Cyber security of the smart grid control systems: intrusion detection in IEC 61850 communication networks

Maëlle Kabir-Querrec

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTE UNIVERSITE GRENOBLE ALPES

Spécialité : **Automatique - Productique**

Arrêté ministériel : 25 mai 2016

Présentée par

Maëlle KABIR-QUERREC

Thèse dirigée par **Jean-Marc THIRIET, Professeur, Université Grenoble Alpes**, et
codirigée par **Stéphane MOCANU, Maître de Conférence, Institut Polytechnique de Grenoble**

préparée au sein du **Laboratoire Grenoble Images Parole Signal Automatique (GIPSA-lab), département Automatique**
dans **l'École Doctorale Electronique, Electrotechnique, Automatique, Traitement du signal (EEATS)**

Cyber security of the smart grid control systems: intrusion detection in IEC 61850 communication networks

Thèse soutenue publiquement le **28 juin 2017**,
devant le jury composé de :

Mme. Mireille Bayart
Professeur, Université de Lille, Rapporteur
M. Jérémie Guiochet
Maître de conférences, Université Toulouse III, Rapporteur
M. Eric Gnaedinger
Maître de Conférences, Université de Lorraine, Examinateur
Mme. Marie-Laure Potet
Professeur, Université Grenoble Alpes, Présidente
M. Jean-Marc Thiriet
Professeur, Université Grenoble Alpes, Directeur de thèse
M. Stéphane Mocanu
Maître de conférences, Institut Polytechnique de Grenoble, Co-encadrant de thèse
M. Yves-Gaël Billet
Automatique & Industrie, pour la société Euro-System, Invité

Remerciements

Au début de cette thèse, j'avais l'impression de me trouver au pied de l'Éverest avec pour mission d'atteindre le sommet équipée seulement d'une paire de tongues : compliqué. Trois ans et quelques mois plus tard, au moment de soutenir mes travaux en vue de l'obtention du Doctorat, je suis toujours aussi loin du sommet mais j'ai troqué mes tongues contre une bonne paire de chaussures de montagne ! De quoi poursuivre l'aventure de la Recherche avec enthousiasme.

Je tiens donc à exprimer ma reconnaissance à toutes les personnes qui m'ont aidés à m'équiper ces dernières années.

Merci à Eric Savary de la société Euro-Système d'avoir initié cette thèse et de m'avoir offert les moyens de réaliser ces travaux de recherche de façon confortable. Merci à lui ainsi qu'à mes encadrants de thèse, Jean-Marc Thiriet et Stéphane Mocanu du GIPSA-lab de m'avoir accordé leur confiance pour mener cette recherche. Merci pour votre accompagnement scientifique et humain, j'ai énormèment appris sur ces deux plans à votre contact. Ce bagage que vous m'avez transmis m'accompagne précieusement pour la suite de ma vie professionnelle.

Merci aux membres du jury d'avoir accepté d'évaluer mes travaux de thèse. Et en particulier merci à Mireille Bayart et Jérémy Guichet d'avoir endossé le rôle de rapporteurs. Merci à tous pour vos retours constructifs.

Merci aux doctorants, chercheurs et personnels du GIPSA-lab pour la bonne atmosphère de recherche à laquelle vous contribuez tous ! Merci à mes collègues d'Euro-Système et, pour les derniers mois, d'Automatique & Industrie : vous côtoyer a été enrichissant tant professionnellement que personnellement. Mention particulière à mes co-bureaux des différents lieux pour leur présence quotidienne, les bons moments partagés.

Merci à mes parents de m'avoir transmis leur curiosité du monde, chacun à leur manière, tellement différentes mais tellement complémentaires ! Vous me ferez toujours grandir. Merci à ma famille (incluse la belle-famille, est-il nécessaire de le préciser ?!) pour votre présence même à distance, votre soutien sous toute ses formes notamment gastronomique. Votre amour, m'est un trésor.

Merci aux copains... d'être encore mes copains !! J'aurais peut-être craqué à votre place ;-) Et en particulier, merci à mes colocs de 2 mois qui m'ont finalement accueillie bien plus longtemps.

Et merci à l'Amoureux, merci d'être.

Maëlle Kabir-Querrec

Abstract — Information and Communication Technologies have been pervading Industrial Automation and Control Systems (IACS) for a few decades now. Initially, IACS ran proprietary protocols on closed networks, thus ensuring some level of security through obscurity and isolation. Technologies and usages have evolved and today this intrinsic security does not exist any longer, though.This transition is in progress in the electricity domain, the power infrastructure turning into the "smart grid".

The IEC 61850 standard is key to the smart grid development. It is aimed at making interoperability possible in "Communication networks and systems for power utility automation". It thus defines a common data object model and a stack of protocols answering different purposes. Although the cyber risk in IACS is now widely acknowledged, IEC 61850 does not address cyber security in any way whatsoever.

This work tackles the question of cyber security through network intrusion detection in IEC 61850 networks, and more specifically in real-time GOOSE communications. The idea is to get the most out of the protocol specifications and system configuration while developing a tailored NIDS. This enables detection accuracy.

