simulations for the network. These configurations and other *iptables* samples were then experimented with in the test bed.

5.1 CYPRES Test Bed

The CYPRES team has a test bed which has been configured with several Linux virtual machines to simulate and emulate a smart grid environment, as shown in the diagram in Figure 5.1. PowerWorld is used as the power simulator, to simulate the substation, and is connected to the test bed through the Ethernet ports shown at the bottom of Figure 5.1.

The network emulator that is used is called Common Open Research Emulator (CORE) [25], which allows this test bed to run test cases in real-time with the PowerWorld power simulator, DNP3 client and the other components of the utility control center and balancing authority. CORE contains the virtual machines (routers) where the *iptables* (i.e., firewall rules) will be created.

Each color block of the test bed is representative of a different part of a smart grid network including the UCC, substation, and BA. The substation is modeled by PowerWorld and represents the DNP3 Outstation (O/S) shown in Figure 5.1 as the node labeled "ens193" connected to the RTAC switch in the substation.

In this diagram, the firewall rules are configured on each router, as gateways into each network. The switch labeled rtacSwitch represents the subnet where the relays network would be located.

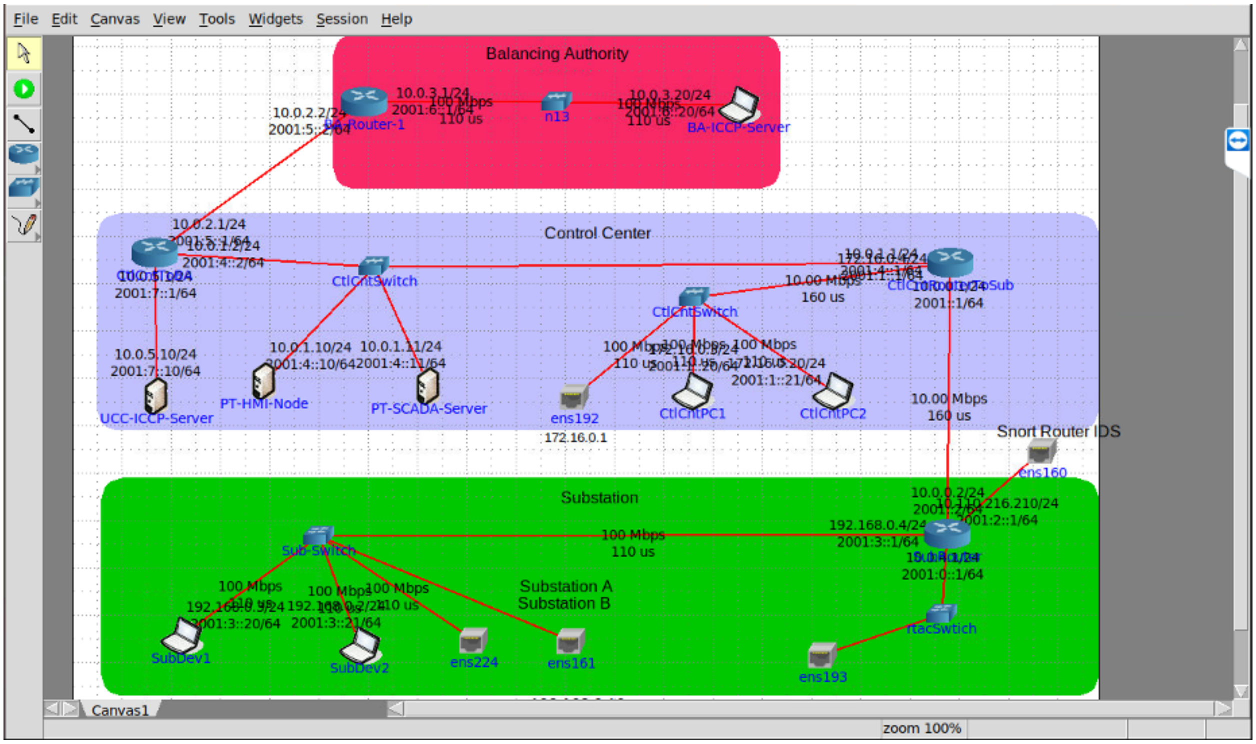


Figure 5.1: Test Bed Overview

The SCADA Server, Human Machine Interface (HMI) node, and ICCP servers are labeled as well. However, the test bed network does not have DMZ's implemented, hence the firewall rules were written to be enforced on all interfaces of each router.

Table 5.1 shows the mapping of all nodes in the test bed used in the *iptables* configurations and their IP addresses. These are the nodes which will be used to test each data flow in the test bed. Note that these IP address assignments are different from the IP address assignments used in the model developed for this research, as discussed in the previous chapters.

5.2 *Iptables* Conversion

Iptables are used for the firewalls inside of our test bed environment, which runs on Linux operating system. They are a Linux-based command-line firewall which can be configured with rules called policy chains. As nodes try to establish a connection with a node inside the network, these rules are scanned for a match to determine what to do with that packet. These rules are

Table 5.1: Test Bed IP Addresses

Location	Node	IP Address
Substation	DNP3 O/S (ens193)	192.168.0.5
Substation	SubPC1	192.168.0.3
Utility Control Center	DNP3 Master (ens192)	172.16.0.2
Utility Control Center	SCADA Server	10.0.1.11
Utility Control Center	HMI Node	10.0.1.10
Utility Control Center	CtlPC1	172.16.0.3
Utility Control Center	UCC ICCP Server	10.0.5.10
Balancing Authority	BA ICCP Server	10.0.3.20

scanned for both incoming and outgoing packets. There are different ways to set these policies up, if a packet does not match a rule it can either silently discard the packet, or reject it and inform the node that tried to connect to it that it was rejected, depending on the configuration. *Iptables* were useful for this project as they allowed for testing of the firewall configurations in the test bed environment, before being implemented in a real network.

To convert ACL's from Cisco ASA configurations into *iptables* [26], the default filter table is used. This table is pre-defined with three chains: the forward chain, input chain, and output chain. The forward chain is used to inspect all packets incoming to the firewall, before they are either forwarded to their destination or discarded. This matches the inbound direction of the ACL's written for the Cisco ASA firewall rules, so that the firewall inspects the packets as they enter through a specific interface, as opposed to as the packets are leaving the network.

