9-29-2019

The IEC 61850 sampled measured values protocol: Analysis, threat identification, and feasibility of using NN forecasters to detect spoofed packets †

Mohamad El Hariri
Florida International University

Eric Harmon
Florida International University

Tarek Youssef
University of West Florida

Mahmoud Saleh
Florida Polytechnic University

Hany Habib
Florida International University

See next page for additional authors

energies

MDPI

Article

The IEC 61850 Sampled Measured Values Protocol: Analysis, Threat Identification, and Feasibility of Using NN Forecasters to Detect Spoofed Packets †

Mohamad El Hariri [1,*], Eric Harmon [1], Tarek Youssef [2,*], Mahmoud Saleh [3], Hany Habib [1] and Osama Mohammed [1]

1 Department of Electrical and Computer Engineering, Florida International University, Miami, FL 33174, USA; eharm007@fiu.edu (E.H.); hhabi003@fiu.edu (H.H.); mohammed@fiu.edu (O.M.)
2 Department of Electrical and Computer Engineering, University of West Florida, Pensacola, Fl 32514, USA
3 Department of Electrical and Computer Engineering, Florida Polytechnic University, Lakeland, FL 33805, USA; mahmoudsalahem@gmail.com
* Correspondence: melha003@fiu.edu (M.E.H.); tyoussef@uwf.edu (T.Y.); Tel.: +1-305-542-8827 (M.E.H.)
† This work is an extension of the authors' paper presented at IEEE EEEIC 2019 Genoa, Italy, and is selected for publication in Energies.

Received: 27 August 2019; Accepted: 27 September 2019; Published: 29 September 2019

Abstract: The operation of the smart grid is anticipated to rely profoundly on distributed microprocessor-based control. Therefore, interoperability standards are needed to address the heterogeneous nature of the smart grid data. Since the IEC 61850 emerged as a wide-spread interoperability standard widely accepted by the industry, the Sampled Measured Values method has been used to communicate digitized voltage and current measurements. Realizing that current and voltage measurements (i.e., feedback measurements) are necessary for reliable and secure noperation of the power grid, firstly, this manuscript provides a detailed analysis of the Sampled Measured Values protocol emphasizing its advantages, then, it identifies vulnerabilities in this protocol and explains the cyber threats associated to these vulnerabilities. Secondly, current efforts to mitigate these vulnerabilities are outlined and the feasibility of using neural network forecasters to detect spoofed sampled values is investigated. It was shown that although such forecasters have high spoofed data detection accuracy, they are prone to the accumulation of forecasting error. Accordingly, this paper also proposes an algorithm to detect the accumulation of the forecasting error based on lightweight statistical indicators. The effectiveness of the proposed methods is experimentally verified in a laboratory-scale smart grid testbed.

Keywords: artificial intelligence; attack detection; cyber security; microgrid; process bus; IEC 61850; sampled measured values; neural networks; forecasting; message spoofing

1. Introduction

The envisioned smart grid is conceived to be an automated cyber-physical system that integrates and utilizes high processing power and real-time monitoring and control capabilities to fulfill the expected requirements (e.g., maximum resilience and economic benefits). This vision includes a myriad of microgrids operating autonomously and coordinating with each to achieve the necessary objective functions (e.g., maximize their economic benefits by exchanging power directly, sell/buy power to/from the utility, or enhance the resilience of the network).

For the aforementioned vision to be realized, a decentralized monitoring and control architecture is expected to be utilized, or a combination between centralized and decentralized schemes (e.g., centralized controller could be utilized in small scale systems combined with a decentralized layer that manages the centralized controllers). Therefore, in these decentralized topologies, a connection to

existing industrial standards with interoperable protocols and data structures is needed to govern the secure and feasible exchange of information in the complex heterogeneous smart grid [1].

The International Electrotechnical Commission's (IEC) 61850 standard defines comprehensive and unified data models for electric substations' automation and control systems. It was developed by IEEE and the Technical Committee No. 57 Working Group 10 (TC57 WG10) for Ethernet (i.e., IEEE 802.3)-based communication. It's a well-known and widely accepted standard in the industry, which offers an inclusive data modeling method and an abstraction approach that unifies the data structure formats among equipment from different vendors [2]. Moreover, the IEC 61850 requirements presented a novel approach to allow all sensors measurements to be available simultaneously to all controllers within a control network by introducing the process bus concept. Hence, IEC 61850 could be adopted for distributed and decentralized control schemes. As shown in Figure 1, this is achieved by inserting a communication bus, namely the process bus, which separates the input/output of a controller from the controller itself, where the logic takes place. In this scheme, the merging units (MU)s are used to digitize the measurements and publish them to the process bus. The MUs subscribe to the process bus to get the specific commands with specific IDs to their respective physical device (e.g., boost converter. Also, the controllers follow the publisher/subscriber model, where they subscribe to the process bus for their respective measurements with unique ID and publish their commands. Accordingly, all the measurements are accessible to all IEC 61850-compatible devices within the network of the process bus in real-time, and in a form that follows the Sampled Measure Values (SMV) protocol.

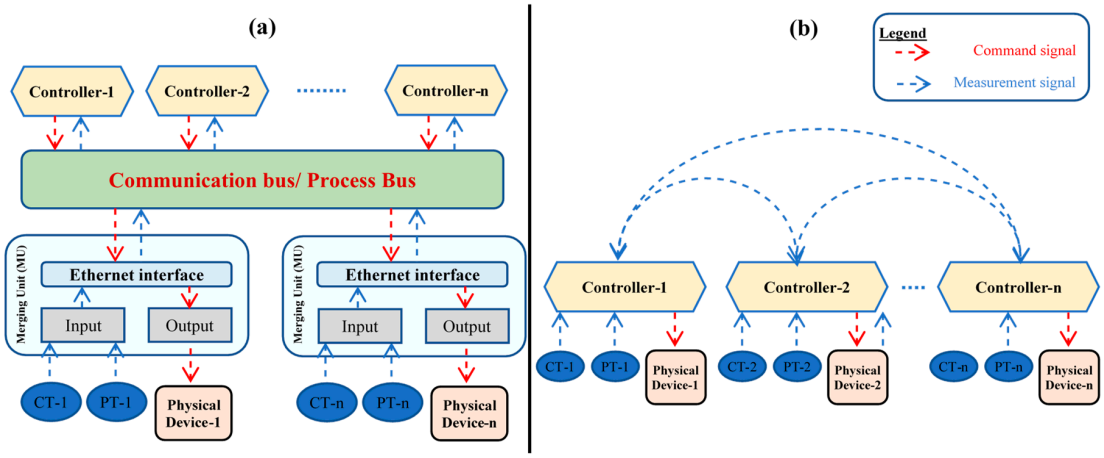


Figure 1. Control schemes according to: (**a**) the IEC 61850 process bus concept, (**b**) the conventional distributed control.

Each communication protocol has specific structure for messages. IEC 61850 maps the data to three different protocols, according to the application. The protocols are:

(1) Manufacturing Message Specification (MMS) protocol, which is utilized in high-level control layers and automation functionalities;
(2) Generic Object-Oriented Substation Event (GOOSE), which is used in real-time event-driven situations (e.g., sending a signal to open a circuit breaker);
(3) Sampled Measured Values (SMV) that is utilized in continuous real-time monitoring and control.

According to the above-mentioned description of each protocol, it can be perceived that SMV could be utilized in smart grid control operations. In this regard, SMV could offer numerous advantages such as facilitating the interoperability of data and reducing the complexity of the system connectivity. However, SMV serves real-time applications, which comes at the expenses of the vulnerability to cyber threats due to the absence of encryption for example. Therefore, in this manuscript, the authors extend their previous work [3] as follows:

- The authors' previous work examined the SMV protocol, its pros and cons, and the feasibility of incorporating it for smart grid applications. A discussion and explanation of the current cyber vulnerabilities of the smart grid and the possible counter measures, emphasizing on the SMV cyber threats were presented. Finally, an investigation of the feasibility of using a Neural Network Forecaster (NN-F) to detect spoofed SMV packets on a simulated microgrid was presented.

The previous work concluded that although NN-Fs have a high detection accuracy, they could still be susceptible to the accumulation of the forecasting error. The contributions of this manuscript can be summarized as follows:

(1) Proposing a lightweight bi-layer algorithm that is able to detect the accumulation of forecasting error and successfully identify spoofed packets.
(2) Implementing and verifying the proposed bi-layer algorithm on a hardware hybrid microgrid with microcontrollers running a Linux kernel and communicating over a real IEC 61850 network.

The rest of the paper is organized as follows:

- Section 2 highlights the work related to the applications of artificial intelligence tools (e.g., Machine learning) in cybersecurity;
- Section 3 discusses the structure of the SMV datagram and gives a general overview about the SMV advantages;
- Section 4 explains the various types of cyber threats and the possible protection actions. Also, it emphasizes on SMV spoofing attack and outlines the conducted research regarding this topic;
- Section 5 introduces a bi-layer forecasting algorithm to detect the SMV spoofing attacks using NN-F. Also, it discusses in detail the construction of Neural Network Forecaster (NN-F) detection-based framework;
- Section 6 describes the simulation model used for creating rich training and testing data sets.
- Section 7 discusses the results.
- Section 8 concludes the paper.

2. Applications of Artificial Intelligence Tools in Attack Detection

Machine learning has been utilized in power systems for application like forecasting energy consumption and forecasting the production capacity of renewables [4,5]. Given the recent worldwide interest in cyber security of the smart grid, researchers from academia and industry have been extensively exploring uses of machine learning for early attack detection.

Generally speaking, the work in this area could be classified as follows:

- Classification of measurement data as false or legitimate

The work in [6] utilizes data mining approaches to categorize the data of Phasor Measurement Units as normal or attacked. [7] presents analyzes the performance supervised learning techniques for classifying measurement data. In [8], in order to detect bad data, the authors introduce a weighted least square based detection algorithm. The aim of this work was to identify failures in current and potential transformers. [9] Classifies synchrophasor data as anomalous or not using machine learning techniques. [10] Compares the performance of several data-stream-mining-based intrusion detection techniques to detect anomalous behaviour in smart meter data. [11] Provides an integrated anomaly detection system on the host and network levels to detect attacks on a single substation or on several substations based on a predefined set of violations indications. [12] Uses common paths mining to detect intrusions on an electric transmission distance protection system using system data from synchrophasors and network logs. [13] Uses data mining and forensics tools to identify users' login behaviors, thus, preventing, detecting, and blocking insiders or attackers from performing malicious activities in closed environments.

- Learning from previous attack activity

In [14], the authors propose an early-warning-system to anticipate the occurrence of distributed denial of service attacks on wireless smart meter networks by modeling the malicious activity, which may compromise the privacy of users, as a Gaussian process. Then, this model is utilized to forecast future attack activity. The work in [15] detects network traffic that deviates from the expected communication pattern or physical limitations that the system must obey. The work in [16] developed an intrusion detection system that has the ability to classify substation events as real or fake using previous data.

- Simulation Model-based attack detection

In [17], a framework was designed to increase the integrity of Automatic Generation Control (AGC) data using the help of synthetic data. In [18], the authors detected false data injection attacks on PMUs and smart meters through one-step-ahead state prediction techniques. The work in [19] applies offline checkup for anomalies by combining the overall system variables using Markov models. [20] proposes the use of on-the-fly power system dynamics simulation (i.e., power flow analysis) for command authentication. [21] proposed a method to authenticate commands by faster than real time simulation using a real time model of the affected subsystem. [22] presents a multi-attribute intrusion detection system specific for Supervisory Control and Data Acquisition (SCADA) systems.

Since the SMV protocol has specific requirements, such as high data transmission rate, the authors will analyze the SMV protocol and discuss possible attacks in the following section before outlining the countermeasures and the proposed defense mechanisms.

