12-5-2016

On the Implementation of the IEC 61850 Standard: Will Different Manufacturer Devices Behave Similarly under Identical Conditions?

Mohamad El Hariri
Department of Electrical and Computer Engineering, *Florida International University*, melha003@fiu.edu

Tarek A. Youssef
Department of Electrical and Computer Engineering, *Florida International University*, tyous001@fiu.edu

Osama A. Mohammed
Department of Electrical and Computer Engineering, *Florida International University*, mohammed@fiu.edu

electronics

MDPI

Article

On the Implementation of the IEC 61850 Standard: Will Different Manufacturer Devices Behave Similarly under Identical Conditions?

Mohamad El Hariri, Tarek A. Youssef and Osama A. Mohammed *

Energy Systems Research Laboratory, Department of Electrical and Computer Engineering, Florida International University, Miami, FL 33174, USA; melha003@fiu.edu (M.E.H.); tyous001@fiu.edu (T.A.Y.)
* Correspondence: mohammed@fiu.edu; Tel.: +1-305-348-3040; Fax: +1-305-348-3707

Academic Editors: Alfredo Vaccaro and Jin (Wei) Kocsis
Received: 3 November 2016; Accepted: 25 November 2016; Published: 5 December 2016

Abstract: Standardization in smart grid communications is necessary to facilitate complex operations of modern power system functions. However, the strong coupling between the cyber and physical domains of the contemporary grid exposes the system to vulnerabilities and thus places more burden on standards' developers. As such, standards need to be continuously assessed for reliability and are expected to be implemented properly on field devices. However, the actual implementation of common standards varies between vendors, which may lead to different behaviors of the devices even if present under similar conditions. The work in this paper tested the implementation of the International Electro-technical Commission's Generic Object Oriented Substation Event GOOSE (IEC 61850 GOOSE) messaging protocol on commercial Intelligent Electronic Devices (IEDs) and the open source libiec61850 library—also used in commercial devices—which showed different behaviors in identical situations. Based on the test results and analysis of some features of the IEC 61850 GOOSE protocol itself, this paper proposes guidelines and recommendations for proper implementation of the standard functionalities.

Keywords: cyber-attack; GOOSE; IEC 61850; IED; standardization; Substation Automation

1. Introduction

Communication protocols are the basis for determining how a cyber-physical system gathers information and sends it as control signals. Therefore, an accurate definition of communication protocols is of paramount importance in defining the architecture of control systems [1]. However, intricate interdependencies between the cyber and physical components of a cyber-physical system increase the difficulty of devising communication protocols that ensure proper information flow in such systems, and thus complicates the design process of control algorithms. The challenge lies in the fact that in a highly interconnected cyber-physical system, a slight exploit in the cyber domain can have a significant impact in the physical domain and vice-versa [2]. In current days, the operation of commercial, industrial, medical, military, and many critical infrastructures relies on the cyber-physical smart grid. The reliance of such critical infrastructure on the smart grid means reliance on the grid's cyber domain, physical domain, and most importantly, the interactions between them [2]. Therefore, understanding and modelling data exchange in the smart grid is a noticeably challenging process with considerable effort placed on accurately capturing the interactions between both the cyber component and the physical component of the grid.

In order to solve the information flow modelling problem and facilitate the design of cyber-physical smart grid applications, various data communication standards were developed for different parts of the smart grid. One of the vital standards in the electrical automation systems

around which many automation projects have been built is IEC 61850 [3–5]. An essential part in power generation and distribution processes, this work focuses on data modelling, specifically the IEC 61850 communication standard in Substation Automation Systems (SAS). Communication between substation devices, namely Intelligent Electronic Devices (IEDs), is integral for substations to keep up with their real-time operations. IEDs are embedded microcontroller systems that support Ethernet-based communication and perform several protective and control functions in an SAS, such as data and file transfer [3,6]. In order to ensure interoperability between IEDs, the IEC 61850 standard was developed by the International Electro-technical Commission (IEC) Technical Committee Number 57 Working Group 10 (TC57 WG10) and IEEE for Ethernet (IEEE 802.3)-based communication in electrical substations [7]. The IEC 61850 provides a comprehensive data modelling and organization method that unifies data structure definitions across all IED brands. The standard abstracts the definition of the service and data items to be independent from the underlying protocols. The abstracted data items and services can thus be mapped into any other communication protocol. IEC 61850 maps the data to three different protocols based on the application: the Manufacturing Message Specification (MMS) protocol is used for control and automation functions whereas the Generic Object Oriented Substation Event (GOOSE) and Sampled Measured Values (SMV) protocols are used for real-time operations [8]. Recently, the IEC 61850 has been extended (IEC 61850-90-1) to cover applications that require inter-substation communication such as teleprotection, which requires the use of GOOSE messaging protocol over wide area networks (WAN) for fast propagation of control signals [9–11]. A close look at the current engineering literature also shows a trend of utilizing IEC 61850 GOOSE messages in some microgrid control applications [12,13].