5.2.1 Substation Firewall

The following text shows the *iptables* rule set for the firewall in the substation. The format is similar to that of the ACL's written for the Cisco ASA rules described in Section 3.1.

```
  iptables -I FORWARD -p tcp -s 172.16.1.3 -d 10.1.1.2 -dport 20000 -j
ACCEPT
  iptables -I FORWARD -p tcp -s 172.16.1.3 -d 10.1.1.3 -dport 20000 -j
```

```
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.3 -d 10.1.1.4 -dport 20000 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 80 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 443 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 8080
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 8443
-j ACCEPT
   iptables -A FORWARD -j DROP
   iptables -I FORWARD -p tcp -s 10.1.1.2 -d 10.1.1.130 -dport 1433 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 10.1.1.3 -d 10.1.1.130 -dport 1433 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 10.1.1.4 -d 10.1.1.130 -dport 1433 -j
ACCEPT
   iptables -A FORWARD -j DROP
```

Each *iptables* rule starts with the word "iptables", and then follows with either an "-A", "-I", "-R", or "-D", to signify that the rule will be appended, inserted, replaced, or deleted. These rules were written with a "-I" to insert each new rule at the beginning of the chain. Next, a keyword of either "forward", "input", or "output" is used. In this case, the "forward" keyword indicates that the firewall should receive the packet for inspection, but is not the source or destination address. If the packet matches the rules written, the packet will be forwarded to its appropriate destination. Next, the "-i" keyword is used which stands for "interface" and tells the device which interface to apply the rule on. Without specifying an interface, the rule would be applied to all interfaces. Following this, a "-p" is written which stands for the protocol, and tells the system that the protocol type is written next, in this case, all applications permitted use the TCP protocol. The next part of the rule is the source and destination addresses. Before the source address "-s" is given to signify source, and then a "-d" is written after to signify that the destination address of the packet is given next. The last piece of information to match on the packet is the application used, which is matched by the port number. For this the keyword "–dport" is used, before the port number, which stands for destination port. Lastly, a "-j" is given, telling the device to jump to either accept, reject, or drop the packet, if it fully matches the rule configured. In this case all rules show permitted traffic except for the last rule, which is to deny all other traffic not matching the ruleset.

The *iptables* rules are configured for the firewall to allow the data flow over DNP3 from the inside interface and the data flows over DNP3 and HTTP from the outside interface. First, the inside interface allows the relays to initiate a connection with the local database in the substation DMZ over SQL (port 1443). Then, the outside interface allows the SCADA server to be able to initiate a connection with the relays in the substation over DNP3 (port 20000), and the HMI node to be able to initiate a connection with the local web server in the substation DMZ over HTTP (ports 80, 443, 8080, and 8443.)

5.2.2 Firewall Separating Control Center and Substation

The following text shows the *iptables* configuration for the firewall connecting the control center and the substation, which is mainly for DNP3 and web traffic data flows.

```
   iptables -I FORWARD -p tcp -s 172.16.1.3 -d 10.1.1.2 -dport 20000 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.3 -d 10.1.1.3 -dport 20000 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.3 -d 10.1.1.4 -dport 20000 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 80 -j
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 443 -j
```

```
ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 8080
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.4 -d 10.1.1.131 -dport 8443
-j ACCEPT
   iptables -A FORWARD -j DROP
```

5.2.3 DMZ Firewall

Next, the firewall separating the DMZ's in the control center was converted into *iptables*. This firewall established the data flows allowed between the public DMZ, vendor DMZ, and corporate DMZ. First, the vendor node is allowed to initiate a connection with the database in the public DMZ over SQL (port 1433). Next, the same vendor node is allowed to access the web server in the public DMZ over HTTP (ports 80, 443, 8080, 8443). The corporate server is also allowed these same two data flows: it is permitted to access the public database over SQL, and the public web server over HTTP. The following text shows the *iptables* configuration for the firewall in the control center separating the DMZ's.

```
   iptables -I FORWARD -p tcp -s 172.16.1.66 -d 172.16.1.35 -dport 1433 -
j
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.66 -d 172.16.1.36 -dport 80 -j
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.66 -d 172.16.1.36 -dport 443 -
j
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.66 -d 172.16.1.36 -dport 8080 -
j
ACCEPT
```

```
   iptables -I FORWARD -p tcp -s 172.16.1.66 -d 172.16.1.36 -dport 8443 -
j
ACCEPT
   iptables -A FORWARD -j DROP
   iptables -I FORWARD -p tcp -s 172.16.1.97 -d 172.16.1.35 -dport 1433
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.97 -d 172.16.1.36 -dport 80
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.97 -d 172.16.1.36 -dport 443
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.97 -d 172.16.1.36 -dport 8080
-j ACCEPT
   iptables -I FORWARD -p tcp -s 172.16.1.97 -d 172.16.1.36 -dport 8443
-j ACCEPT
   iptables -A FORWARD -j DROP
```

5.2.4 Firewall Separating Control Center and Balancing Authority

Next, the firewall between the UCC and the BA was converted into *iptables*. Only one rule was written to allow the data flow between the ICCP server in the control center and the ICCP server in the BA over ICCP. The following text shows the *iptables* configuration for the firewall connecting the control center and the balancing authority.

```
   iptables -I FORWARD -p tcp -s 172.16.1.130 -d 172.16.1.35 -dport
1433 -j ACCEPT
   iptables -A FORWARD -j DROP
   iptables -I FORWARD -p tcp -s 192.168.1.2 -d 172.16.1.130 -dport 102
-j ACCEPT
   iptables -A FORWARD -j DROP
```

5.2.5 Balancing Authority Firewall

Lastly the firewall in the balancing authority network was converted. This firewall had only one rule, to allow the BA's ICCP server to initiate a connection the ICCP server in the control center over ICCP (port 102). The following text shows the *iptables* configuration for the firewall within the balancing authority.

```
   iptables -I FORWARD -p tcp -s 192.168.1.2 -d 172.16.1.130 -dport 102
-j ACCEPT
   iptables -A FORWARD -j DROP
```

The second two lines allow for two-way communication even though the connection can only be established one way. This ensures that the SCADA server can only initiate connections, not receive any.