3. The Sampled Measure Values Protocol

3.1. Structure of the SMV Datagram

An SMV datagram follows a modified Abstract Syntax Notation One (ASN.1) Basic Encoding Rules (BER) Tag/Length pair encoding scheme [23]. Figure 2 shows the structure of an SMV datagram. It is composed of three parts, namely a header, a payload and a checksum. The SMV header starts with a destination address, which is one of the multicast MAC addresses listed in the IEC 61850 standard that start with 01 0C CD 04, followed by the source device's MAC address. IEC 61850-9-2, which covers the specifications for SMV messaging [24], also recommends the implementation of Virtual Lan (VLAN) and priority tagging based on IEEE 802.1Q followed by the Ethernet type, which is 0x88BA for SMV messages [25].

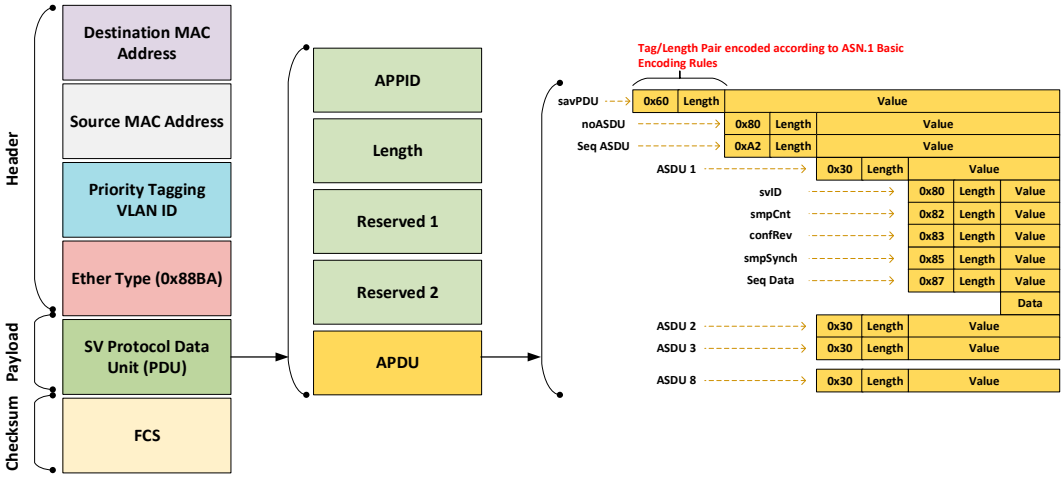


Figure 2. Structure of the SMV Datagram.

The payload of an SMV packet is composed of an SV Protocol Data Unit (SVPDU). In its turn, the SVPDU is composed of an Application ID (APPID) field, which is a unique identifier used to classify/distinguish SMV messages, a Length field, which is a hexadecimal number that represents the length of the entire SVPDU, two reserved fields for future use, and an Application Protocol Data Unit (APDU) field.

In an APDU, there can be several Application Service Data Units (ASDU)s. These are objects containing information relevant for an individual node. Using several ASDUs allows a single device to send or receive physical sensor data from several nodes within the power system in the same SMV message. The number of ASDUs is stored in noASDU [26]. Each ASDU then contains seven subfields which are as follows [27]:

- svID: Sample Value ID;
- SmpCnt: Counter that increments each time a new sampling value is taken;
- ConfRev: Value that indicates the number of configuration changes;
- SmpSynch: A Boolean value that is true if the SV is synchronized by a clock signal and false if it is not;
- Seq Data: Sequence of data;
- Data: The actual dataset.

3.2. Advantages of an SMV Process Bus

Conventionally, a device called Merging Unit (MU) is used to convert voltage and current readings from potential and current transformers, respectively, into digital data and publish them as Sampled Measured Values. Recently, modern devices called non-conventional instrument transformers have an Ethernet interface and publish measurements as SMV packets directly to the process bus. SMV packets are broadcast messages published on a Local Area Network (LAN) that uses switched Ethernet communication. In depth details are given in IEC 61850-9-2 and the more recent Light Edition of the Standard. SMV messages are mapped to the Data Link Layer of the Open System Interconnect (OSI) model. This model was developed in order to provide interoperability to computer networks, and is divided into seven layers. The Data Link layer is in fact Layer 2 of the OSI model. SMV messages are broadcast messages and an svID field is utilized to distinguish and classify them. Applications utilizing SMV messaging follow the publisher/subscriber communication scheme. That is, on one hand, MUs publish SMV messages to the process bus, and on the other, Intelligent Electronic Devices (IED)s, the devices that have the control logic, subscribe to one or more SMV message. The IED receives only the messages it is configured to subscribe to, rather than receiving all SMV messages.

This concept of the process bus and SMV messaging has a lot of advantages in the context of microgrid control:

- In terms of system monitoring, the messages stream on a networked process bus will be composed of well-structured SMV packets, each with its unique identifier. This will allow for easy classification and improve the visibility of the power network [2].
- In terms of cost, the networked process bus replaces point-to-point analogue connections between devices as shown in Figure 1b. This will drastically reduce the copper wiring involved, thus reducing costs and saving on installation and maintenance.
- In terms of control, the SMV process bus facilitates the migration to decentralized and/or distributed control from the current centralized control scheme, which will enhance the reliability of the power network. Centralized control is characterized by a single server, which receives measurements from all the sensors in a system and gives actuation commands. A centralized server is considered as a single-point-of-failure (i.e., a bottleneck). Decentralized and distributed control avoid this vulnerability.
- In terms of data handling, the SMV process bus provides vendor-independence. That is, IEC 61850 SMV-compliant devices should abide by the message structure shown in Figure 2. Therefore, achieving data interoperability.

4. Threat Identification and Current Mitigation Strategies

In this section, the IEC 61850 Sampled Measured Values protocol is exploited. We start by a discussion of the sources of attacks, followed by an explanation of how the attacks work, and a discussion on viable countermeasures.

SMV messages are directly mapped to Layer 2 of the OSI model. Therefore, they are (1) non-routable, and (2) non-blocking. In the following, we discuss the threats that could compromise the SMV process bus.

Attacks on IEC 61850 network could originate from several sources. These include (a) inside personnel with access to the IEC 61850 network. These individuals have the capability to infect the system with malware either intentionally (e.g., disgruntled employee), or unintentionally (e.g., improper use of infected devices). (b) Attacks can originate from the supply chain. That is, malware infection occurs during a device's production phase [28]. (c) An attacker can achieve access to a local area network through corporate networks or personal devices of the employees. Attackers can use techniques, such as password cracking, spam emails, and backdoors [29]. Two successful attacks on industrial control systems and power system control networks, which were publicly disclosed, are the Stuxnet incident, where the attacks targeted PLCs in a nuclear plant, and the Crash Override incident, where attackers targeted substations' networks in Ukraine [30,31]. Once an attacker successfully penetrates the network, (s)he has the potential to launch several attacks, as described below.

4.1. Denial of Service (DoS)

DoS refers to the time when a device (or a user) is denied access to a service by an attacker. A possible method to launch a DoS attack could be by flooding the network to delay delivery of packets beyond the critical flooding rate. This is achieved by clogging the communication channel and draining the computational resources of the communicating devices. Network flooding can occur on the Network or the Application layer of the OSI model. In this manuscript, we are interested in DoS attacks on the process bus. As can be seen in Figure 3, there are two ways to flood a process bus with unwanted traffic. The first is by transmitting SMV packets with a high transmission rate, which will delay the transmission of legitimate SMV packets beyond 3 ms. Utilizing the APPID of SMV messages and monitoring the rate of each message will allow the detection of this for of DoS. Only two publishing rate are allowed for SMV messages per the recommendations of the IEC 61850-9-2LE standard. These are: 80 samples/AC cycle or 256 samples/AC cycle for 50 Hz and 60 Hz systems, respectively. Violating these rates for a certain SMV stream will allow the detection of a DoS attack.

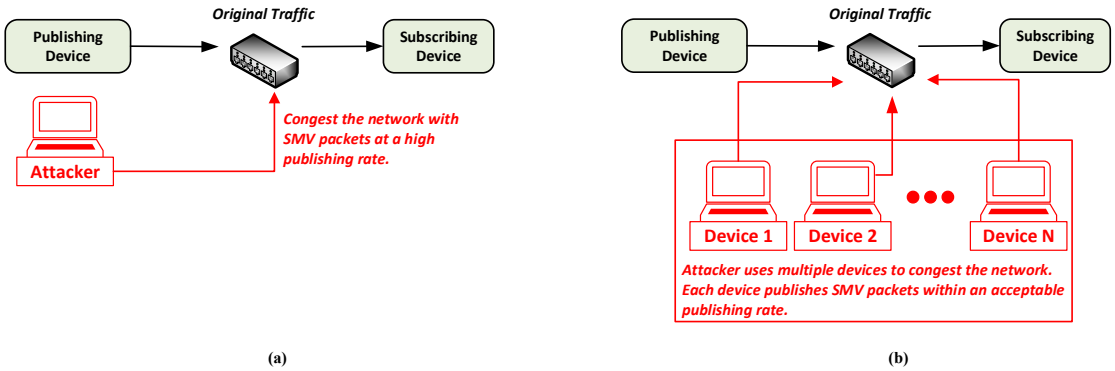


Figure 3. (**a**) Denial of service; (**b**) distributed denial of service.

However, an attacker could still flood the network by utilizing more than one publisher to transmit SMV messages without violating the aforementioned rates. To defend against that, a list associating svIDs and the data in the svPDU should be created. This will help identify undefined svIDs. Additionally, each SMV is associated with a sample counter. Therefore, when an attacker uses

unregistered IDs, repeated or out-of-sequence sample counters could be detected. This is because an attacker cannot block messages on Layer 2 of the OSI model.

4.2. Eavesdropping

As shown in Figure 4, eavesdropping refers to when an attacker passively sniffs network traffic. Thus, the term eavesdrop. Although this is a passive attack, the attacker performs eavesdropping attacks to increase his/her advantage by gaining knowledge about the power system. Starting from the IP Layer of the OSI model, encryption can be used to defend against eavesdropping using techniques, such as Secure HTTP (HTTPS), Secure File Transfer Protocol (SFTP), and Secure Shell (SSH). Scanning network cards in promiscuous mode as well as monitoring network traffic are viable defensive measures [32].

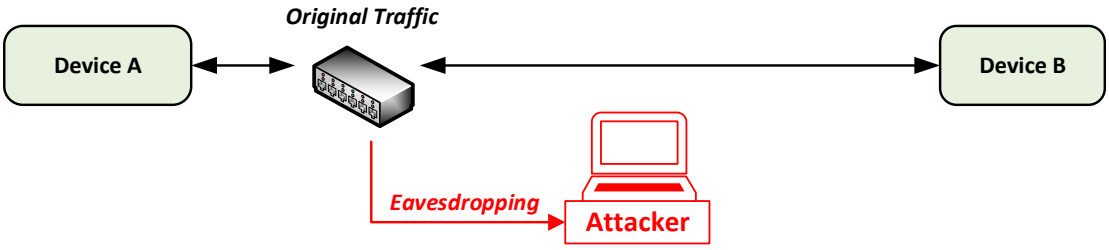


Figure 4. Eavesdropping attack.

However, IEC 62351-6 standard, which dictates the security recommendations for SMV messages, recommends that for "applications using SMV and requiring 3 ms response times, multicast configurations, and low CPU overhead, encryption is not recommended" [24]. Accordingly, SMV packets are left without encryption and eavesdropping on the process bus is a threat, which gives attackers an advantage.

4.3. Replay

In this attack, an attacker listens on the network and records the transmitted packets to later replay them on the network. This is shown in Figure 5. Given that SMV packets are non-blocking and that there exists a sample counter field for each packet, replayed SMV packets could be detected by monitoring the process bus for sample counters that are repeated or out-of-sequence.