Standards are developed to be implemented. Nonetheless, due to different device topologies and hardware used, not all vendors follow the same implementation process. Here, concerns arise about the degree of compliance of devices from different vendors with the standard being implemented. Due to its criticality, failing to implement the IEC 61850 standard properly on field devices may expose the overall system they operate in to unwanted vulnerabilities. In fact, the criticality of IEC 61850-based communications in terms of data transfer, reliability, availability and efficiency has been the concern of several research works [14,15]. In this work, a case study of the implementation of IEC 61850 GOOSE messaging on commercial IEDs present at the Smart Grid test bed at Florida International University and the open source libiec61850 [16] library, which is also implemented on commercial devices, was performed. GOOSE messaging protocol in particular is of paramount importance due to its application in transporting time-critical power system protection commands. Several experiments were conducted to test critical features of the standard, which are detailed later in this paper. The results showed that different devices produce different responses under similar conditions. This paved the way to launch a successful data manipulation cyber-attack on the devices under study.

The rest of the paper is organized as follows: Section 2 details the anatomy of a GOOSE message and describes the algorithm for processing GOOSE messages. The actual performed experiments are presented in Section 3 along with details about the performed cyber-attack. Results of the conducted experiments are presented in Section 4, which also proposes guidelines for proper implementation of the GOOSE protocol. Finally, Section 5 is a conclusion of the work presented.

2. IEC 61850 GOOSE Messaging Protocol

GOOSE messaging is a fast, non-routable, and reliable data exchange protocol between IEDs defined in IEC 61850-8-1, which is the basis of critical power system functions such as power line protection. GOOSE messages are Ethernet messages sent over layer 2 of the Open System Interconnect (OSI) model (IEEE 802.3) following a publish/subscribe model, unlike MMS messages, which are routable and sent over layer 3 of the OSI model. That is, the publishing IED creates a multicast message to which a number of destination IEDs subscribe concurrently. In order to ensure delivery of the message, at every substation event, the publishing IED repeatedly transmits the same GOOSE message with an increasing transmission period until a maximum predefined period is reached [17].

2.1. The Anatomy of a GOOSE Message

As shown in Figure 1, the GOOSE datagram follows a modified Abstract Syntax Notation One (ASN.1) Basic Encoding Rules (BER) Tag/Length pair encoding scheme [17]. The Tag field represents the type of information, which is represented in the following GOOSE frame. Each of the fields has a unique tag value specified by the standard. As shown in Figure 2, the tag for the GOOSE Protocol Data Unit (goosePDU) field is 81, whereas the tags for the stNum and sqNum fields are 85 and 86, respectively. The Length field represents the number of bytes in the following GOOSE frame. For example, the sqNum in Figure 2 has a Length field of 03, which means that the following three hex pairs (00-c9-06) are the sqNum itself. The sqNum here is in hexadecimal.

Destination MAC Address		Source MAC Address		Priority Tagging/VLAN ID	
Ethertype (88B8)		APPID		Length	
Reserved 1		Reserved 2	Tag	Length	goosePDU
Tag	Length	gocbRef	Tag	Length	timeAllowedtoLive
Tag	Length	datSet	Tag	Length	goID
Tag	Length	t	Tag	Length	stNum
Tag	Length	sqNum	Tag	Length	test
Tag	Length	confRev	Tag	Length	ndsCom
Tag	Length	numDatSetEntries	Tag	Length	allData
Tag	Length	Data 1 (Boolean)	Tag	Length	Data 2 (Float)
●●●●●●		●●●●●●	Tag	Length	Data N

Figure 1. Structure of a Generic Object Oriented Substation Event (GOOSE) Datagram, MAC: media access control, APPID: application ID, VLAN ID: virtual local area network ID, goosePDU: GOOSE protocol data unit, gocbRef: GOOSE control block reference, datSet: data set, goID: GOOSE ID, t: time, stNum: status number, sqNum: sequence number, confRev: configuration revision, ndsCom: needs commissioning, numDatSetEntries: number of data set entries.

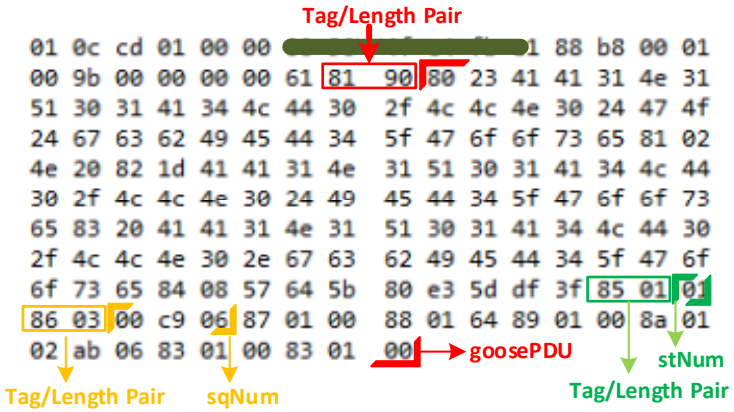

Figure 2. Hexadecimal Representation of a GOOSE Datagram.

The GOOSE datagram starts with the Destination Media Access Control (MAC) Address, which is a multicast address reserved for IEC 61850 applications always starting with 01-0c-cd, and is followed by a six-octet source MAC address. This is the MAC address of the publishing IED. A GOOSE message has an IEEE 802.1Q Virtual Local Area Network ID (VLAN ID) and a unique Ethernet type (88-b8).