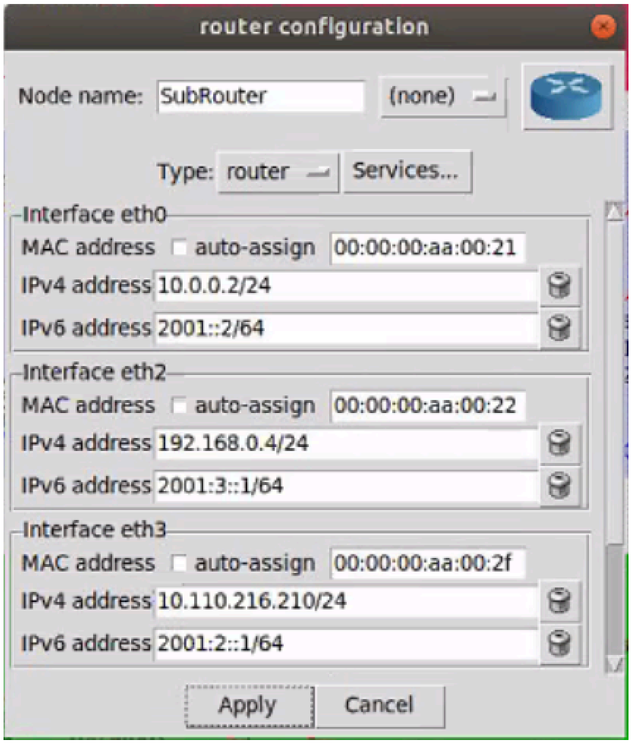
5.3 Experiments with *Iptables* in the Test Bed

Once these Cisco ASA rules were converted into *iptables*, the rules were tested in the CYPRES team's test bed. To test the *iptables*, rules were configured into the substation router of the network in CORE emulator shown on the bottom right side of Figure 5.1. The interface configuration for this router is shown in Figure 5.2.

Interface Ethernet 0, shown at the top of Figure 5.2, connects the substation router to the control center router, and interface Ethernet 2 connects the substation router to the DNP3 O/S and relay network in the substation network.

Next, the router on the right side of the control center in Figure 5.1 was configured. Interface Ethernet 0, shown at the top of Figure 5.3, connects the control center network to the substation network. Interface Ethernet 1 connects with the control center network where the SCADA server and HMI node is located. Lastly, interface Ethernet 2 connects with rest of the control center where the remaining devices are located. This configuration is shown in Figure 5.3.

To test each *iptables* rule, a set of commands was loaded into the routers, and then each specific application was tested between the affected nodes. To test if the rules were getting "matches" or



Node name: SubRouter (none)

Type: router

Interface eth0

MAC address □ auto-assign 00:00:00:aa:00:21

IPv4 address 10.0.0.2/24

IPv6 address 2001::2/64

Interface eth2

MAC address □ auto-assign 00:00:00:aa:00:22

IPv4 address 192.168.0.4/24

IPv6 address 2001:3::1/64

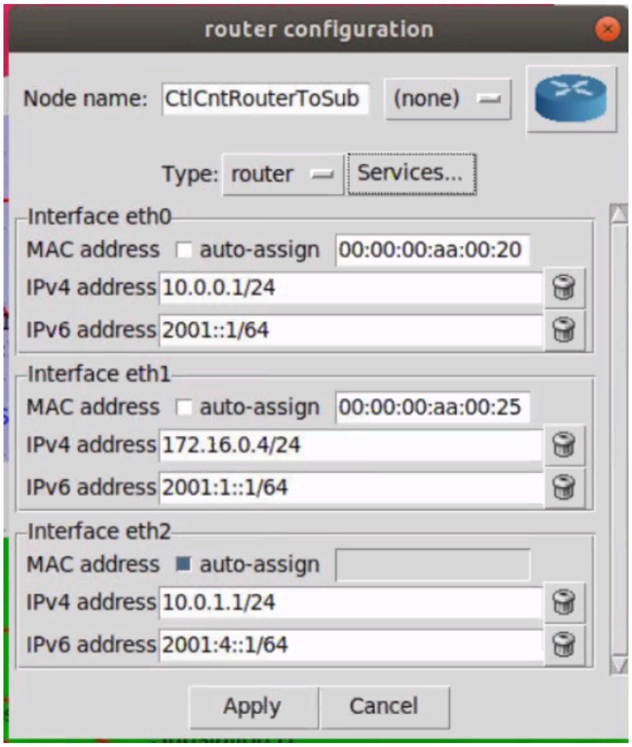
Interface eth3

MAC address □ auto-assign 00:00:00:aa:00:2f

IPv4 address 10.110.216.210/24

IPv6 address 2001:2::1/64

Figure 5.2: Interface Configuration for the Substation Router



Node name: CtlCntRouterToSub (none)

Type: router

Interface eth0

MAC address □ auto-assign 00:00:00:aa:00:20

IPv4 address 10.0.0.1/24

IPv6 address 2001::1/64

Interface eth1

MAC address □ auto-assign 00:00:00:aa:00:25

IPv4 address 172.16.0.4/24

IPv6 address 2001:1::1/64

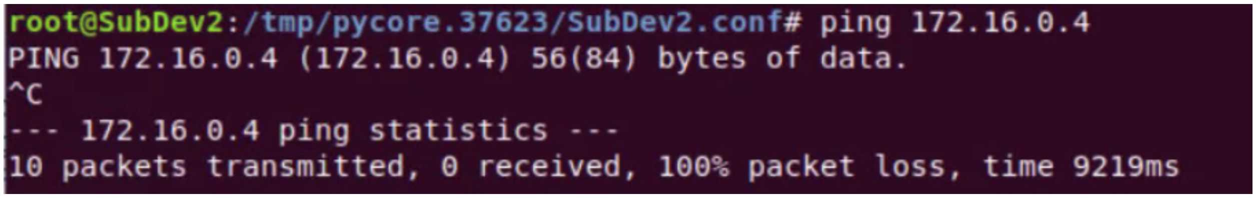
Interface eth2

MAC address ■ auto-assign

IPv4 address 10.0.1.1/24

IPv6 address 2001:4::1/64

Figure 5.3: Interface Configuration for the Control Center Router Connected to the Substation



```
root@SubDev2:/tmp/pycore.37623/SubDev2.conf# ping 172.16.0.4
PING 172.16.0.4 (172.16.0.4) 56(84) bytes of data.
^C
--- 172.16.0.4 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9219ms
```

Figure 5.4: Testing a Ping from SubDev2 to Control Center Router

"hits," the command "iptables -vL" was entered in the terminal to show each rule table configured, and the number of times a match was received.