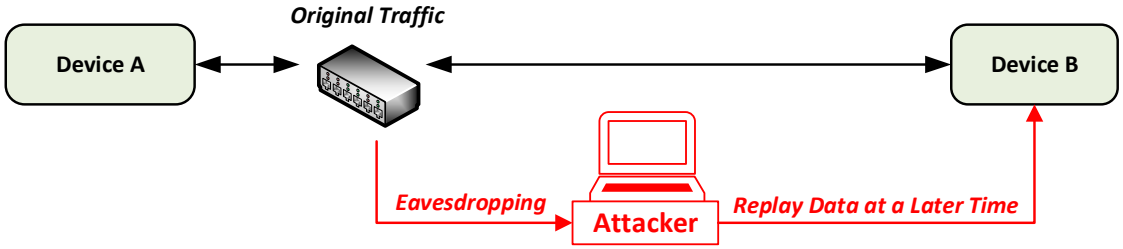


Figure 5. Replay attack.

4.4. Man-in-the-Middle (MITM)

In an MITM attack, an attacker intercepts traffic between two devices through Address Resolution Protocol (ARP) Poisoning. ARP is a communication protocol, which is used to convert IP addresses into MAC addresses [33]. As shown in Figure 6, in ARP Poisoning, an ARP reply is sent by an attacker to change the IP to MAC addresses mapping on a given the network. Therefore, there will be incorrect association between IP and MAC addresses. This will allow an attacker to interrupt the messages transmitted between the communicating devices, and to manipulate them.

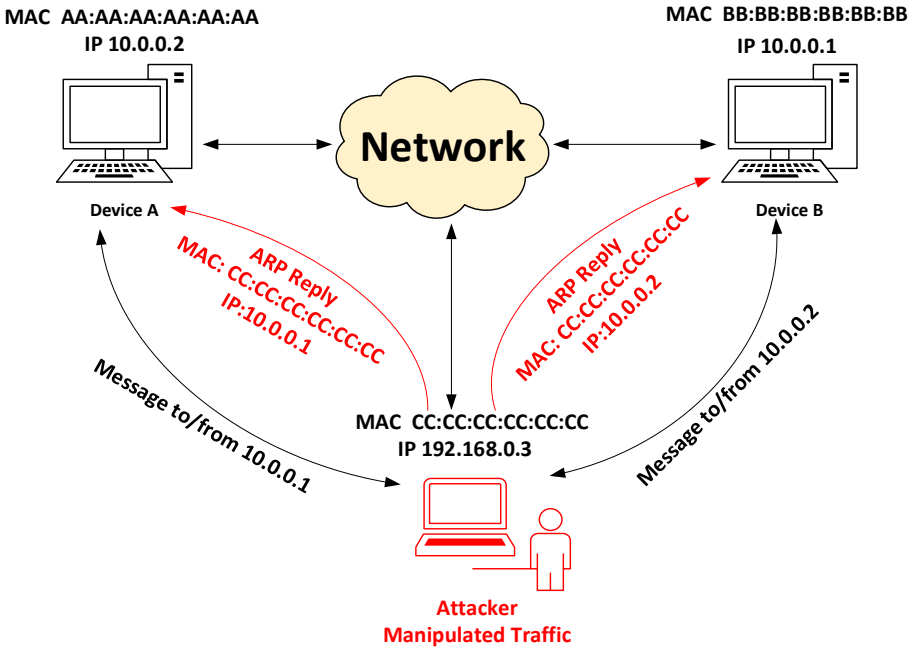


Figure 6. ARP Poisoning and MITM attack.

A Man-in-the-Middle attack is not possible on the SMV protocol because SMV messages are broadcasted on Layer 2. An exception exists if an insider compromised the network.

4.5. SMV Spoofing

In an SMV spoofing attack, an SMV packet with changed data field(s) is transmitted to the process bus by an attacker's device, while keeping the MAC address of the legitimate publisher unchanged, as depicted in Figure 7. This is feasible since SMV packets are not encrypted. An attacker can record an SMV packet, decode its contents, manipulate certain fields, and retransmit the spoofed message to the process bus.

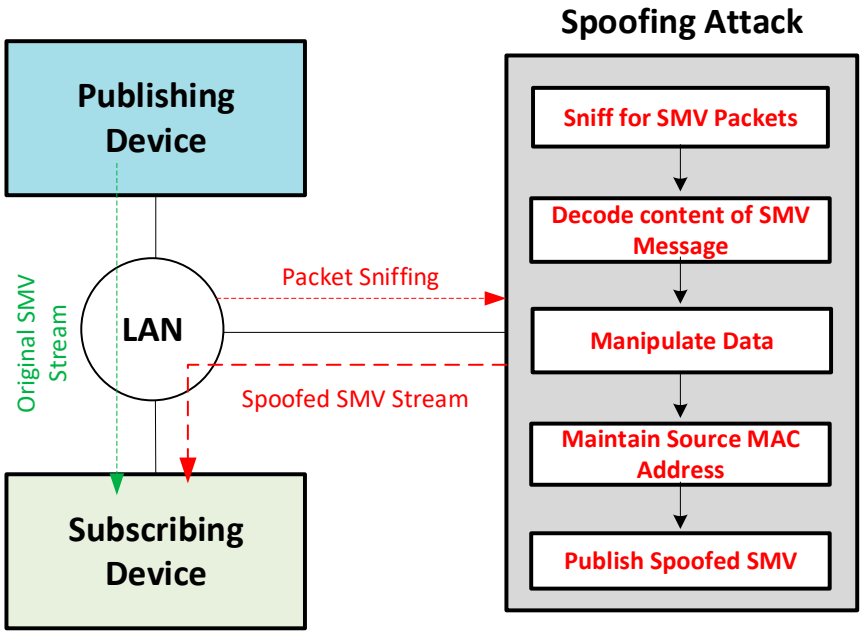


Figure 7. SMV spoofing attack.

To counterfeit is attack, the IEC 62351 standard recommends the use of RSA-based signatures for authenticating SMV packets and ensuring their integrity. However, since RSA-based authentication is computationally expensive, it is unfitting for the SMV protocol. This is because the SMV protocol requires an end-to-end time delay of utmost 3 ms [34,35].

IEC 62351 ale recommends the use of a Hash-based Message Authentication Code (HMAC). HMAC was found to have an average latency in the 100 microseconds range. As indicated in [36], this latency increases drastically with the increase in the size of the packet. The experiments on latency that were reported in the literature were executed on high-end processors, such as Intel core i7 processors and FPGAs [36,37]. Such processing power is expensive and is seldom an option in IEDs and MUs, especially not on current devices I the field.

Additionally, in applications requiring broadcast communication, such as the process bus, non-repudiation is not provided by MAC authentication [36]. For instance, any device can craft a message with a spoofed MAC address and publish it mimicking the original publisher. Here, the attacker needs to have knowledge about the secret key. Attacks reported in the literature, such as length extension attacks, internal state attacks, key recovery attacks, and forgery attacks can make this possible [38–40].

Although an attacker can publish a spoofed SMV message, and a subscriber can verify it, the original SMV packet cannot be blocked from being also published by the legitimate publisher. In such an event, the subscribing device will receive an SMV packet that has a repeated or an out-of-sequence sample counter. The issue at hand could confuse the subscribing device since it cannot identify the legitimate packet. Replay attacks will put the subscribing devices in the same situation.

5. Feasibility Study and Detailed Description of Using Neural-Network-Forecasters in Power Systems for Detection of SMV Spoofing Attacks

In the aforementioned discussion, it can be noticed that the SMV protocol structure along with the related IEC 62351-6 recommendations ensure the safety and resiliency of the SMV process bus against various types of cyber-attacks. Yet, knowledgeable attackers with enough time and effort could successfully interrupt a power system's operation and trigger a sequence of unwanted events (e.g., tripping Circuit breakers, which leads to a cascaded failure).

In this section we will address the following:

- Firstly, describe the system understudy and the flow/exchange of data within this system;
- Secondly, explain the SMV spoofing attack formulation;
- Finally, introduce a bi-layer Neural-Network-forecasting algorithm to detect and prevent such attacks.

5.1. System Understudy

The system being studied is a DC microgrid tied to an AC microgrid by a bidirectional inverter, introduced in Figure 8. The AC microgrid comprises:

- Generators 1 and 2 with ratings 13.8 KVA and 10.3 KVA, respectively. Both are 230 V;
- Two variable loads, each has 10 steps (300 W to 3 kW). Both loads were set to 600 W during the experiment;
- Measurement and protection devices: each bus has a 3-phase input/output relay, and each phase has its own potential and current transformer for measurements.

The DC side (i.e., DC microgrid) contains:

- A DC source resembling a DER;
- A 60 ohms constant DC load and a 12 ohms resistive pulse DC load.

The two microgrids exchange power to maintain demand/generation balance, when necessary. This takes place by adjusting I_d (i.e., the current direct component) within the inverter controller,

which could be observed in Figure 9. The typical dq transformation is utilized to acquire the abc currents' references from the dq currents' references. Then these generated abc currents' references are compared to the actual measured abc currents. The output errors are used to adjust the switching and accordingly adjust the actual abc currents to the desired references through a hysteresis current controller. This controller operates the voltage source inverter in a current control mode. The software of all the MUs and the IEDs were coded in C++ on embedded microcontrollers running on a Linux-kernel.

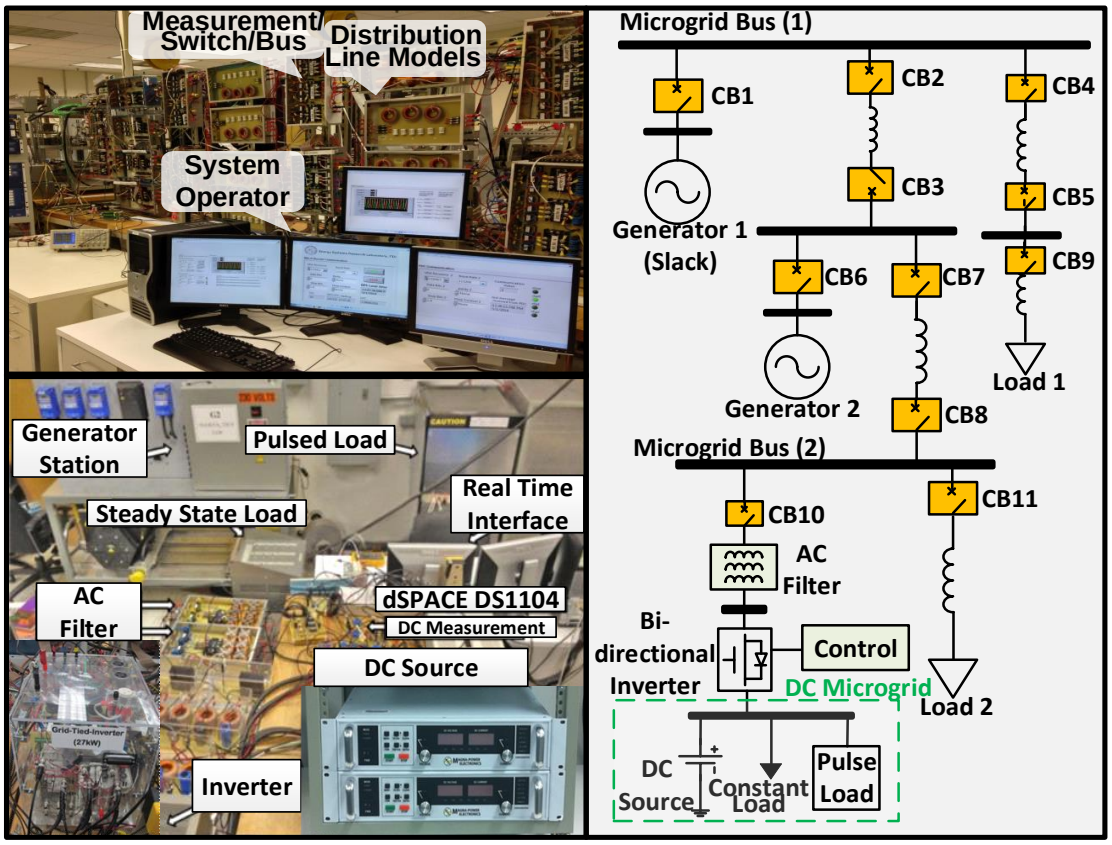


Figure 8. General overview of the AC/DC microgrid (CB: Circuit Breaker).