The Application ID (APPID) field is a four-octet field, which the subscribing IEDs use to identify messages to which they are subscribing. The Length field represents the length of the overall GOOSE datagram minus eight bytes and is followed by two reserved fields left out by the standard for future use. The goosePDU field itself is composed of twelve subfields that follow the modified ASN.1 BER encoding scheme as well [17]. The goosePDU consists of the following:

- gocbRef: GOOSE control block reference
- timeAllowedtoLive: the time a receiver waits before receiving a re-transmitted message
- datSet: name of the Data Set
- goID: ID of publishing IED
- t: time stamp indicating a new GOOSE event
- stNum: counter that increments with every GOOSE event
- sqNum: counter that increments with every repeated GOOSE message
- test: specifies if a message is/is not intended for testing
- confRev: number of times Data Set has changed
- ndsCom: needs commissioning field
- numDataEntries: number of data elements in allData
- allData: actual data being sent (bool, integer, float,...)

2.2. IEC Standard Guidelines for Processing GOOSE Messages

IEC 61850-8-1 defines the structure of a GOOSE message and the means by which it is communicated over a network. Despite its criticality, IEC 61850 advanced in an era where substations operated in isolated proprietary networks and thus did not include any cyber security measures for data communication. However, this will no longer be the case as operators are moving towards open networks and remote access of substation control systems through the aid of contemporary communication technologies such as cloud services. For instance, authors in [18] investigated the use of IEC 61850 for teleprotection outside the boundaries of a single substation over wide area networks (WAN). Also, in an effort called Cloud IEC 61850, authors in [19] investigated the idea of having virtualization and cloud technologies as the underlying infrastructure of electrical automation systems with a specific example of a substation automatic voltage control. Recent literature also shows the application of IEC 61850 in hierarchical microgrid control, where the authors in [20] propose a comprehensive hybrid agent framework combining the foundation for intelligent physical agents (FIPA), IEC 61850, and data distribution service (DDS) standards. With the realization of this modern communication infrastructure, IEC 62351 emerged in order to tackle the shortcomings of IEC 61850 in terms of communication security. IEC 62351 was developed by IEC TC57 WG15 and consists of eleven parts to cover end-to-end security issues in power systems communications [1]. IEC 62351-6 covers communication security within the boundaries of a substation covering MMS, GOOSE and SMV protocols.

IEC 62351-6 devises an algorithm for proper processing of GOOSE messages in order to mitigate some cyber-attacks such as replay and man-in-the-middle. From the publishing IED side, each GOOSE message has a status and a sequence number field (stNum and sqNum respectively). When a substation event occurs, for example, an overcurrent is sensed, the publishing IED instantly transmits a message with an incremented stNum field. The message is then repeated with variable increasing time delay until the maximum defined period is reached. The sqNum counter increments with every repeated message until the maximum number count ($2^{32}-1$) is reached; the point at which the sqNum counter rolls over. IEC 62351-6 states that a subscriber IED that detects a new message with a new stNum must discard any message having an stNum less than or equal to the previous message and which time allowed to live has not timed-out yet, unless a rollover of the stNum counter occurs. If none of the above conditions are true, the subscribing IEDs process the messages. A flowchart describing the algorithm for processing GOOSE messages set by IEC 62351-6 is presented in Figure 3.

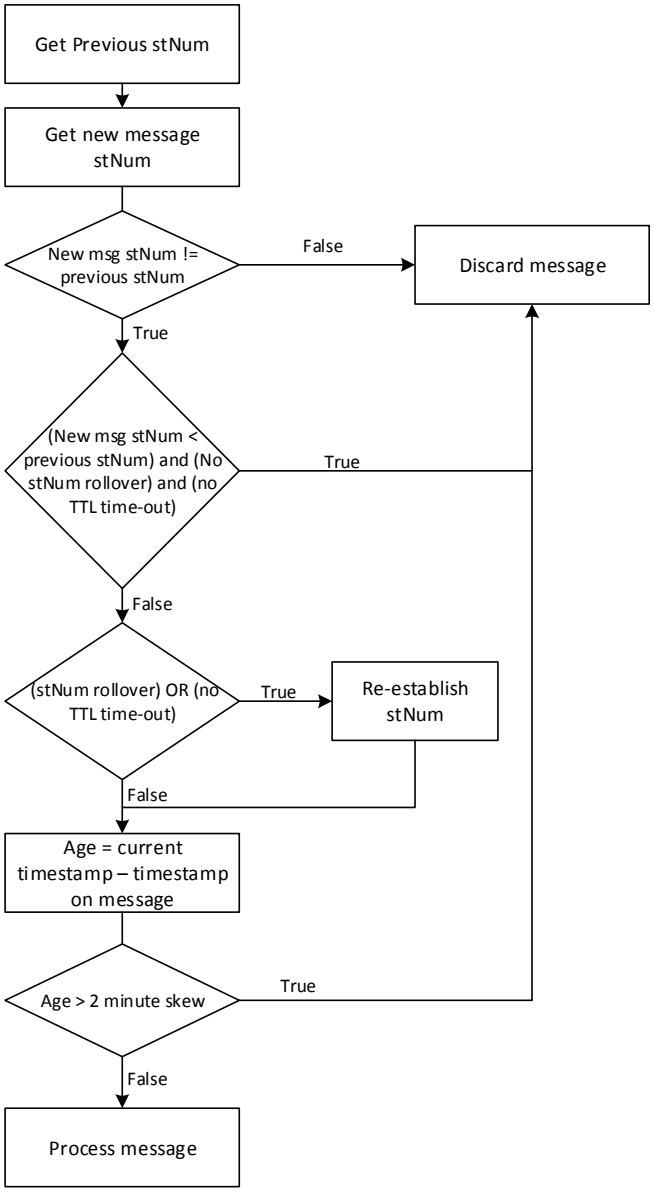

Figure 3. GOOSE Messages Processing Algorithm.