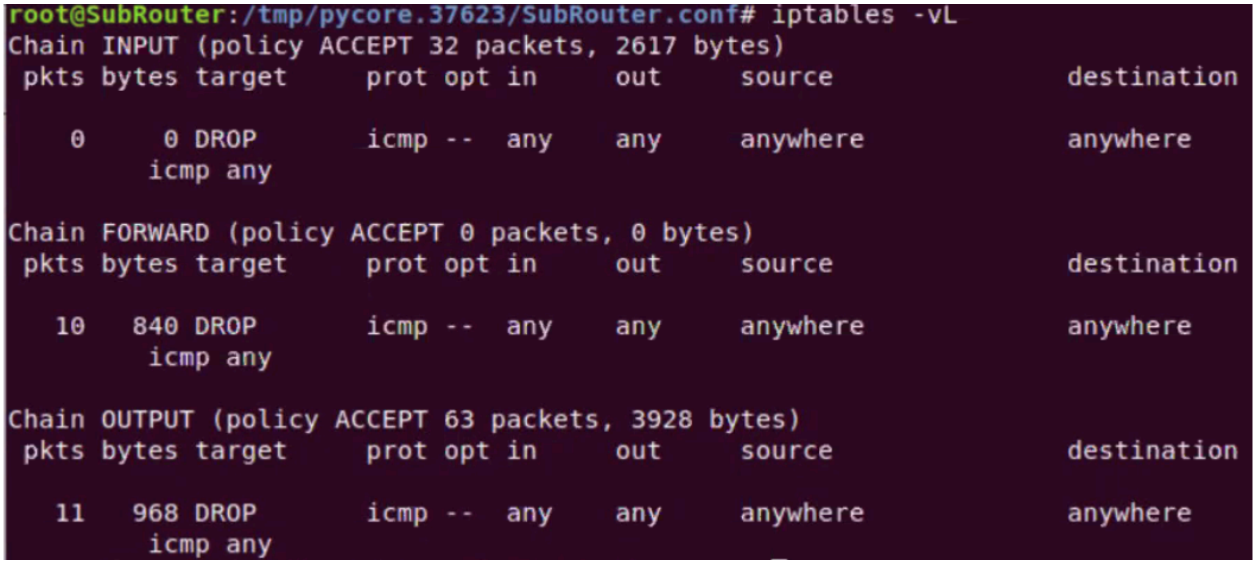
5.3.1 Internet Control Message Protocol (ICMP) Test

The first test was to block "pings" and ensure that no device could access any node in the substation by pinging it over Internet Control Message Protocol (ICMP). Also that no device in the substation could access the control center over ICMP, which is a debugging tool that accompanies the internet protocol (IP) and is used to test reachability. Figure 5.4 shows the results of a ping after the rule was implemented, which had a 100% packet loss. Figure 5.5 shows the number of hits after running this ping test. Hits were received on the output chain because the packets were sent to the outgoing interface of the router towards the control center network, thus 968 packets were dropped. Hits were also received on the forward chain, since the destination address was not one of the router's interfaces, and the packet would have been forwarded if it was accepted.

After conducting this test, the same rule was tested but with a drop probability created to see how many packets would then match each of the chains. The rule was created with a 40% drop policy. The result was that 41 packets were dropped out of the 100 sent, so there was a 59% packet loss. The drop probability parameter is set in the *iptables* rule using the command:

```
icmp any statistic mode random probability 0.39999999991
```

This configures the router to drop packets with a 40% probability.



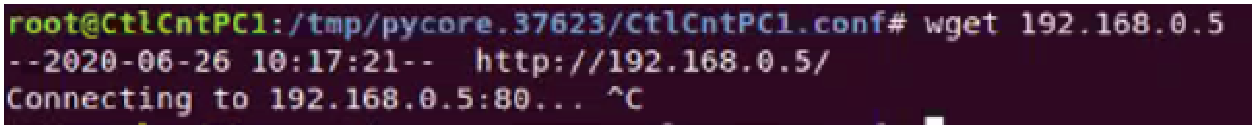
```
root@SubRouter:/tmp/pycore.37623/SubRouter.conf# iptables -vL
Chain INPUT (policy ACCEPT 32 packets, 2617 bytes)
 pkts bytes target     prot opt in     out     source               destination

    0     0 DROP       icmp --  any    any     anywhere             anywhere
         icmp any

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

   10   840 DROP       icmp --  any    any     anywhere             anywhere
         icmp any

Chain OUTPUT (policy ACCEPT 63 packets, 3928 bytes)
 pkts bytes target     prot opt in     out     source               destination

   11   968 DROP       icmp --  any    any     anywhere             anywhere
         icmp any
```