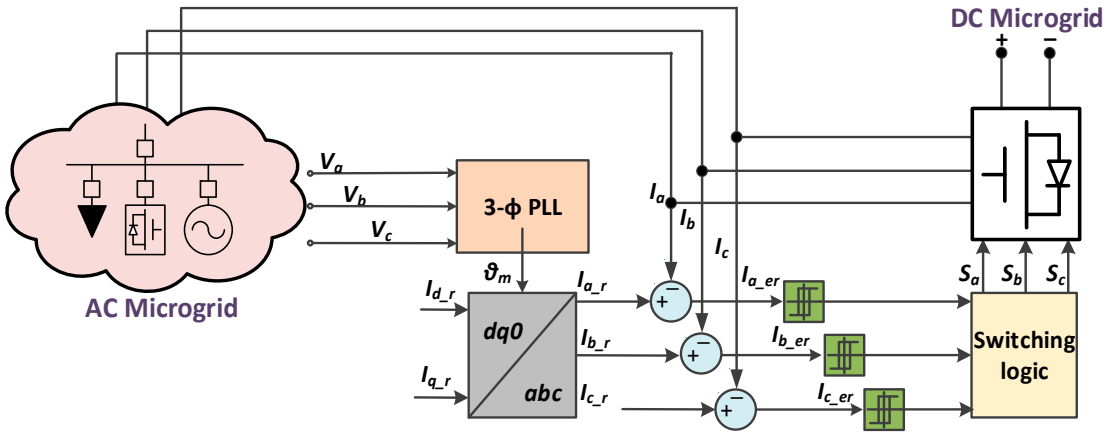


Figure 9. Control of the bi-directional inverter.

5.2. SMV Spoofing Attack Scenario

To have effective detection mechanisms for cyber-attacks, we proposed to add a physical-based layer forecaster to the cyber layer, as shown in Figure 10. The typical cyber layer checks the standard

conditions (e.g., repeated sample counters). However, in our algorithm, the physical-based layer relies on the incoming SMV measurements from the physical system (e.g., digitized currents) and utilizes the Neural-Network-Forecaster to detect and prevent an SMV spoofing attack.

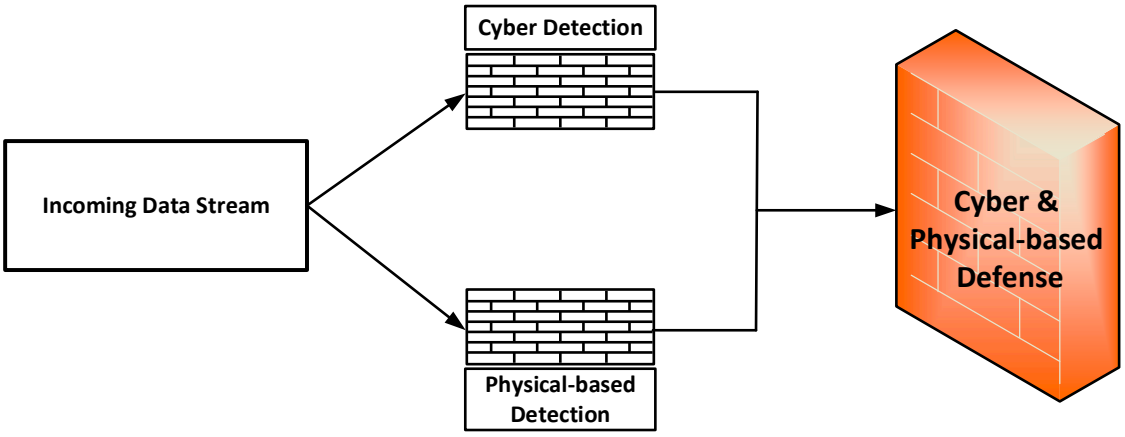


Figure 10. Cyber-physical detection of attacks.

Having the proper knowledge of the structure of the SMV datagram, it is feasible to create a malware that targets precisely the measurements being sent through the process bus. The breakdown of the malware attack stages could be seen in Figure 11. Initially, on the LAN, the malware sniffs packets and looks for an SMV packet. Once an SMV Ethertype is verified, the malware takes out the header information and stores it. Then it will start publishing spoofed packets, containing false measurements, through the process bus using the stored header fields to make the message appears legitimate. In other words, the malware spoofs the header of an SMV message and uses it to construct a new SMV datagram with a fake data field, which is the field that contains the measurements. Finally, it publishes it to the process bus. In this paper, Python along with packet crafting and network sniffing libraries from Scapy were used to write/code the malware.

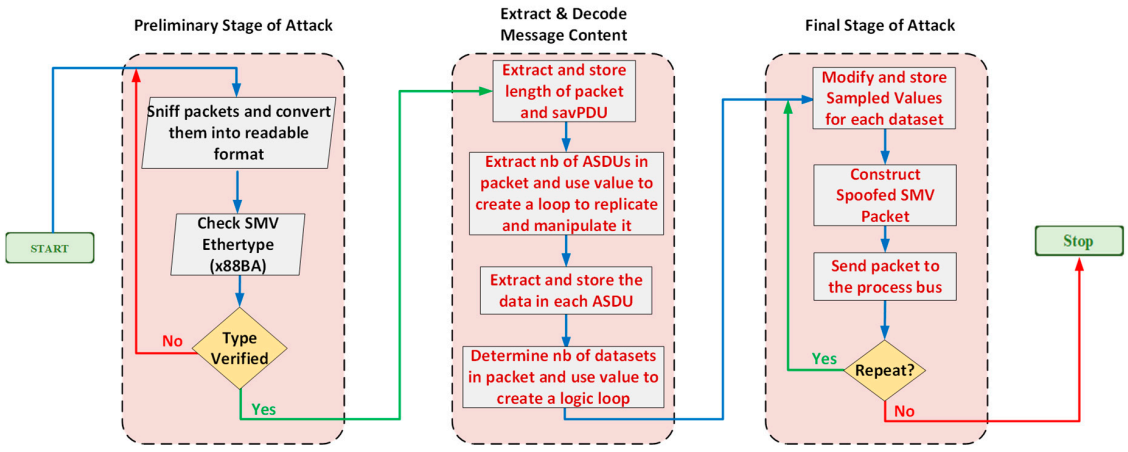


Figure 11. Malware Development Process in Python using libraries for Scapy.

5.3. The Bi-Layer Neural-Network-Forecasting Algorithm to Detect Spoofed SMV Attack

Earlier it was discussed that injecting a spoofed SMV will cause an out of sequence or a repeated sample counter, which makes the detection seems to be trivial. However, it should be noted that the subscribing-device will be unable to decide which message contains the true measurement value. To re-iterate, the spoofed message will have the MAC address of the original sender. Therefore, this

subsection examines the feasibility of utilizing neural-network-forecasters to distinguish between spoofed and legitimate messages.

5.3.1. Layer 1: The Neural-Network-Forecaster

In the framework proposed, shown in Figure 12, there are mainly publishers, subscribers, and the process bus. The publisher (e.g., MU) receives the instantaneous analogue measurements from potential (PTs) and/or current transformers (CTs), then digitizes them via A/D converter, and finally publishes them as SMV packets over the LAN to the process bus. The subscriber (e.g., the control agent/IED) subscribes to the process bus with a unique SMV ID to access these measurements. During this process, if the control agent detected an anomaly (e.g., a sample counter is repeated), it passes the suspected two samples to the first layer, which is the Neural-Network-Forecaster (NNF).

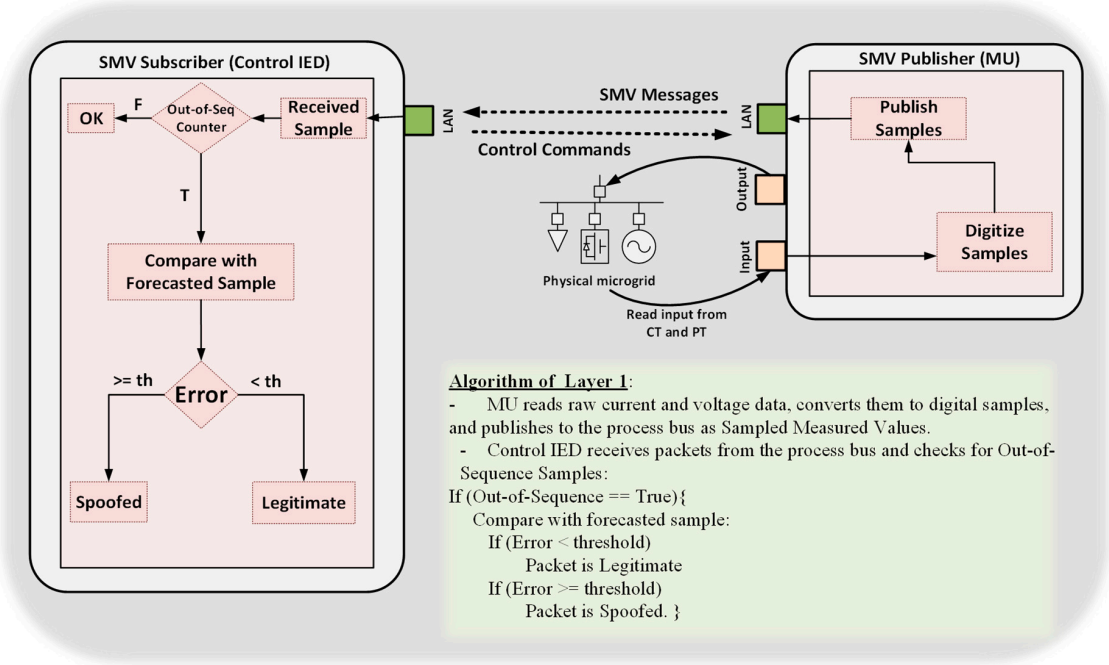


Figure 12. Algorithm of Layer 1.

There are four main categories under which cybersecurity work could be classified:

(1) Prevention: to develop cyber security solutions that ensure resilient delivery of energy. Work in this area includes authentication, encryption, key management and storage, and others.
(2) Detection: to develop cyber security solutions that are capable of identifying the occurrence of cyber incidents. Work in this area includes anomaly detection, detection of false (or spoofed) data injection attacks, detection of password cracking attempts, and others.
(3) Response: to develop cyber security solutions that take appropriate measures in response to the detected cyber incidents.
(4) Recovery: to develop tools and techniques that allow the restoration of energy and other services that were lost due to cyber incidents.

In this manuscript, the proposed algorithm falls into the second and third categories. This is because it detects spoofed SMV messages and then responds by discarding the spoofed messages and utilizing the legitimate ones.

The detection and response functionality of this layer could be briefly laid out as follows: the NNF predicts the future value of the incoming measurements using N-previous samples (e.g., digitized measurements from CTs/PTs). Then an error comparison will be performed between the received

suspected samples from the control agent and the forecasted one from the NNF. Finally, the received SMV that comes out to be associated with an error greater than a predefined limit will be discarded and the other SMV is considered as legitimate.

In this work, we consider the current waveform as a single-variable time-series. Generally speaking, let x be the variable representing the value of the current and t be time. Therefore, moving backward from time t, our aim is to predict the value of the current at some point in the future $(t + s\Delta t)$, where Δt is the time-step used in sampling:

$$x(t + s\Delta t) = f(\ x(t),\ x(t - \Delta t),\ x(t - 2\Delta t),\ \ldots.) \tag{1}$$

Since we are forecasting the value of one incoming sample, the forecasting window $s = 1$ in Equation (1). Therefore, Equation (1) becomes:

$$x(t + \Delta t) = f(\ x(t),\ x(t - \Delta t),\ x(t - 2\Delta t),\ \ldots,\ x(t - n\Delta t) \tag{2}$$

Eventually, the problem becomes a function approximation problem, where n previous values of x are utilized as input to predict the future values of x.