3. Testing of Commercial IEDs Communicating with IEC 61850 GOOSE Messaging Protocol

A case study of the implementation of IEC 61850 GOOSE messaging on commercial IEDs present at the Smart Grid test bed at Florida International University and the open source libiec61850 [16] library, which is also implemented on commercial devices, was performed in this paper.

Figure 4 shows the experimental setup with the commercial IEDs having the vendor's proprietary implementation of IEC 61850. Manufacturer details of the commercial IEDs under test are intentionally omitted. Under normal conditions, the publishing IED is programmed to broadcast a GOOSE message with two Boolean data fields set to False (00-00). The subscribing IEDs read the Boolean data and control the status of the circuit breaker accordingly. In this case, the data read (False) maintains the relay's un-tripped status and the circuit breaker's closed status.

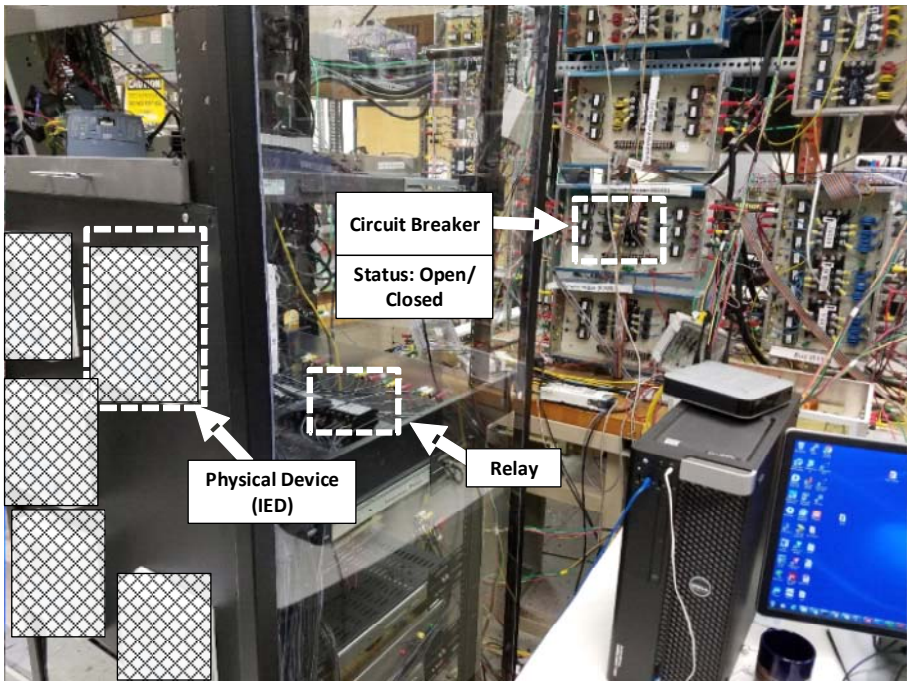

Figure 4. Experimental Setup with Commercial Intelligent Electronic Devices (IEDs).

Similarly, Figure 5 shows the experimental setup in which the libiec61850 GOOSE open source library has been implemented on two embedded boards. The publishing IED has the *goose_publisher* routine implemented on it, whereas the receiving IED has the *goose_subscriber* routine. More details about the implemented routines can be found on the open source library's website [16]. It is worth mentioning that this library has also been implemented on other commercial devices. The device on the left is the publishing IED, whereas the device on the right is the subscribing IED. The publishing IED also transmits Boolean data (either True or False), which the subscribing IED reads and triggers a digital output accordingly as marked in red in Figure 5. The subscribing boards were designed with connection capabilities to the solid-state circuit breakers shown in Figure 4.

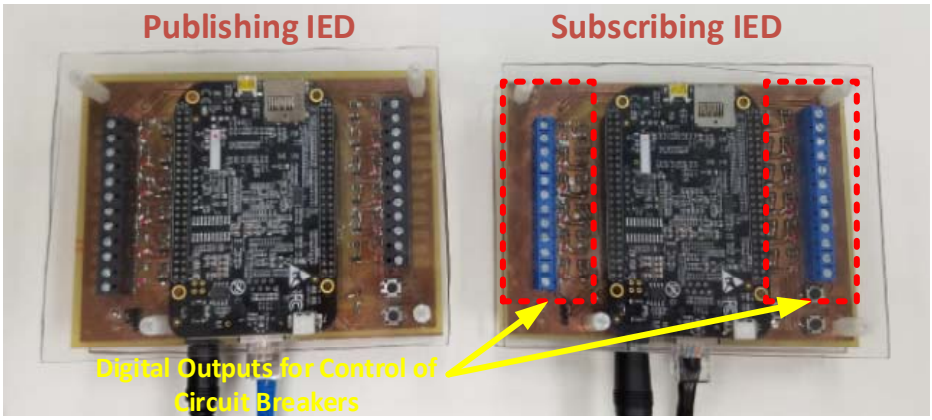

Figure 5. Experimental Setup with libiec61850 Implemented on Embedded Boards.