Figure 5.5: Rule Matching for ICMP

5.3.2 HyperText Transfer Protocol (HTTP) Test

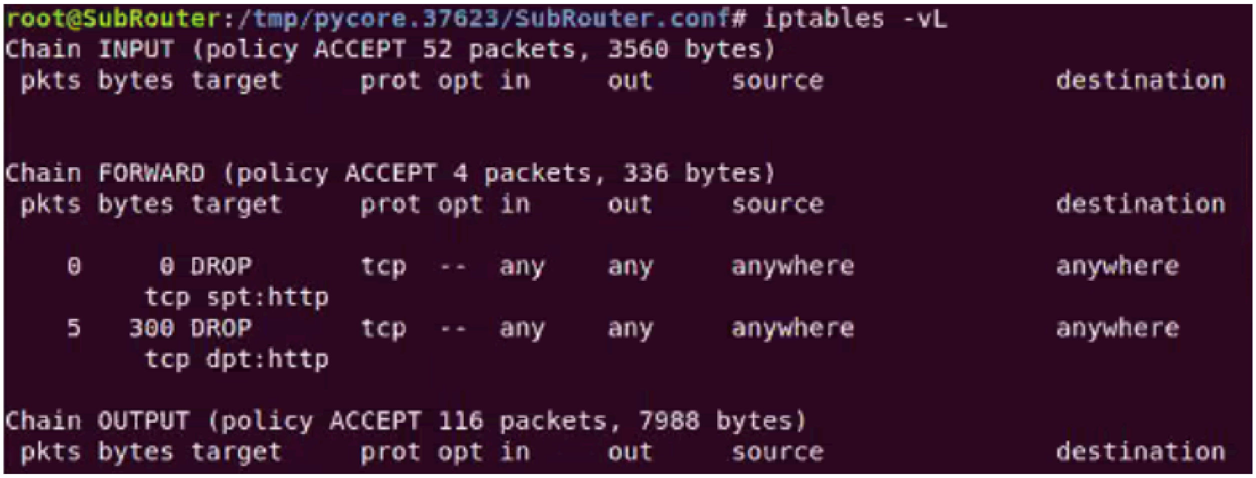
The next test was to ensure that the HTTP rules were functioning correctly. To test this, the "wget" command was used to connect to a web server with IP address of 192.168.0.5, before and after configuring an *iptables* rule to drop HTTP packets. To drop the HTTP packets, port 80 was blocked as both a destination port and a source port using the "dport" and "sport" keywords. Figure 5.6 shows the result of the attempt of the substation router trying to connect to the web server after implementing the rule with no success. Figure 5.7 then shows the number of hits on each of the three policy chains after the packets were sent.

To further test the HTTP application, one specific node was blocked on one exact interface over port 80. Specifically, IP address 172.16.0.3 was blocked from accessing the same web server at IP address 192.168.0.5. Figure 5.8 shows the connection test after this specific node was blocked.



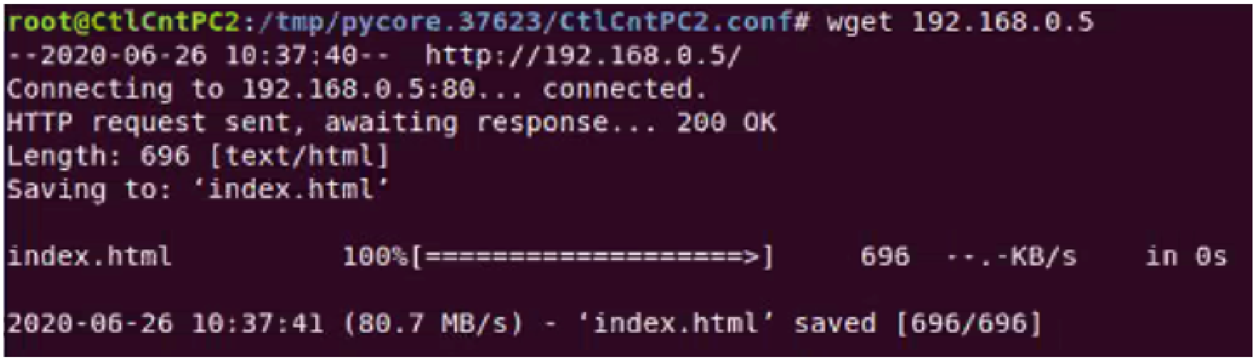
```
root@CtlCntPC1:/tmp/pycore.37623/CtlCntPC1.conf# wget 192.168.0.5
--2020-06-26 10:17:21--  http://192.168.0.5/
Connecting to 192.168.0.5:80... ^C
```

Figure 5.6: Testing a Ping from SubDev2 to Control Center Router



```
root@SubRouter:/tmp/pycore.37623/SubRouter.conf# iptables -vL
Chain INPUT (policy ACCEPT 52 packets, 3560 bytes)
 pkts bytes target     prot opt in     out     source               destination


Chain FORWARD (policy ACCEPT 4 packets, 336 bytes)
 pkts bytes target     prot opt in     out     source               destination

    0     0 DROP       tcp  --  any    any     anywhere             anywhere
        tcp spt:http
    5   300 DROP       tcp  --  any    any     anywhere             anywhere
        tcp dpt:http

Chain OUTPUT (policy ACCEPT 116 packets, 7988 bytes)
 pkts bytes target     prot opt in     out     source               destination
```

Figure 5.7: Rule Matching for HTTP



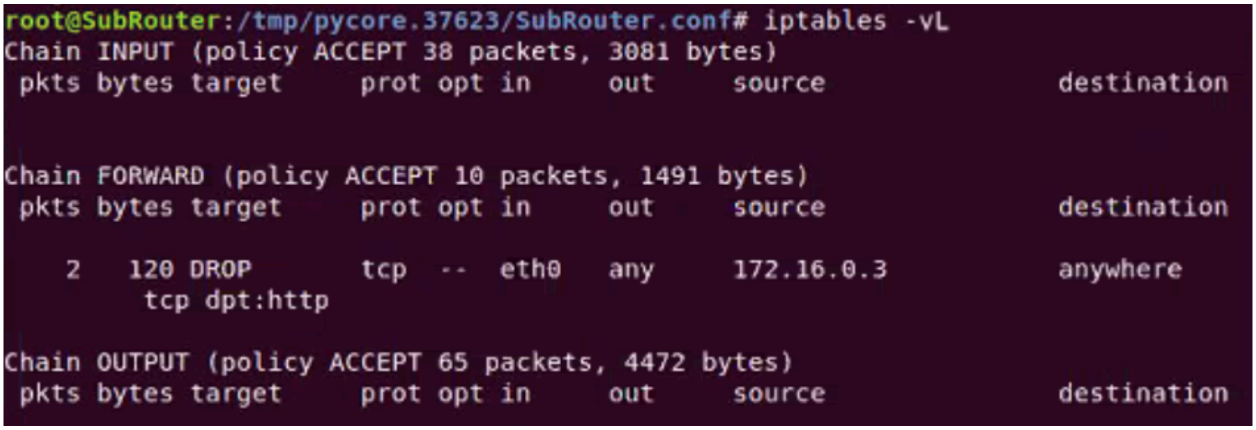
```
root@CtlCntPC2:/tmp/pycore.37623/CtlCntPC2.conf# wget 192.168.0.5
--2020-06-26 10:37:40--  http://192.168.0.5/
Connecting to 192.168.0.5:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 696 [text/html]
Saving to: ‘index.html’

index.html          100%[===================>]     696  --.-KB/s    in 0s

2020-06-26 10:37:41 (80.7 MB/s) - ‘index.html’ saved [696/696]
```