Before we proceed, it is important to mention the difference between time-series analysis and time-series prediction/forecasting. In time-series analysis, which is also referred to as descriptive modelling, a time-series is modelled in order to determine its components in terms of seasonal patterns, trends, relation to external factors, etc. Whereas, time series forecasting uses the information in a time series (e.g., history data) to forecast future values of that series [41]. Although understanding a dataset (i.e., time-series analysis) can help enhance forecasting, it is not required and can result in a large technical investment in time and expertise not directly aligned with the desired outcome, which is forecasting the future [41].

To solve the problem of approximating f, there a several methods that could be used. The most popular methods in the literature are Auto Regression (AR), Moving Average (MA), Autoregressive Moving Average (ARIMA) and ARIMA-based techniques (e.g., Seasonal ARIMA (SARIMA), SARIMA with Exogenous Regressors (SARIMAX), ...). Put in simple terms, these methods approximate f as a liner function of the differenced observations and residual errors at prior time steps. Other methods, such as Simple Exponential Smoothing (SES), and Holt Winter's Exponential Smoothing (HWES) approximate f as an exponentially weighted linear function of observations at prior time steps.

The above-mentioned methods are best used in order to give an indication of the trend that the time-series follows. For example, they could be utilized to forecast an increase or a decrease in sales of a certain brand, or an increase or a decrease in demand for energy in a certain area.

In our specific application, the current data we are trying to forecast is highly non-linear. Also, we are interested in predicting the value of the current in an incoming SMV sample. In order to do that with very high accuracy, we need to use non-linear approximation methods for f. Since Lapedes and Farber published their work in 1987, neural networks have repeatedly proven themselves as promising forecasting tools [42]. Not only do NNs capture the complex underlying relationship in many real-world problems, but also, they can be retrained and adapted, online, to learn new characteristics of the data that could appear with time [43].

According to the discussion above, a feedforward neural network was selected as the forecaster in our proposed algorithm.

The NNF contains three main layers: input, hidden, and output layers. Let the input layer be $X1[I + 1] = \{x1_1, x1_2, \ldots, x1_i, 1\}$, where $i \in \{1, I\}$, I is the input layer dimension, and $x1_{I+1} = 1$ is its bias. Within this layer, inputs are multiplied by weights $(w1_{h,i})$ to obtain $N1[H] = \{n1_1, n1_2, \ldots, x1_h\}$ vector, which will be inputted to the hidden layer. H is the dimension of the hidden layer and $h \in \{1, H\}$. $N1$ vector's elements are being calculated according to (3):

$$n1_h = \sum_{i=1}^{I+1} x1_i \times w1_{h,i} \tag{3}$$

Then, each of elements will be handled via a neuron in the next layer (i.e., hidden layer). This will result in vector $X2[H+1] = \{x2_1, x2_2, \ldots, x1_h, 1\}$. $X2$'s elements are then obtained through (4) and (5):

$$x2_h = \frac{2}{1+e^{-n2_h}} - 1 \tag{4}$$

$$x2_{H+1} = 1 \tag{5}$$

The bias of the hidden layer is $x2_{H+1}$. Equation (4) is the activation function for the neurons in the hidden layer. It is a sigmoid function.

Lastly, $X2$ gets processed though the output layer to obtain the output vector, $S[K] = \{s_1, s_2, \ldots, s_k\}$. The output vector's elements are obtained according to (6):

$$s_k = \frac{2}{1+e^{-\sum_{h=1}^{H+!} x2_h \times w2_{h,k}}} - 1 \tag{6}$$

where K is the dimension of S and $k \in \{1, K\}$. $w2_{h,k}$ represent the weights between the hidden and output layers. The sigmoid function is also utilized for the neuron in the output layer as in (4).

In order to train, verify, and test the NNF, various data sets were generated from the MG described earlier in Section 5.1, for the NNF to be able to distinguish between legitimate and spoofed SMV messages, before being implemented in the hardware experiment. The data sets correspond to normal operation of the MG, various fault events such as tripping of transmission lines, and contingencies such as disconnection of generating units. All the data sets were collected in a history log file. Portion of the data was gathered directly from the hardware experiment, and to generate a rich set of data, a simulation approach was used. In this approach a Simulink based model of the MG was used to acquire the data for the rest of the contingency and fault events. Further details regarding model accuracy is discussed in Section 6.

NNF Structure: in this paper, the NN utilizes 20-previous samples to predict the value of sample 21. Hence, the NNF has 20 neurons in the input layer and one in its output layer. Ten neurons were utilized in the hidden layer. The reason behind this is founded on an empirical study of the NNF accuracy verses its computational time and it came out that using 20, 10, and 1 neuron/s in the input, hidden, and output layers, respectively, would yield least computational time with highest accuracy.

NN Type Utilized: the neural network used is a feed forward neural network trained with the sliding window approach. Starting from the 1st sample, 20-samples were counted as input and sample No. 21 was set to be forecasted (i.e., set as the target output). Next, the window slide/ move one sample, where the input samples will then become No. 2 to 21, inclusive, and the forecasted sample now (i.e., target output) is sample No. 22, and the sliding-forecasting process continues. The general process is explained in Figure 13.

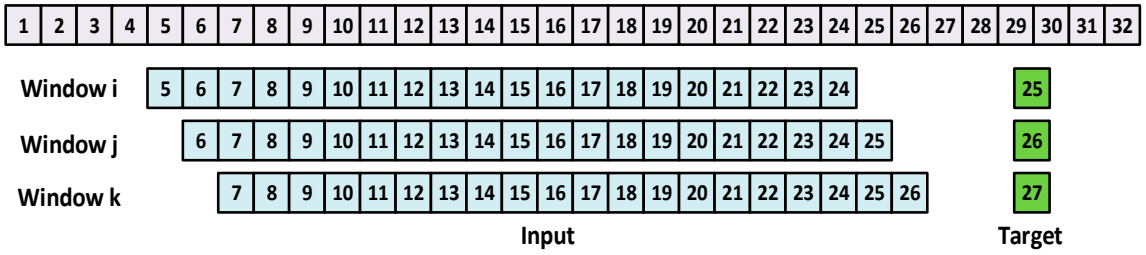


Figure 13. Sliding window approach [3].

Setting the Decision Threshold: earlier we mentioned that an error comparison will be performed between the received suspected samples from the control agent and the forecasted one from the NNF. In order to set the decision threshold whether an SMV packet is legitimate or not, a Monte-Carlo-analysis

was done for over 2500 test cases for current waveforms that were gathered from Load 1 (could be seen in Figure 8). Each test case contains 2001 samples of measurement.

In order to train the NNF to distinguish between legitimate and fake samples, for each of the test cases' data, at different instants, fake random data was inserted according to (7). This fake data that was randomly injected ranged between 4 and –4 Amps, which is 1.5 times the rated current of the MG system described earlier:

$$fake\ data = (B - A) \times rand() + A \tag{7}$$

where $A = -4$, $B = 4$, and $rand()$ is pseudo-random number function that generates a random value between 0 and 1. Then, the error between the forecasted sample of the neural network and the fake sample inserted was recorded. It was found that the highest accuracy for detecting a spoofed sample is at a 2% decision threshold.

Layer 1 was implemented and tested on the control IED managing CB 9 of Load 1. Fake SMV packets with current measurement were injected according to (7). In Figure 14, it can be observed that the NNF has a high forecasting accuracy for the incoming measurement value. The conducted study reveals that utilizing an NNF has a good potential to identify and respond to spoofed messages based on learning the system's characteristics. However, over time, forecasters are vulnerable to the buildup of the forecasting error. Knowing that, attacker/s could publish spoofed messages such that they don't exceeds the 2% threshold. If the control IED uses the fake SMV message, the buffer used by the NNF to forecast the next measurements will have an incorrect misleading entry. In this case, the control IED will become indecisive. Based on the results from Layer 1, when the control IED receives an SMV with a repeated or out of sequence sample counter, and is unable to distinguish the spoofed message, it will activate Layer 2 of the defense mechanism. The accumulation of the forecasting error is shown later along with the results of Layer 2, in Figure 17.

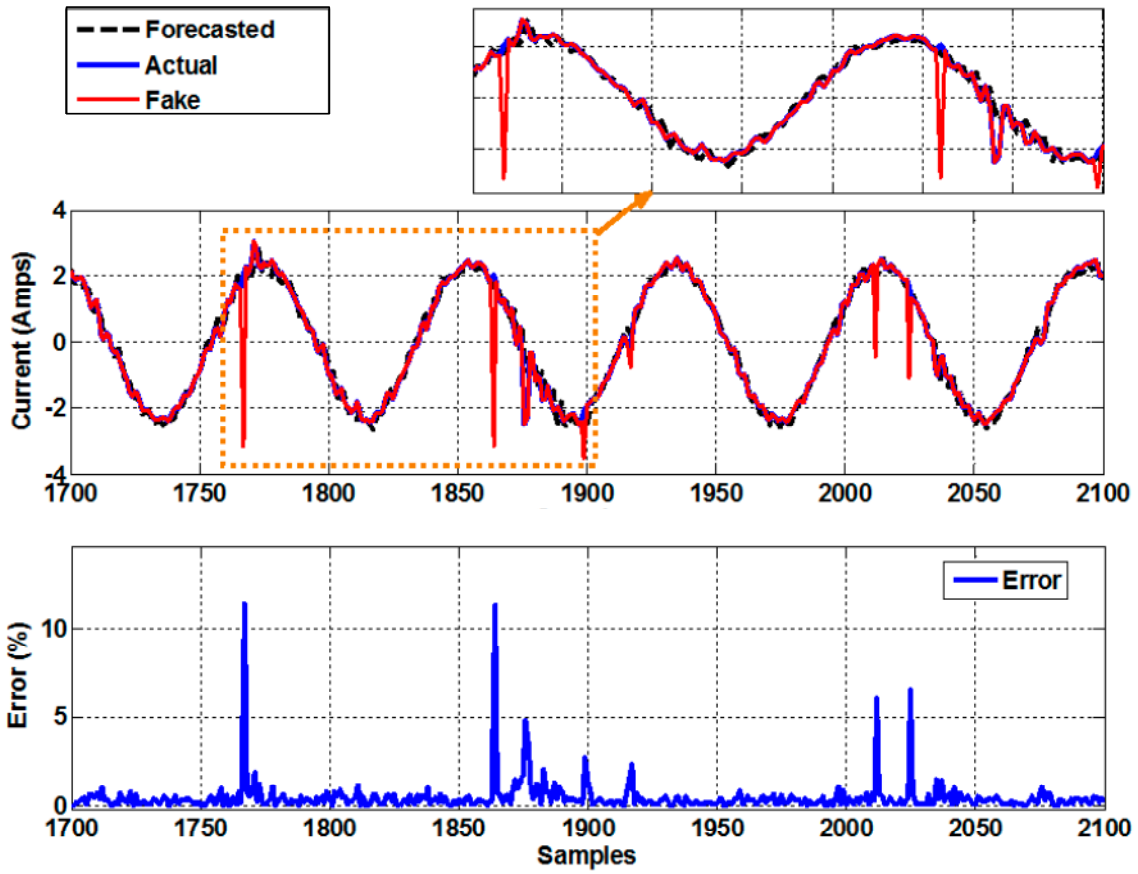


Figure 14. Performance of the neural network forecaster [3].

5.3.2. Layer 2: Enhancing the Resiliency of the Neural Network Forecaster

- Attacker Sends One Spoofed Message

As mentioned earlier, if the IED uses the fake message to forecast the new sample, the buffer used to forecast the incoming measurements will have a misleading entry. Eventually the accumulation of the forecasting error with time, will lead the NN-F to start discarding legitimate messages, thus, disrupting the control operation of the IED. To re-iterate, when an attacker sends one spoofed message that doesn't violate the detection threshold of the NN-F, the control IED will have two SMV packets, but it cannot decide which one is legitimate.