In order to perform the tests, a Python script was written in conjunction with network traffic capturing and packet crafting libraries from Scapy [21,22]. The developed script takes advantage of the simplicity of the unencrypted GOOSE message structure defined in Section 2 in order to monitor the Local Area Network (LAN) and capture Ethertype (88-b8) GOOSE messages. Each field in the

captured GOOSE messages was properly decoded based on the IEC 61850-8-1 modified ASN.1 BER mechanism. The script then modifies the content of the messages, encodes all the fields, crafts the new fake packet, and broadcasts it over the LAN. For each test, a certain field in the GOOSE message was modified and the results of the various tests are discussed below. A general overview of the data manipulation procedure performed is shown in Figure 6. Figure 7 shows a screen shot of the captured messages by the developed script and the corresponding decoded fields. The script was run on a Virtual Machine operating with Ubuntu Version 16.04.

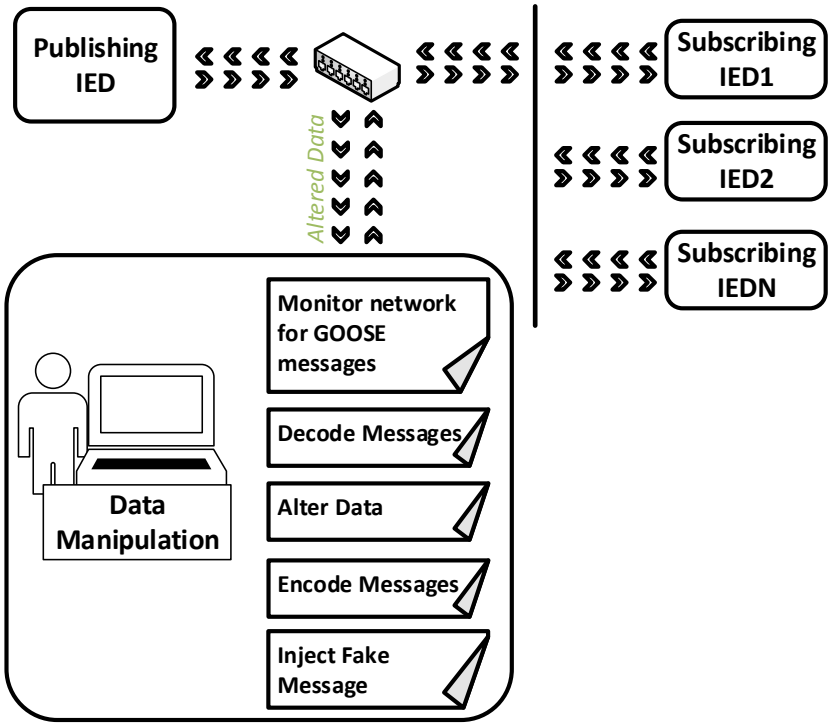

Figure 6. GOOSE Messages Data Manipulation Overview.

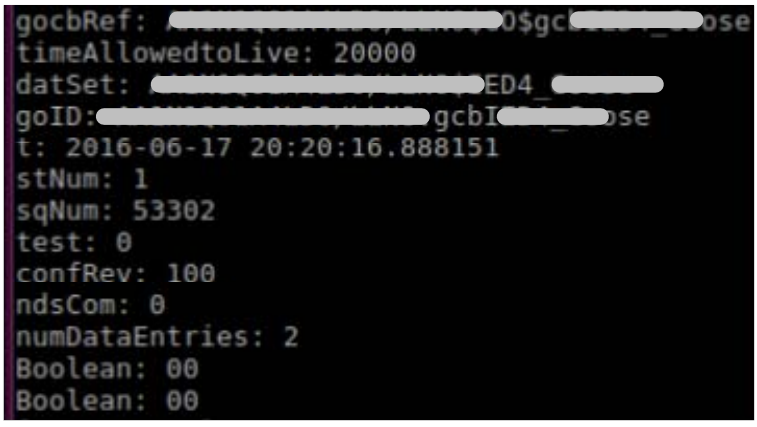

Figure 7. Captured Packet (manufacturer's details are intentionally omitted).

In the experimental setup for both the commercial IEDs and the developed embedded devices running the open source libiec61850 library, the publishing IED transmits messages with a status number stNum = 1, False Boolean Data fields (00-00), and incrementing sequence numbers.

3.1. Processing of Status Number

As explained in the flowchart of Figure 3, any GOOSE message with a status number different than that of its predecessor shall be discarded if it has an stNum equal to or less than that of the previous message and is still within its valid time allowed to live.

3.1.1. Commercial IEDs

In this test, first a message with stNum = 2 (>1) and True (01-01) Boolean data fields was sent. As anticipated from the standard, this message was processed and the circuit breakers status changes from closed to open. Next, a message with stNum = 3 (>2) and False Boolean fields (00-00) was transmitted and was also processed. Finally, another fake message with stNum = 2 (<3) with True Boolean fields (01-01) was broadcasted. Although this final message had a lower stNum than its predecessor, it was processed and the status of the circuit breakers changed from closed to open. The GOOSE datagrams of the broadcasted messages are shown in Figure 8. It should be noted here that all messages had the same time stamp, which was three days old. When compared with the subscribing IED time stamp, it was noticed that the 2-min time skew mentioned in Figure 3 was exceeded; however, the messages were still processed.

Figure 8. Wireshark Capture of Transmitted Messages (manufacturer's details are intentionally omitted), red: original message with low status number, green: fake message with high status number, blue: original message retransmission with low status number.