Figure 5.8: Testing a Ping from SubDev2 to Control Center Router

The test was successful, since other interfaces were allowed to access the web server over HTTP. Figure 5.9 shows the policy chains after testing this rule. Matches are only seen on the forward chain, since the packet’s destination did not match the IP address assigned to any interface of the router.

5.3.3 Distributed Network Protocol 3 (DNP3) Test

The next test conducted on the test bed was to test the DNP3 application, which is used between the DNP3 Master in the UCC and the DNP3 Outstation in the substation. In this network the DNP3 Master has an IP address of 172.16.0.2 and the DNP3 O/S has an IP address of 192.168.0.5. To



```
root@SubRouter:/tmp/pycore.37623/SubRouter.conf# iptables -vL
Chain INPUT (policy ACCEPT 38 packets, 3081 bytes)
 pkts bytes target     prot opt in     out     source               destination


Chain FORWARD (policy ACCEPT 10 packets, 1491 bytes)
 pkts bytes target     prot opt in     out     source               destination

    2   120 DROP       tcp  --  eth0   any     172.16.0.3           anywhere
         tcp dpt:http

Chain OUTPUT (policy ACCEPT 65 packets, 4472 bytes)
 pkts bytes target     prot opt in     out     source               destination
```

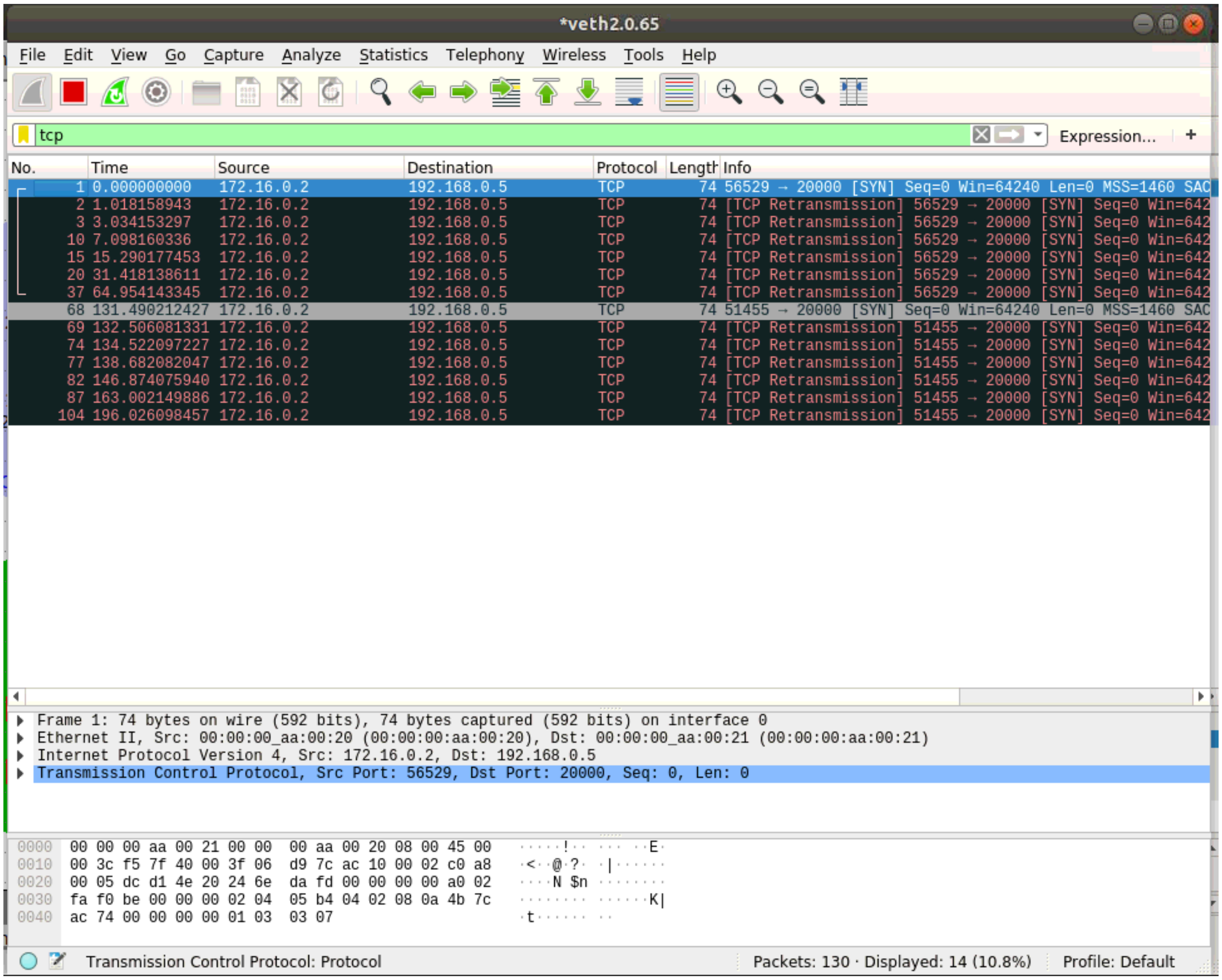
Figure 5.9: Rule Matching for HTTP

test that the DNP3 connection is working, a packet sniffing application called Wireshark is used to see each packet. Figure 5.10 shows a screenshot of the DNP3 packets, which are TCP packets on port 20000. These packets are flowing from the DNP3 Master to the DNP3 O/S.