To address this problem, the IED will hold onto the suspected samples. For the sake of simplicity, let us call them sample A and sample B. The IED will then receive 5 new consecutive samples. For every one of the new samples, the error between the forecasted sample and the received one will be recorded. Then the error derivative will be calculated. If the derivative indicates that the error is increasing, a flag is issued. At this stage, thread 2 of the control agent is activated. Thread 2 will create two datasets:

(1) The first set contains one of the suspected samples (Sample A), five previous samples, and the five new samples, which were monitored.
(2) The second dataset contains the other suspected sample (Sample B), five previous samples, and the five new monitored samples.

The IED will then calculate the mean, the variance and the standard deviation of these datasets. These datasets are all from the measurements received over the network.

Simultaneously, the control agent sends the flag and the position of the suspected sample to thread 2 of the MU agent over an out-of-band trusted network. The MU will assemble a similar dataset composed of the suspected sample, five previous, and five later samples. However, this dataset will be from the local digitized data that hasn't been altered (i.e., these messages are trusted).

Note: In this work, we trust that the measurements that are directly read by the MU through Analogue Inputs are correct. The MU is the source of collection of the measurements. There are still cases where, for example, saturation of CTs and/or PTs occur resulting in bad data (as a result of CT saturation and not cyber-attack). However, there are solutions that could detect such incidents, such as the solution presented in [8]. In this manuscript, we assume that the data digitized by and saved in the MU's memory is correct.

Similarly, thread 2 of the MU agent will calculate the mean, the variance, and the standard deviation of the assembled (trusted) dataset, and will send them to the control agent over the trusted network. Finally, the control agent will compare the received set of indicators with the results of the set containing Sample A and the results of the set containing Sample B. The dataset which has matching statistical indicators is the one bearing the legitimate sample. This process is shown in Figure 15.

- Attacker simulates a fault condition or hides a fault condition by replaying normal condition.

The NN-F is trained to forecast measurement values under fault conditions and other contingencies. The attacker could inject spoofed messages simulating a fake fault and the NN-F could identify them as legitimate. The attacker simulates a fault, one sample after the other. The attacker sends the 1st packet, which doesn't violate the threshold. If the IED uses it, in the next time step, the NN-F will forecast a higher current value (fault) and will discard the legitimate message.

In this case, if two repeated or out-of-sequence samples are detected after each other, then Layer 2 is activated to decide if this is a real fault or not. Similar to Case 1, the IED will create two datasets. One dataset has the samples that correspond to the fake fault, while the other has the normal measurements. The dataset which has matching statistical indicators with the MU will be the legitimate one. Conversely, the attacker could hide an actual fault by replaying measurements corresponding to the normal condition. Following the same procedure, the IED could decide which message stream is valid and which is not.

- Selection of the Statistical Indicators

The purpose of the statistical module is to create a small yet indicative feature vector of the two datasets generated by the control IED and the MU. The selected statistical features for the two datasets are the mean, variance, and standard deviation. In order to select the appropriate number of samples for the IED to hold on to, the following study was performed. Consider the current data shown in Figure 16a. Various spoofed data were injected at the peak of the sine wave ranging between ±0.5 to ±1.5 times of that sample.

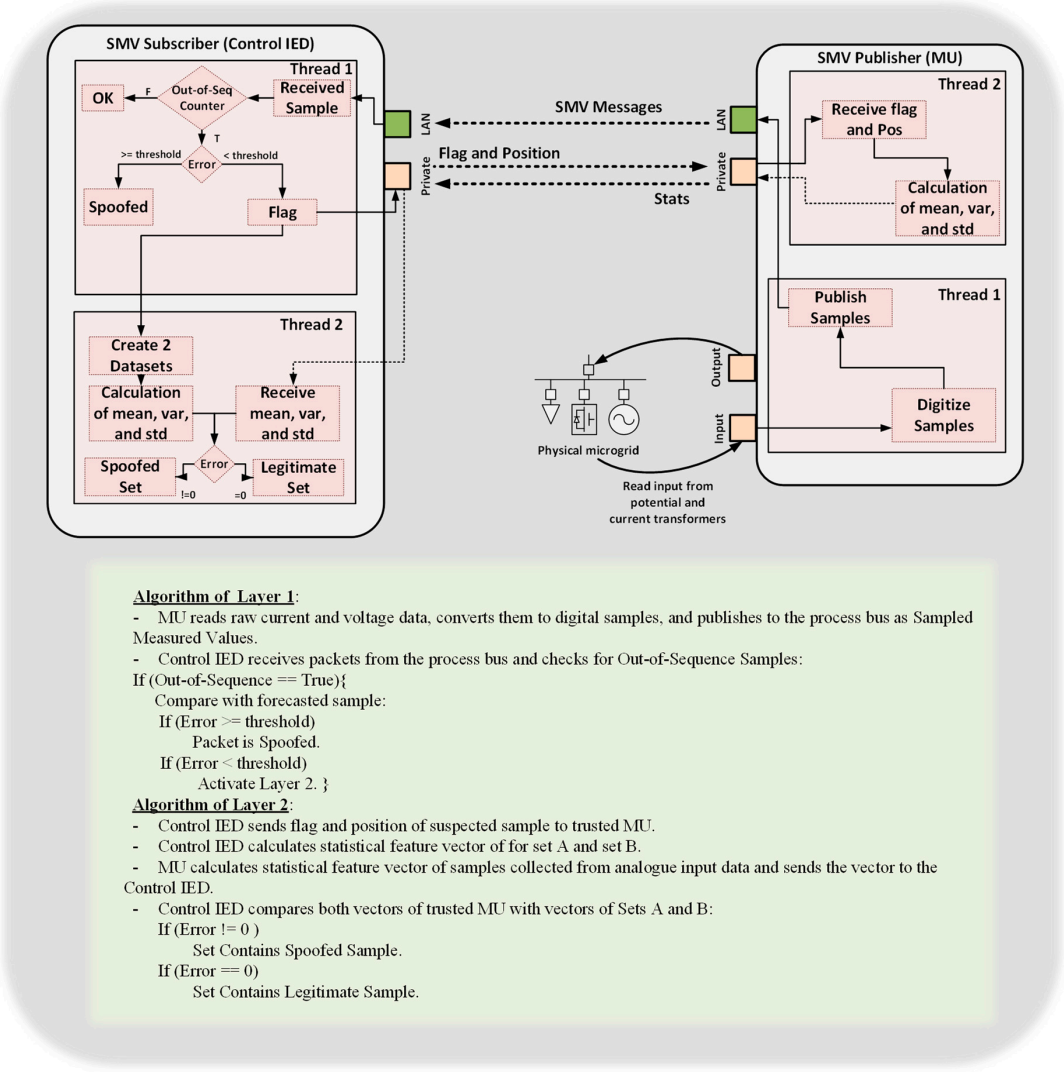
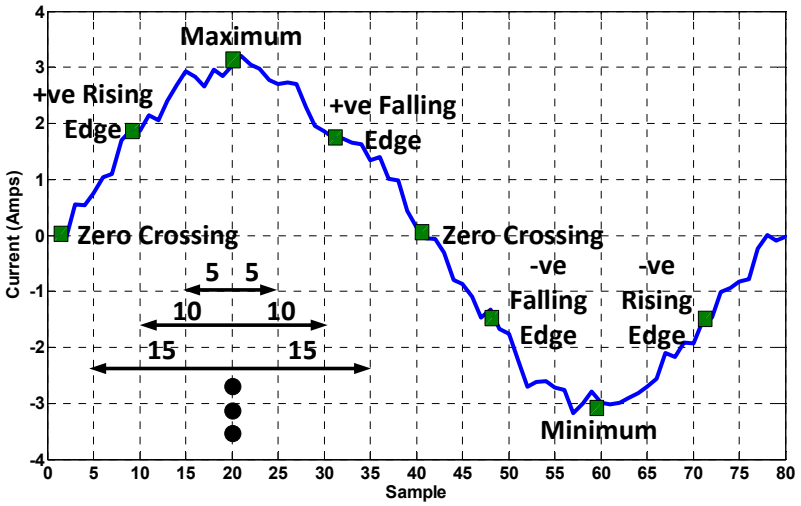
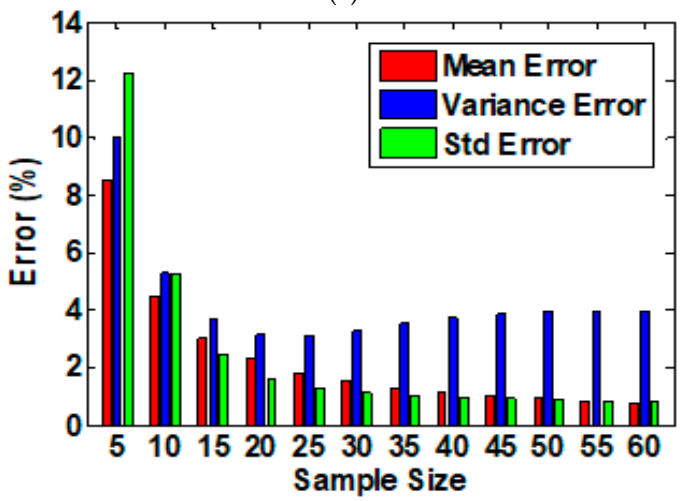


Figure 15. The bilayer detection algorithm.

Next, five samples were taken before and after the peak sample and the mean, variance, and standard deviation were calculated. The same were also calculated to the same sample from the original data (i.e., with the actual value instead of the fake sample). The error of both statistical vector indicators was then calculated for all the fake data cases. Next, the dataset size was increased to 10 before and 10 after the fake sample and the error was calculated. The same procedure was then repeated reaching a dataset size of 60 samples before and 60 samples after the fake sample. The same procedure was repeated at all the critical locations in the sinusoid: the minimum, negative/positive rising/falling edge, and zero crossing. All the error data were then averaged for each data set size and are plotted in Figure 16b. As shown in Figure 16b the most indicative dataset (largest errors) was for 5 samples before the suspected sample and 5 samples after it.



(**a**)



(**b**)

Figure 16. (**a**) Performed fake data injection attack study; (**b**) selection of indicators.

6. Model Development and Verification

As mentioned earlier, the NN-F was trained using datasets for different operational states/contingencies of the microgrid collected from history recorded data or from simulation data. To ensure the fidelity of the simulation model, the simulated microgrid developed in Matlab/Simulink was verified by comparing bus voltages, currents, and power measurements with experimental data. Table 1 shows that the simulated microgrid accurately describes the real one.

Table 1. Simulated model verification.

	G1 Bus		G2 Bus		AC Load 1		AC Load 2		DC Microgrid	
	S	**E**	**S**	**E**	**S**	**E**	**S**	**E**	**S**	**E**
$\mathbf{I_{rms}}$	2.1	2.6	0.7	0.8	1.5	1.2	1.3	1.5	0.4	0.4
$\mathbf{V_{rms}}$	120	119	120	120	116	116	118	119	118	118
P (W)	761	758	243	255	431	435	441	442	170	170

Note: E: Experimental; S: Simulation.

In practice, each utility or system operators has their own tools to model their power systems for internal studies. Such models and tools can be used to generate training data.

7. Results and Discussion

To study and analyze the effectiveness of the proposed framework, we implemented a MU at Load 1 and an associated control agent that receives current measurements of Load 1 on the other side of the process bus. Several experiments were performed to assess the effectiveness of the different modules in this framework against spoofed measurement attacks and the results are reported in this section.

7.1. Experiment 1

The malware script was used to inject a small perturbation to the current value at sample 1200. In this small perturbation attack, the NN forecasting error started to accumulate. At this stage, the AI module monitored the rate of change of the error of the next few samples. As seen in Figure 17, the error derivative was increasing indicating that indeed an accumulation of the forecasting error between the received and forecasted samples was occurring. Therefore, the IED sent a flag to the merging unit including the position (i.e., sample 1200) of the suspected sample to activate layer 2 of the proposed framework. Both the control agent and its merging unit performed the statistical study explained earlier. The results of the statistical analyses showed a difference in the calculated mean, variance, and standard deviation. This means that the forecasting error accumulated due the perturbation attack.