3.1.2. Libiec61850

The same test was repeated on the open source libiec61850 library implemented on the two developed embedded devices. Here, the final message with a low stNum (2 < 3) was not processed. This is because libiec61850 has an *IsValid()* function which checks if the TimeAllowedToLive timeout is not elapsed and if GOOSE messages were received with correct state and sequence IDs [16].

3.2. Message Time Stamp

Each GOOSE message has a time stamp field which is updated with each increment of the status number (i.e. with each substation event). Therefore, subscribing IEDs receiving a new message with changed data fields and an incremented stNum field must expect to have a message with an updated time stamp.

3.2.1. Commercial IEDs

In this test, we sent a fake GOOSE message with an incremented status number (stNum = 2) and altered data True (01-01), but with an old time stamp (three days old). As shown in Figure 9, the device processed the message and the status of the circuit breaker changed from closed to open.

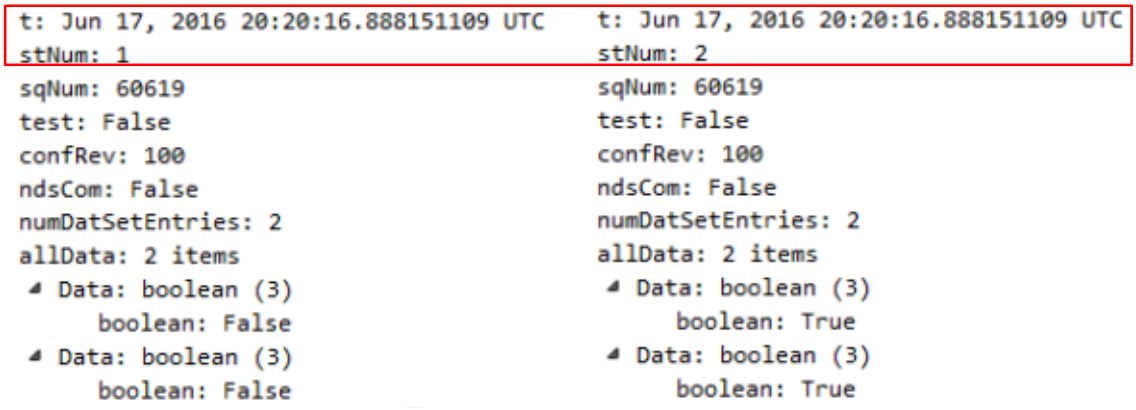

Figure 9. Wireshark Capture of Transmitted Messages with Same Time Stamp (t) and Incremented status number (stNum).

3.2.2. LibIEC61850

The test was repeated on the open source libiec61850 library implemented on two developed embedded devices. The subscribing IEDs processed the messages even though they have the same time stamps and incremented status numbers. The red LED in Figure 10 indicates that the message was processed and a digital output (HIGH) was produced signaling a circuit breaker trip.

It is noteworthy to point out that, according to the flowchart of Figure 3, IEC 61850 recommends checking for a message's time stamp only if it recognizes an stNum different than that of the previous message. The experiments revealed that when sending new messages with three-day-old time stamps exceeding the 2-min skew, they were processed as long as they had status numbers equal to or higher than the previous message.

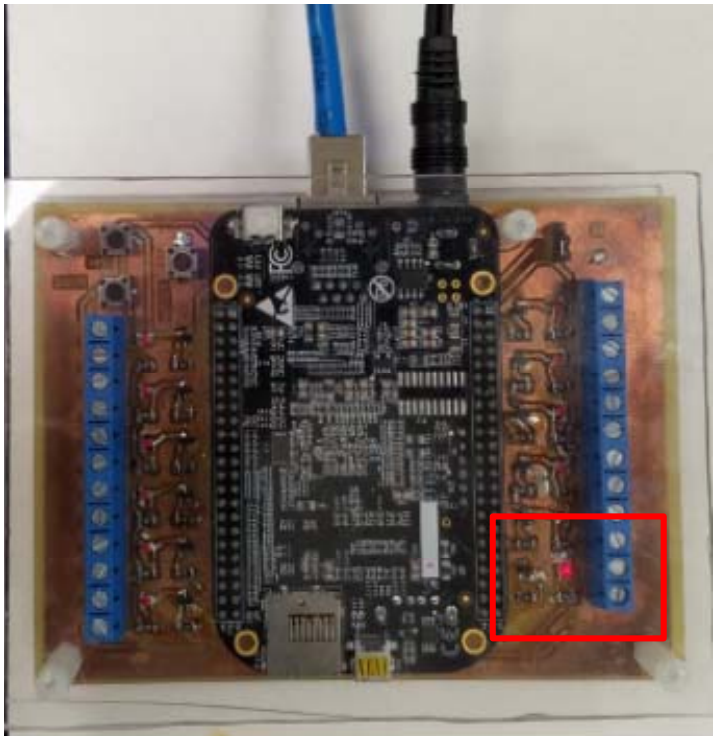

Figure 10. Subscribing IED Processing Fake Message (output port triggered as indicated in the red box).