5.3.4 Secure Shell (SSH) Test

The last application tested in the CORE network was SSH, which would be used by a vendor node to remotely access a server or database in the public DMZ. To test this, an SSH connection (port 22) was established from a node in the control center at IP address 172.16.0.3 to a node in the substation at IP address 192.168.0.3. First, an *iptables* rule was written to allow connections initiating from either of these two nodes to any destination. The result was that the SSH attempt was allowed.

In summary, all Cisco ASA rulesets established in Chapter 3 were converted into Linux-based *iptables* in this chapter. These conversions were then tested in the CYPRES team's test bed, to ensure that the data flows were correct and followed the diagram in Figure 2.1.


*veth2.0.65
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp
Expression...
+

```
No.   Time          Source       Destination   Protocol Length Info
   1 0.000000000   172.16.0.2   192.168.0.5   TCP      74 56529 → 20000 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SAC
   2 1.018158943   172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 56529 → 20000 [SYN] Seq=0 Win=642
   3 3.034153297   172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 56529 → 20000 [SYN] Seq=0 Win=642
  10 7.098160336   172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 56529 → 20000 [SYN] Seq=0 Win=642
  15 15.290177453  172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 56529 → 20000 [SYN] Seq=0 Win=642
  20 31.418138611  172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 56529 → 20000 [SYN] Seq=0 Win=642
  37 64.954143345  172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 56529 → 20000 [SYN] Seq=0 Win=642
  68 131.490212427 172.16.0.2   192.168.0.5   TCP      74 51455 → 20000 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SAC
  69 132.506081331 172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 51455 → 20000 [SYN] Seq=0 Win=642
  74 134.522097227 172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 51455 → 20000 [SYN] Seq=0 Win=642
  77 138.682082047 172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 51455 → 20000 [SYN] Seq=0 Win=642
  82 146.874075940 172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 51455 → 20000 [SYN] Seq=0 Win=642
  87 163.002149886 172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 51455 → 20000 [SYN] Seq=0 Win=642
 104 196.026098457 172.16.0.2   192.168.0.5   TCP      74 [TCP Retransmission] 51455 → 20000 [SYN] Seq=0 Win=642
```

```
▸ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
▸ Ethernet II, Src: 00:00:00_aa:00:20 (00:00:00:aa:00:20), Dst: 00:00:00_aa:00:21 (00:00:00:aa:00:21)
▸ Internet Protocol Version 4, Src: 172.16.0.2, Dst: 192.168.0.5
▸ Transmission Control Protocol, Src Port: 56529, Dst Port: 20000, Seq: 0, Len: 0
```

```
0000  00 00 00 aa 00 21 00 00  00 aa 00 20 08 00 45 00   ·····!·· ··· ··E·
0010  00 3c f5 7f 40 00 3f 06  d9 7c ac 10 00 02 c0 a8   ·<··@·?· ·|······
0020  00 05 dc d1 4e 20 24 6e  da fd 00 00 00 00 a0 02   ····N $n ········
0030  fa f0 be 00 00 00 02 04  05 b4 04 02 08 0a 4b 7c   ········ ······K|
0040  ac 74 00 00 00 00 01 03  03 07                     ·t······ ··
```

Transmission Control Protocol: Protocol   Packets: 130 · Displayed: 14 (10.8%)   Profile: Default

Figure 5.10: Wireshark View Showing DNP3 Packets

6. SUMMARY AND CONCLUSIONS

The objective of this research was to design and test a set of firewall configuration files for use in a smart grid network, which could then be implemented into the Texas 2000-bus synthetic grid. These types of networks are more vulnerable to attacks, since more communication occurs throughout the network, creating more connections and possible paths through the network.

6.1 Contributions

To accomplish this goal, the main data flows required by a power grid network were first established using best-practice industry standards. It is important to define exactly who needs to access which nodes, and how those nodes need to be accessed. This data flow model was used to create a cyber-topology for a basic smart grid network representing the OT network traffic of an electric utility company, including a substation, control center, balancing authority and several DMZ's. This topology can later be used as a base model for the entire Texas 2000-bus synthetic grid network. Then, the five firewalls defined in the network were configured and tested on Cisco ASA's using Cisco Packet Tracer. This allowed for a practical implementation, since industry experts do program Cisco ASA firewalls using the command-line interface (CLI), as Cisco Packet Tracer allows the user to do. Next, Network Perception's NP-View tool was used to test these rules and highlight every path into and out of the network. The tool was also used to generate a table of the rules to verify that they matched the data flow model.Finally, these firewall configurations were implemented in the CYPRES team's test bed. First, the rules were converted into Linux-based *iptables*, and the IP addresses used in the model were mapped to the IP addresses used in the test bed. Next, several of the rules were tested by allowing or denying a specific application over the appropriate protocol. The rules for ICMP, HTTP, DNP3, and SSH were tested, to ensure that the device was forwarding or dropping the packets according to the configured rules.

6.1.1 Lessons Learned

Many challenges were overcome throughout this research project, including learning how to use a Linux environment to test the firewall configurations. The rules written for the Cisco ASA firewalls do not translate exactly to the Linux-based *iptables*, and so some manipulation of the rules was required. For instance, without the use of DMZ's in the test bed, the rules were written to be applied to all interfaces of the router.

From this experience, I also learned much about how industry experts configure firewalls for use in their utility networks. Static routes are often hard-coded onto the router, seeming counter-intuitive, as the router cannot dynamically change the network path if a link goes down. This adds a level of security, and allows operators to quickly identify if there is a problem in the network or if there has been a breach.

6.2 Future Work

The next phase of the project could be to implement the firewall configurations for this model into Cyber Physical Security Assessment (CyPSA) project. CyPSA was developed as a tool for utility companies to use to conduct a security assessment of their current network and power grid and produce an attack graph. The attack graph would show any vulnerabilities in the network based off of the network topology and firewall rule configurations. Compared to the earlier implementation of CyPSA with an 8-bus model, the new model would include a Balancing Authority network, meaning that it represents a more complete model of a true smart grid network. CyPSA would then process the extra nodes of the BA's ICCP server and the BA firewall, and use them in the creation of the attack graphs to assess all paths into and out of the network.