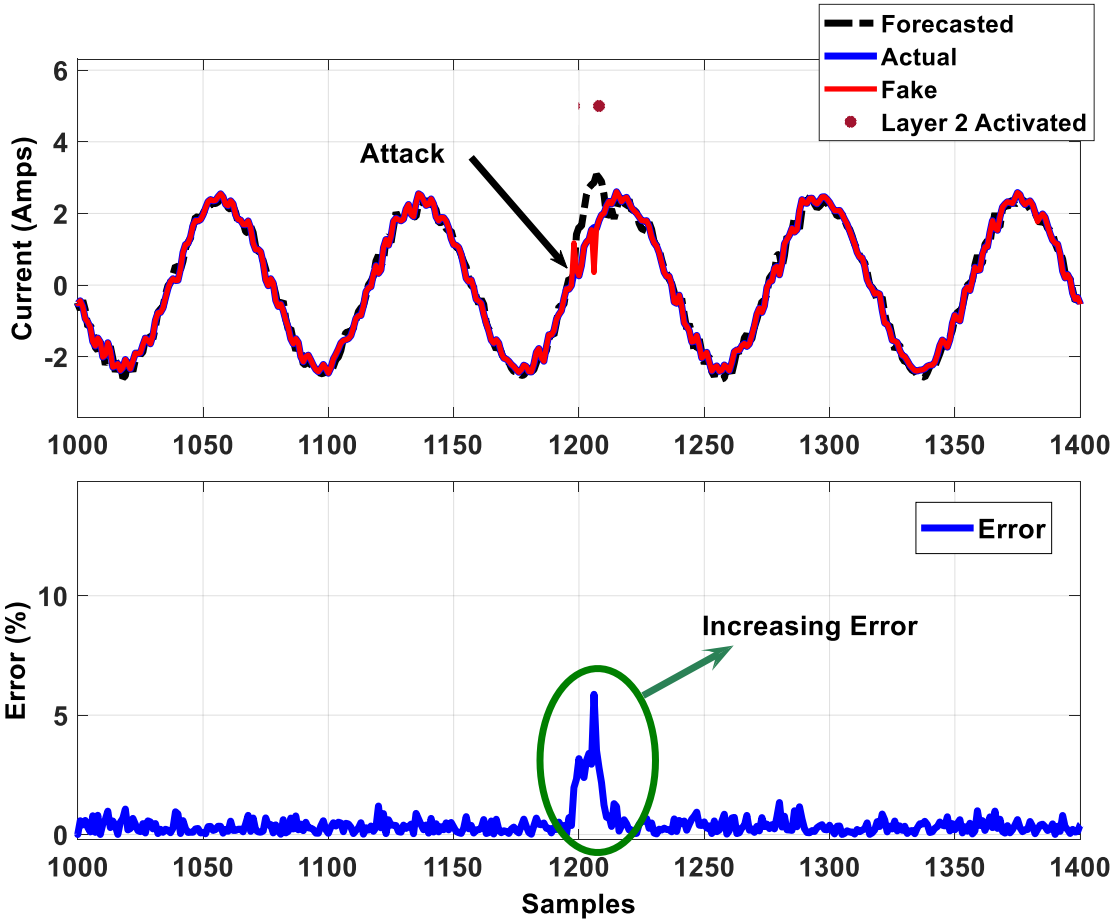


Figure 17. Detection of perturbation attack.

7.2. Experiment 2

The malware script was used to inject spoofed messages simulating a fake fault. As seen in Figure 18, the attacker injected current measurement values corresponding to a fake fault condition. Since the first spoofed sample does not violate the detection threshold, it is assumed that the IED uses it to forecast the next sample. The next forecasted sample will be closer to the fault condition, rather than the legitimate SMV. Therefore, the IED will forecast that there is a fault, and will act accordingly. However, given the proposed cyber and physical defense mechanism, when the IED noticed two SMV messages, one after the other, each with repeated sample counters, it performed the algorithm in Section V B. Based on comparing the statistical indicators it received from the MU and the ones it calculated, it was able to identify that there is an attack and that this was a fake fault condition.

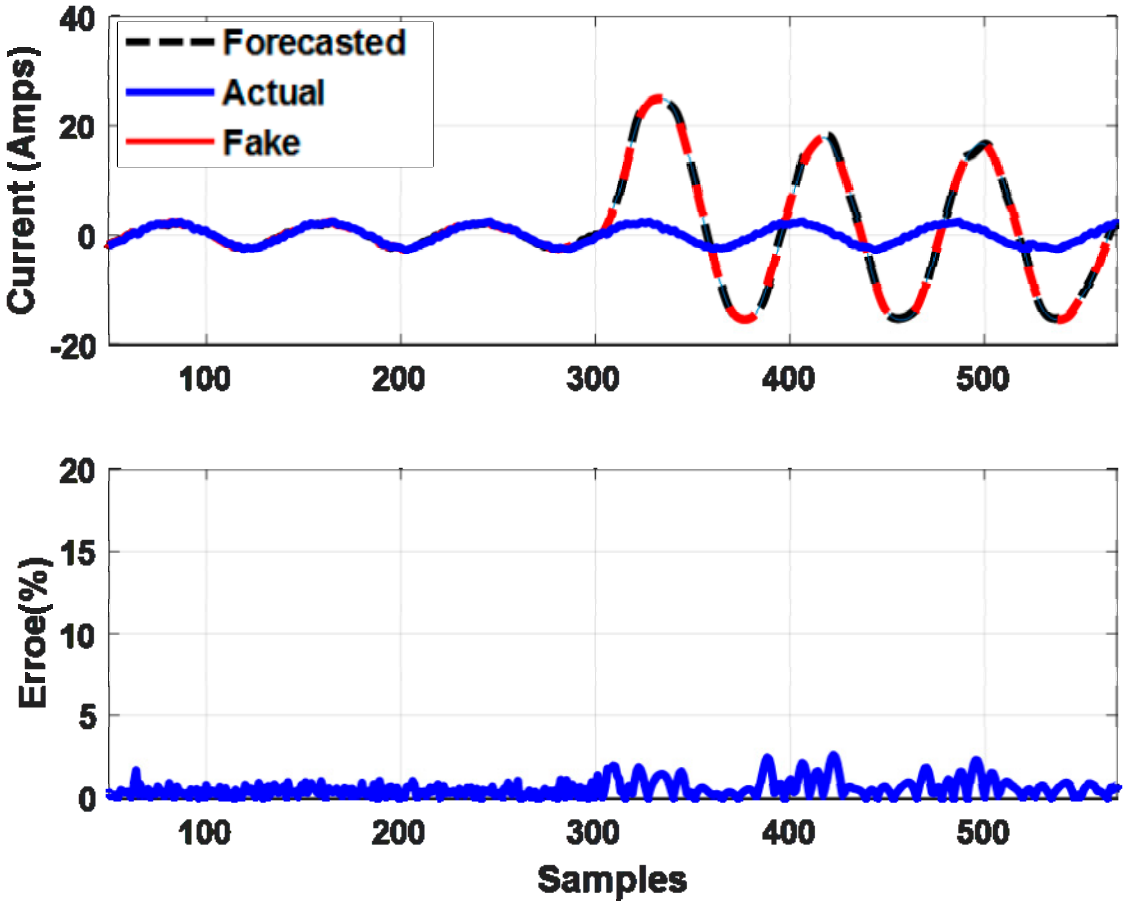


Figure 18. Detection of a fake fault.

7.3. Experiment 3

Finally, the latency of the complete detection process including the information exchange over the out-of-band network and the hardware time required for packet crafting was assessed.

For the hardware implementation, the authors used the open source libIEC61850 library to program two Odroid C2 microcontrollers. One was programmed as an IED that subscribes to SMV packets, whereas the other was programmed as a MU that publishes SMV packets. The bi-layer algorithm was implemented on those devices as shown in Figure 15 of the manuscript.

In order to calculate the latency of the proposed algorithm, the following experimental setup was prepared. A High logic value was put on the digital input of the Odroid device representing the IED. The device was programmed to send the flag to the MU once a High value is detected on its digital input. The device then calculated its own statistical indicators, waits until it received the output of

the MU, and takes a decision. Once the decision is made, the IED generates a logic High on one of its digital outputs. The digital input and the digital output of the IED were connected to the oscilloscope shown in Figure 19 in the manuscript.

In Figure 19, t_1 is the time starting from the instant of issuing the flag, the time to calculate the two statistical vectors in the control agent. t_2 is the time it takes the first packet, which contains the flag, to reach the MU. $t_3 + t_4$ is the time for calculating the statistics vector by the MU and the time for the packet, which contains the statistics, to reach the control agent, and finally issuing a decision. Therefore, the total detection latency is calculated in Equation (8):

$$t_T = t_1 + t_2 + t_3 + t_4 = 0.4 + 0.25 + 0.25 + 0.3 = 1.2\ ms \tag{8}$$

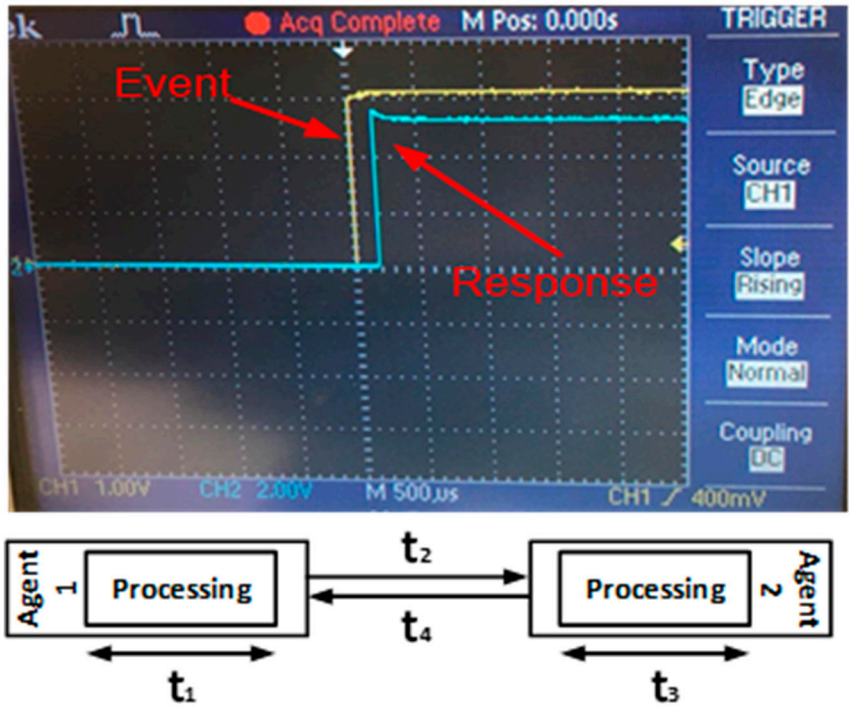


Figure 19. Detection latency.

8. Conclusions and Recommendations for Future Work

8.1. Conclusions

In this paper, an analysis of the SMV process bus was presented. An overview of the process bus and its benefits was first given, along with the detailed structure of the SMV datagram. Then, a discussion of the vulnerabilities of the SMV protocol was presented, and the current security countermeasures were outlined, such as those stipulated in IEC 62351. After that, the reliability of utilizing neural network forecasters to detect spoofed messages was studied. The results of this study showed that although they have a high detection accuracy, they could still be prone to the accumulation of the forecasting error. Accordingly, this paper also presented a lightweight algorithm, based on statistical indicators, to enhance the reliability of the neural network forecaster in terms of detecting spoofed SMV packets. This algorithm is considered as a second line of defense, complementing the cyber security methods, such as encryption and authentication. The studies were conducted on a laboratory-scale hardware microgrid with a commercial network switch to be more representative of the microgrid environment than event-based simulation models.

8.2. Recommendations for Future Work

Finally, there are two layers on which the proposed algorithm can be implemented. One is the network layer, and the other is the host layer. Network-based systems monitor the entire traffic flowing in and out of a network, and they are most likely to be placed at the data entry and exit points, such as routers and switches. Host-based systems, on the other hand, monitor data coming to and going from a certain device in a network. Therefore, they are implemented on the device itself.