3.3. Processing of Source MAC Address

All fake messages in the three tests performed above were sent from the virtual machine with a spoofed MAC address mimicking that of the publisher IED. That is, all IEDs subscribing to this message process the fake messages as if they were originating from the publisher IED. One common network defense procedure to counter MAC address spoofing incidents is to apply a MAC address filter to network switches. This will deny any machine connected to the network from sending a message with a source MAC address other than its own. After applying this filter, the messages with the fake MAC address were blocked from being sent over the network. However, in this test, we sent fake GOOSE messages with the MAC address of the virtual machine and altered data and noticed that the subscriber IEDs processed these messages. The circuit breaker's status changed from closed to open.

This test actually exploits a vulnerability in the GOOSE messaging protocol itself rather than its implementation in commercial devices. In the GOOSE protocol, subscribing IEDs use the APPID field to subscribe to desired GOOSE messages. Since the subscribing IEDs in this case do not check for the source MAC address, they will process any message with their defined APPID, regardless of its origin.

4. Guidelines for Proper Implementation of IEC 61850 GOOSE Protocol

Table 1 summarizes the results of the performed tests on both the commercial IEDs and libiec61850.

Table 1. Compliance Test Results.

Test	Commercial IED's	Libiec61850	Standard Practice
Processing of messages with lower stNum	Y	N	Discard message
Processing of messages with outdated time stamp	Y	Y	Discard message
Processing of messages with unspoofed MAC address	Y	Y	Not specified by standard

Y: Message processed; N: Message not processed.

It can be concluded from the results of the performed experiments that the actual implementation of IEC 61850 and its associated IEC 62351 cyber security standard on field devices depends on the vendors themselves. While vendors try to fully abide by the standard, differences in the implementation process might still be found as shown in this paper. The presence of such differences in the implementation process might expose the system to unwanted vulnerabilities, which might be exploited by prying eyes to launch cyber-attacks on the power grid [7,23]. As GOOSE messaging is the base protocol for critical applications such as power systems protection, any vulnerability in the system might lead to devastating consequences, ranging from system disturbances to complete blackouts.

Recent literature shows several security concerns about the IEC 61850 standard itself [8]. Therefore, in order to avoid additional exploits, extreme care must be placed on implementing IEC 61850 functionalities on commercial devices as well as abiding by the cyber security requirements set by IEC 62351. The analysis of the outcome of this work distinguishes between two levels of vulnerabilities: one on the device level and the other on the network level. On the device level, when devices are configured to communicate via GOOSE messages, the firmware on the subscribing IEDs must be tested for proper processing of messages as stated by the IEC 61850-8-1 and IEC 62351-6 standards. Since IEC 61850 does not provide any cyber security measure by itself, manufacturers should also make sure that their devices comply with IEC 62351 requirements. First, as stated by IEC 62351-6, messages with repeated or old status numbers must not be processed by subscribing devices. In fact, the open source libIEC61850 has an *IsValid()* function to ensure this, whereas the tested commercial IEDs lack this important check and thus processed fake messages. In addition, the association of a new time stamp with every increment of the status number must be checked for before publishing and/or processing messages. In the case of a new GOOSE event (i.e. an incremented stNum), it is important to compare a newly received message's time stamp with subscribing machines time to check whether or not the 2-min skew set by IEC 62351 was exceeded. Also, every change in the Data fields of a GOOSE message must be checked for association with an increment in the status number field. Finally, repeated messages with a change in their control signal (i.e. data fields) must be rejected. Message retransmissions should be identical with no alterations in any field except for an incrementing sequence number.

On the network level, in order to avoid compromised machines from publishing fake GOOSE messages using a spoofed MAC address, MAC filters must be applied to all switches in the substation's local area network to prevent MAC address spoofing. As concluded from the presented case studies, MAC filtering did not prevent subscribing IEDs from processing fake messages with unspoofed MAC addresses. Therefore, the source MAC address field in GOOSE messages must be checked to be belonging to an authenticated machine authorized to communicate via the GOOSE protocol within a substation's local area network. In fact, this vulnerability has not been accounted for, neither in IEC 61850 nor in IEC 62351. Until the standards cover this issue, it is up to the substation's network administrators to make sure that only authenticated devices are allowed to communicate via GOOSE messages.

5. Conclusions

Testing of two different available implementations of the IEC 61850 GOOSE messaging protocol was performed on commercial IEC 61850-based devices and on the open source libiec61850 library. The results demonstrated that different implementations of the same standard might lead to different behaviors even if the devices were present under similar conditions. Deviation from the actual procedures set forth by the IEC 61850 standard and its complementary cyber security IEC 62351 standard were found in the responses of the devices. From the experiments in this paper, it was found that the processing of the GOOSE messages status number was not properly implemented on the commercial devices as recommended by IEC 62351. This vulnerability provides a strong attack surface from prying eyes to inject malicious activities in power systems, such as the data manipulation attack demonstrated in this work. Additionally, all the tested devices were processing messages with old

time stamps, which is another attack surface for launching replay attacks. This point is of importance since GOOSE messages are broadcast in nature and, therefore, sniffing and replaying them is possible when an attacker is in the same LAN. Moving to the network level, it was shown that as long as it has a valid APPID field, a GOOSE message is processed whether originating from an authentic device or a malicious one. Since both IEC 61850 and 62351 do not clearly outline how to present clear rules for authenticating source MAC addresses, it is up to the substation network designers to take this issue into consideration and apply the appropriate defense mechanisms. Thus, this paper raises a serious issue as such devices are out in the field and are controlling critical and potentially dangerous power system operations. The work in this paper also proposes guidelines to better enhance utilization of IEC 61850. Proper processing of a message's source MAC address, better utilization of the time stamp field to check for messages' validity, and the association of new message content with a status number increment are advised.