Another step in furthering the development of this model is to implement DMZ's in the CYPRES test bed. Currently there are no DMZ's, and the rulesets were configured onto routers instead of firewalls. Adding the DMZ's would enhance the ability of the team to test the cyber-topology model in an real industry environment.

This research is crucial because networks across all industries are becoming more vulnerable to cyber-attacks as technology is evolving. Any type of attack on a power grid could have disastrous consequences. One attack could leave hundreds of thousands of people without power and unable to perform their daily activities or jobs. Therefore, it is important to work to create a secure network structure that can be recommended and used in smart grid networks. It is important to follow best practices for the industry and ensure that these networks are as secure as possible. To follow these standards, the network is secure when the defined data flows are permitted between the specified nodes. This means that only a node from the control center or node from the operational network is allowed to initiate a connection depending on the specified data flow, which complies with NERC-CIP-005. The use of DMZs in the network allows for the separation of network devices for different operations. This research will produce a set of firewall rules and configurations to be implemented in a large-scale smart grid network that will follow industry standards and best practices.

REFERENCES

[1] CIP-005-5, Cyber security electronic security perimeter(s). [Online]. Available: https://www.nerc.com/pa/Stand/Reliability%20Standards/CIP-005-5.pdf.

[2] Deep Cyber Physical Situational Awareness for Energy Systems: A Secure Foundation for Next-Generation Energy Management. [Online]. Available: https://cypres.engr.tamu.edu/.

[3] A. B. Birchfield, T. Xu, K. M. Gegner, K. S. Shetye, and T. J. Overbye, "Grid structural characteristics as validation criteria for synthetic networks," *IEEE Transactions on power systems*, vol. 32, no. 4, pp. 3258–3265, 2016.

[4] ACTIVSg2000: 2000-bus synthetic grid on footprint of Texas. [Online]. Available: https://electricgrids.engr.tamu.edu/electric-grid-test-cases/activsg2000/.

[5] P. Wlazlo, K. Price, C. Veloz, A. Sahu, H. Huang, A. Goulart, K. Davis, and S. Zounouz, "A cyber topology model for the texas 2000 synthetic electric power grid," in *2019 Conference on Principles, Systems and Applications of IP Telecommunications (IPTComm)*, IEEE, October, 2019.

[6] K. Davis, R. Berthier, S. Zonouz, G. Weaver, R. Bobba, E. Rogers, P. Sauer, and D. Nicol, "Cyber-physical security assessment (cypsa) for electric power systems," *IEEE-HKN: THE BRIDGE*, 2016.

[7] G. A. Weaver, C. Cheh, E. J. Rogers, W. H. Sanders, and D. Gammel, "Toward a cyber-physical topology language: Applications to nerc cip audit," in *Proceedings of the first ACM workshop on Smart energy grid security*, pp. 93–104, ACM, 2013.

[8] G. A. Weaver, K. Davis, C. M. Davis, E. J. Rogers, R. B. Bobba, S. Zonouz, R. Berthier, P. W. Sauer, and D. M. Nicol, "Cyber-physical models for power grid security analysis: 8-substation case," in *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 140–146, IEEE, 2016.

[9] C. Ten, C. Liu, and M. Govindarasu, "Vulnerability assessment of cybersecurity for scada systems using attack trees," in *2007 IEEE Power Engineering Society General Meeting*, pp. 1–8, June 2007.

[10] L. Peterson and B. Davie, *Computer Networks: A Systems Approach*. Morgan Kaufmann Publishers, 2012.

[11] G. Liang, S. Weller, J. Zhao, F. Luo, and Z. Dong, "The 2015 ukraine blackout: Implications for false data injection attacks," *IEEE Transactions on Power Systems*, vol. PP, pp. 1–1, 11 2016.

[12] D. E. Whitehead, K. Owens, D. Gammel, and J. Smith, "Ukraine cyber-induced power outage: Analysis and practical mitigation strategies," in *2017 70th Annual Conference for Protective Relay Engineers (CPRE)*, pp. 1–8, 2017.

[13] D. Kushner, "The real story of stuxnet," *IEEE Spectrum*, vol. 50, no. 3, pp. 48–53, 2013.

[14] R. Liu, C. Vellaithurai, S. S. Biswas, T. T. Gamage, and A. K. Srivastava, "Analyzing the cyber-physical impact of cyber events on the power grid," *IEEE Transactions on Smart Grid*, vol. 6, no. 5, pp. 2444–2453, 2015.

[15] A. Wool, "A quantitative study of firewall configuration errors," *Computer*, vol. 37, pp. 62–67, June 2004.

[16] Network Perception, 2020. [Online]. Available: https://www.network-perception.com/np-view//.

[17] PowerWorld Simulator, 2019. [Online]. Available: https://www.powerworld.com/.

[18] Z. Trabelsi and H. Saleous, "Exploring the opportunities of cisco packet tracer for hands-on security courses on firewalls," *2019 IEEE Global Engineering Education Conference (EDUCON)*, pp. 411–418, 2019.

[19] S. Singh, *Automatic Verification of Security Policy Implementations*. PhD thesis, Univ. of Illinois at Urbana-Champaign, Urbana, 2012.

[20] J. Frahim and O. Santos, *Cisco ASA: All-in-One Firewall, IPS, Anti-X, and VPN Adaptive Security Appliance*. Cisco Press, 2010.

[21] OSI, "The power behind the pi server, https://www.osisoft.com/pi-system/pi-capabilities/pi-server/."

[22] L. Zhang and M. Huang, "A firewall rules optimized model based on service-grouping," in *2015 12th Web Information System and Application Conference (WISA)*, pp. 142–146, 2015.

[23] F. Soldo, A. Markopoulou, and K. Argyraki, "Optimal filtering of source address prefixes: Models and algorithms," in *IEEE INFOCOM 2009*, pp. 2446–2454, April 2009.

[24] N. Gaudet, A. Sahu, A. E. Goulart, E. Rogers, and K. Davis, "Firewall configuration and path analysis for smartgrid networks," in *2020 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*, pp. 1–6, 2020.

[25] 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.

[26] R. Z. Steve Suehring, *Linux Firewalls*. Novell Press, third ed., 2005.