Each of the network-based and host-based systems has its advantages and disadvantages. The main focus of this manuscript is to conduct a feasibility study of using neural network forecasters to detect spoofed SMV packets. We chose to experimentally verify our proposed bi-layer algorithm on an IED and a MU as a proof of concept. However, the actual implementation in the field (i.e., real systems) could be improved. For example, on the network level, we could monitor the cyber-side information (ID and counters), and if an anomaly is detected (repeated counters for instance) then the specific IED could be triggered. Thus, each IED will be responsible for verifying the integrity of the SMV packets coming into it.

9. Patents

The U.S. Patent US10362056B1 (23 July 2019) entitled "Content-aware spoofed sensor measurement data detection in microgrids" is a result of part of the work reported in this manuscript.

Author Contributions: M.E.H., E.H., and T.Y. conceived and designed the algorithms and experiments, performed the experiments, and analyzed the data. M.S. contributed in writing and thoroughly editing the manuscript, organizing the flow of ideas, presenting the ideas in pictorial diagrams, as well as reviewing and analyzing the data. H.H. helped build the simulation model of the microgrid. O.M. is the main supervisor of the project.

Funding: This research was funded by the Department of Energy under Award DE-OE0000779.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cintuglu, M.H.; Youssef, T.; Mohammed, O.A. Development and Application of a Real-Time Testbed for Multiagent System Interoperability: A Case Study on Hierarchical Microgrid Control. *IEEE Trans. Smart Grid* **2016**, *9*, 1759–1768. [CrossRef]
2. el Hariri, M.; Youssef, T.A.; Mohammed, O.A. On the Implementation of the IEC 61850 Standard: Will Different Manufacturer Devices Behave Similarly under Identical Conditions? *Electronics* **2016**, *5*, 85. [CrossRef]
3. Hariri, M.E.; Youssef, T.; Harmon, E.; Habib, H.; Mohammed, O. The IEC 61850 Sampled Measured Values Protocol: Analysis, Threat Identification, and Feasibility of Using NN Forecasters to Detect of Spoofed Packets. In Proceedings of the 2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe), Genova, Italy, 10–14 June 2019; pp. 1–6.
4. Borges, C.E.; Penya, Y.K.; Fernández, I. Evaluating combined load forecasting in large power systems and smart grids. *IEEE Trans. Ind. Inform.* **2012**, *9*, 1570–1577. [CrossRef]
5. Bessa, R.J.; Trindade, A.; Miranda, V. Spatial-Temporal Solar Power Forecasting for Smart Grids. *IEEE Trans. Ind. Inf.* **2014**, *11*, 232–241. [CrossRef]
6. Evangelos, F.; Brett, A.; Reza, A.; Daniel, B.; Pavel, E.; Murph, S. *Data Mining Techniques and Tools for Synchrophasor Data*; NASPI WHITE PAPER; Engineering Analysis Task Team: San Diego, CA, USA, January 2019. [CrossRef]
7. Ozay, M.; Esnaola, I.; Vural, F.T.Y.; Kulkarni, S.R.; Poor, H.V. Machine Learning Methods for Attack Detection in the Smart Grid. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 1773–1786. [CrossRef] [PubMed]
8. Wu, Y.; Xiao, Y.; Hohn, F.; Nordstrom, L.; Wang, J.; Zhao, W. Bad Data Detection Using Linear WLS and Sampled Values in Digital Substations. *IEEE Trans. Power Deliv.* **2017**, *33*, 150–157. [CrossRef]
9. Vimalkumar, K.; Radhika, N. A big data framework for intrusion detection in smart grids using apache spark. In Proceedings of the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 13–16 September 2017; pp. 198–204.

10. Faisal, M.A.; Aung, Z.; Williams, J.R.; Sanchez, A. Data-Stream-Based Intrusion Detection System for Advanced Metering Infrastructure in Smart Grid: A Feasibility Study. *IEEE Syst. J.* **2015**, *9*, 31–44. [CrossRef]
11. Hong, J.; Liu, C.; Govindarasu, M. Integrated Anomaly Detection for Cyber Security of the Substations. *IEEE Trans. Smart Grid* **2014**, *5*, 1643–1653. [CrossRef]
12. Pan, S.; Morris, T.; Adhikari, U. Developing a Hybrid Intrusion Detection System Using Data Mining for Power Systems. *IEEE Trans. Smart Grid* **2015**, *6*, 3104–3113. [CrossRef]
13. Leu, F.; Tsai, K.; Hsiao, Y.; Yang, C. An Internal Intrusion Detection and Protection System by Using Data Mining and Forensic Techniques. *IEEE Syst. J.* **2017**, *11*, 427–438. [CrossRef]
14. Fadlullah, Z.M.; Fouda, M.M.; Kato, N.; Shen, X.; Nozaki, Y. An early warning system against malicious activities for smart grid communications. *IEEE Netw.* **2011**, *25*, 50–55. [CrossRef]
15. Koutsandria, G.; Muthukumar, V.; Parvania, M.; Peisert, S.; McParland, C.; Scaglione, A. A hybrid network IDS for protective digital relays in the power transmission grid. In Proceedings of the 2014 IEEE International Conference on Smart Grid Communications (SmartGridComm), Venice, Italy, 3–6 November 2014; pp. 908–913.
16. Ten, C.W.; Hong, J.; Liu, C.C. Anomaly Detection for Cybersecurity of the Substations. *IEEE Trans. Smart Grid* **2011**, *2*, 865–873. [CrossRef]
17. Sridhar, S.; Govindarasu, M. Model-Based Attack Detection and Mitigation for Automatic Generation Control. *IEEE Trans. Smart Grid* **2014**, *5*, 580–591. [CrossRef]
18. Zhao, J.; Zhang, G.; la Scala, M.; Dong, Z.Y.; Chen, C.; Wang, J. Short-Term State Forecasting-Aided Method for Detection of Smart Grid General False Data Injection Attacks. *IEEE Trans. Smart Grid* **2015**, *8*, 1580–1590. [CrossRef]
19. Ali, M.Q.; Yousefian, R.; Al-Shaer, E.; Kamalasadan, S.; Zhu, Q. Two-tier data-driven intrusion detection for automatic generation control in smart grid. In Proceedings of the 2014 IEEE Conference on Communications and Network Security, San Francisco, CA, USA, 29–31 October 2014; pp. 292–300.
20. Mashima, D.; Chen, B.; Zhou, T.; Rajendran, R.; Sikdar, B. Securing Substations through Command Authentication Using On-the-fly Simulation of Power System Dynamics. In Proceedings of the 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Aalborg, Denmark, 29–31 October 2018; pp. 1–7.
21. Meliopoulos, S.; Cokkinides, G.; Fan, R.; Sun, L.; Cui, B. Command authentication via faster than real time simulation. In Proceedings of the 2016 IEEE Power and Energy Society General Meeting (PESGM), Boston, MA, USA, 19 July 2016; pp. 1–5.
22. Yang, Y.; McLaughlin, K.; Sezer, S.; Littler, T.; Im, E.G.; Pranggono, B.; Wang, H.F. Multiattribute SCADA-Specific Intrusion Detection System for Power Networks. *IEEE Trans. Power Deliv.* **2014**, *29*, 1092–1102. [CrossRef]
23. Kaliski, B.S., Jr. A Layman's Guide to a Subset of ASN.1, BER, and DER, AN RSA Laboratories Technical Note. Available online: http://luca.ntop.org/Teaching/Appunti/asn1.html (accessed on 19 August 2019).
24. International Electrotechnical Commission. *Security for IEC 61850 Profiles*; IEC 62351-6; International Electrotechnical Commission: Geneva, Switzerland.
25. Kanabar, M.G.; Sidhu, T.S. Performance of IEC 61850-9-2 Process Bus and Corrective Measure for Digital Relaying. *IEEE Trans. Power Deliv.* **2010**, *26*, 725–735. [CrossRef]
26. Ingram, D.M.E.; Schaub, P.; Taylor, R.R.; Campbell, D.A. Performacnce Analysis of IEC 61850 Sampled value Process Bus. *IEEE Trans Ind. Inform.* **2012**, *9*, 1445–1454. [CrossRef]
27. *IEC 61850-9-2 International Standard Communication Networks and Systems in Substations*; International Electrotechnical Commission: Geneva, Switzerland, 2004.
28. Hoyos, J.; Dehus, M.; Brown, T.X. Exploiting the goose protocol: A practical attack on cyber-infrastructure. In Proceedings of the 2012 IEEE Globecom Workshops (GC Wkshps), Anaheim, CA, USA, 3–7 December 2012; pp. 1508–1513.
29. Lin, H.; Slagell, A.; Kalbarczyk, Z.; Sauer, P.; Iyer, R. Runtime Semantic Security Analysis to Detect and Mitigate Control-related Attacks in Power Grids. *IEEE Trans. Smart Grid* **2016**, *9*, 163–178. [CrossRef]
30. Falliere, N.; Murchu, L.O.; Chien, E. W32.Stuxnet Dossier, Symantic Security Response Report. Version 1.4. Available online: https://dragos.com/wp-content/uploads/CrashOverride-01.pdf (accessed on 15 September 2019).
31. DRAGOS INC. Report Version 2.20170613 Crashoverride: Analyses of the Threat to Electric Grid Operation. Available online: https://dragos.com/wp-content/uploads/CrashOverride-01.pdf (accessed on 15 September 2019).

32. Packet Sniffing Attack Prevention. Available online: https://cconell2858.wordpress.com/packet-sniffing-attack-prevention/ (accessed on 15 September 2019).
33. Youssef, T.A.; Hariri, M.E.; Bugay, N.; Mohammed, O.A. IEC 61850: Technology standards and cyber-threats. In Proceedings of the 2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), Florence, Italy, 7–10 June 2016; pp. 1–6.
34. Fuloria, S.; Anderson, R.; Mcgrath, K.; Hansen, K.; Alvarez, F. The Protection of Substation Communications. In Proceedings of the SCADA Security Scientific Symposium, Miami, FL, USA, 18–19 January 2010.
35. Obermeier, S.; Schlegel, R.; Obermeier, S.; Schneider, J. Assessing the Security of IEC 6235. In Proceedings of the 3rd International Symposium for ICS & SCADA Cyber Security Research 2015, Ingolstadt, Germany, 17–18 September 2015.
36. Pubudu Weerathunga, Security Aspects of Smart Grid Communication. Master's Thesis, The School of Graduate and Postdoctoral Studies Western University London, London, UK, 2012.
37. Intel DPDK Validation team, DPDK Intel Cryptodev Performance Report. Available online: https://fast.dpdk.org/doc/perf/DPDK_18_02_Intel_crypto_performance_report.pdf (accessed on 15 September 2019).
38. Peyrin, T.; Sasaki, Y.; Wang, L.; Wang, X.; Sako, K. Generic Related-Key Attacks for HMAC. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, 2–6 December 2012; pp. 580–597.
39. Jayarin, P.J.; Visumathi, J.; Srilakshmi, R.; Pendyala, M. A Secured Key Distribution for Effective File Transfer Using HMAC-SHA Algorithm with Self-Healing Property. *J. Appl. Secur. Res.* **2015**, *10*, 221–237. [CrossRef]
40. Fouque, P.-A.; Leurent, G.; Nguyen, P.Q. Full Key-Recovery Attacks on HMAC/NMAC-MD4 and NMAC-MD5. In Proceedings of the 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, 19–23 August 2007.
41. Shmueli, G.; Lichtendahl, K.C. *Practical Time Series Forecasting with R: A Hands-On Guide*; Axelrod Schnall Publishers: Berlin/Heidelberg, Germany, 2016.
42. Lapedes, A.; Farber, R. *Nonlinear Signal Processing Using Neural Networks: Prediction and System Modeling*; Los Alamos National Laboratory Technical Report: LA-UR-87-2662; IEEE: Piscataway, NJ, USA, 1987.
43. Zhang, G.P. *Neural Networks for Time-Series Forecasting from the Handbook of Natural Computing*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 461–477.