Acknowledgments: This work was partially supported by a grant from the US Department of Energy.

Author Contributions: Mohamad El Hariri and Tarek Youssef conceived and designed the experiments, performed the experiments, and analyzed the data. Osama A. Mohammed is the main originator and supervisor who leads the project and edits the manuscript.

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

References

1. Fuloria, S.; Anderson, R.; Mcgrath, K.; Hansen, K.; Alvarez, F. The Protection of Substation Communications. In Proceedings of SCADA Security Scientific Symposium, Miami, FL, USA, 18–19 January 2010.
2. Akella, R.; Tang, H.; Bruce, M.M. Analysis of Information Flow Security in Cyber-Physical System. *Int. J. Crit. Infrastruct. Prot.* **2010**, *3*, 157–173. [CrossRef]
3. Buchholz, B.M.; Brunner, C.; Naumann, A.; Styczynski, A. Applying IEC standards for communication and data management as the backbone of Smart Distribution. In Proceedings of the IEEE PES General Meeting 2012, San Diego, CA, USA, 22–26 July 2012.
4. Brunner, C.; Naumann, A. The link between IEC 61850 and CIM/IEC 61968/61970—Experience from a smart grid project. In Proceedings of the Distributech, San Antonio, TX, USA, 24–26 January 2012.
5. Naumann, A.; Bielchev, I.; Voropai, N.; Styczynski, Z. Smart grid automation using IEC 61850 and CIM standards. *Control Eng. Pract.* **2014**, *25*, 102–111. [CrossRef]
6. Das, N.; Islam, S. Analysis of power system communication architectures between substations using IEC 61850. In Proceedings of the 5th Brunei International Conference on Engineering and Technology (BICET 2014), Bandar Seri Begawan, Brunei, 1–3 November 2014; pp. 1–6.
7. 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.
8. Youssef, T.A.; El Hariri, M.; Bugay, N.; Mohammed, O.A. IEC 61850: Technology Standards and Cyber-Security Threats. In Proceedings of the 16th IEEE International Conference on Environment and Electrical Engineering (EEEIC), Florence, Italy, 7–10 June 2016.
9. International Electrotechnical Commission. Communication networks and systems in substations—Specific communication service mapping (SCSM). IEC 61850-90-5, 2012.
10. Wen, J.; Hammond, C.; Udren, E.A. Wide-area Ethernet network configuration for system protection messaging. In Proceedings of the 2012 65th Annual Conference for Protective Relay Engineers, College Station, TX, USA, 2–5 April 2012; pp. 52–72.
11. Dehalwar, V.; Kalam, A.; Kolhe, M.L.; Zayegh, A. Review of IEEE 802.22 and IEC 61850 for real-time communication in Smart Grid. In Proceedings on the 2015 International Conference on Computing and Network Communications (CoCoNet), Trivandrum, India, 16–19 December 2015; pp. 571–575.
12. Cintuglu, M.H.; Ma, T.; Mohammed, O. Protection of Autonomous Microgrids using Agent-Based Distributed Communication. *IEEE Trans. Power Deliv.* **2016**. [CrossRef]

13. Cintuglu, M.H.; Mohammed, O.A. Multiagent-based decentralized operation of microgrids considering data interoperability. In Proceedings of the 2015 IEEE International Conference on Smart Grid Communications (SmartGridComm), Miami, FL, USA, 2–5 November 2015; pp. 404–409.
14. Kumar, S.; Das, N.; Islam, S. Performance evaluation of a process bus architecture in a zone substation based on IEC 61850-9-2. In Proceedings of the 2015 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), Brisbane, QLD, Australia, 15–18 November 2015; pp. 1–5.
15. Das, N.; Modi, H.; Islam, S. Investigation on architectures for power system communications between substations using IEC 61850. In Proceedings of the 2014 Australasian Universities Power Engineering Conference (AUPEC), Perth, WA, USA, 28 September–1 October 2014; pp. 1–6.
16. LibIEC61850 Open Source Library. Available online: http://libiec61850.com/libiec61850/ (accessed on 16 June 2016).
17. International Electrotechnical Commission. Communication networks and systems in substations—Specific communication service mapping (SCSM). IEC 61850-8, 2008.
18. Serra, M.; Castro, F. Using IEC 61850 for Teleprotection. In Proceedings of the 19th International Conference on Electricity Distribution, Vienna, Austria, 21–24 May 2007.
19. Lo, B.T.; Mendes, F.M.; Samaniego, L.H.; Oliveira, S.R. Cloud IEC 61850: Architecture and Integration of Electrical Automation Systems. In Proceedings of the 2014 Brazilian Symposium on Computing Systems Engineering (SBESC), Manaus, Brazil, 3–7 November 2014; pp. 13–18.
20. 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**. [CrossRef]
21. Python. Available online: https://www.python.org/ (accessed on 16 June 2016).
22. Scapy. Available online: http://www.secdev.org/projects/scapy/ (accessed on 16 June 2016).
23. Kush, N.; Ahmed, E.; Branagan, M.; Foo, E. Poisoned GOOSE: Exploiting the GOOSE protocol. In Proceedings of the Twelfth Australasian Information Security Conference (AISC 2014), Auckland, New Zealand, 20–23 January 2014.