This dissertation consists of four chapters. The first two ones give an extensive state of the art about intrusion detection in IACS and cyber risk assessment, respectively. The two other ones are the proper contribution of this work. Chapter 3 first explores the cyber risk hanging over a generic substation example and how it may impact the system dependability attributes. It then proposes an extension of the IEC 61850 data object model to handle intrusion detection. After demonstrating the feasibility of intrusions in GOOSE communications, Chapter 4 explains how system configuration files can be leveraged to tune detection rules and presents the proposed algorithm. The latter was integrated into the open-source network traffic analyzer Bro by implementing a GOOSE parser. The dissertation ends with a proposition of an IEC 61850 communication architecture resilient to attacks on GOOSE protocol based on the detector.

Keywords: IEC 61850, network intrusion detection, NIDS, anomaly detection, behavior-based detection, industrial control systems, ICS, IACS, cyber security, GOOSE protocol, smart grid, Bro.

Résumé — Le standard IEC 61850 est clé pour le développement du smart grid ou réseau électrique intelligent. Il a pour objectif de rendre l'interopérabilité possible dans les "Réseaux et systèmes de communication pour l'automatisation des systèmes électriques". Bien que le cyber risque dans les systèmes de contrôle industriels fasse aujourd'hui consensus, la norme IEC 61850 ne traite pas de la cyber sécurité.

Ces travaux de thèse proposent de répondre à cette problématique de cyber sécurité à travers la détection d'intrusion réseau dans les systèmes IEC 61850, plus précisément dans les communication temps-réel GOOSE, impliquées dans la protection électrique.

Nous nous sommes tout d'abord attachés à identifier et comprendre les risques de cyber sécurité auxquels les postes IEC 61850 sont exposés. Les protocoles de communication de la pile IEC 61850 se révèlent vulnérables, en particulier GOOSE. Une preuve de faisabilité d'injection de fausses données dans les communications GOOSE a été faite. En guise de première réponse, nous explorons une mesure passive de sécurité : la détection d'intrusion. Nous proposons ainsi une extension au modèle d'information IEC 61850 dédiée à la détection d'intrusion dans les systèmes d'automatisation des services de distribution électrique. Nous proposons ensuite une approche de détection d'intrusion, de type comportementale, dont les règles de détection sont produites à partir des spécifications du protocole et de la configuration du système. Un analyseur syntaxique pour le protocole GOOSE a été intégré à l'analyseur de trafic réseau open source Bro. L'intégration de notre algorithme de détection à cet outil a permis de produire des résultats de performance préliminaires. Enfin, afin de tendre vers une solution globale de sécurité, nous avons imaginé une architecture du système de contrôle qui soit résiliente aux attaques GOOSE basées sur ce module de détection pour sécuriser activement les communications dans les environnements d'automatisation IEC 61850.

Mots clés : IEC 61850, détection d'intrusion réseau, NIDS, détection basée anomalies, détection comportementale, système de contrôle industriel, ICS, IACS, cybre sécurité, protocol GOOSE, smart grid, Bro.

Contents

List of Figures

List of Tables

Acronyms and abbreviations

ACSI	Application Communication Service Interface
AMI	Advanced Metering Infrastructure
ANSSI	*Agence Nationale de la Sécurité des Systèmes d'Information*
APDU	Application Protocol Data Unit
APT	Advanced Persistent Threat
ARP	Address Resolution Protocol
ASN.1	Abstract Syntax Notation One
BER	Basic Encoding Rules
CEN	European Committee for Standardization
CENELEC	European Committee for Electrotechnical Standardization
CI	Critical Infrastructure
CLUSIF	CLUb for Security of Information in France
CPU	Central Processing Unit
CRAMM	CCTA Risk Analysis and Mangement Method
DCS	Distributed Control System
DER	Distributed Energy Resources
DNP3	Distributed Network Protocol 3
DoS	Denial of Service
EBIOS	*Expression des Besoins et Identification des Objectifs de Sécurité* - Expression of Needs and Identification of Security Objectives
EMS	Energy Management System
ENISA	European Union Agency for Network Information Security
ETSI	European Telecommunications Standards Institute
FMEA	Failure Mode and Effects Analysis
FMECA	Failure Mode, Effects and Criticality Analysis
FNR	False Negative Rate
FPR	False Positive Rate
FTP	File Transfer Protocol
GOOSE	Generic Object Oriented Substation Event
HAZOP	Hazard and Operability Study
HIDS	Host-based Intrusion Detection System
HMAC	Hash Message Authentication Code
HMI	Human Machine Interface
HTTP	HyperText Transfer Protocol
IACS	Industrial Automation and Control System
ICS	Industrial Control System
ICT	Information and Communication Technologies
ICMP	Internet Control Message Protocol
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device

IDS	Intrusion Detection System
IP	Internet Protocol
IPS	Intrusion Prevention System
ISA	International Society of Automation
ISMS	Information Security Management System
ISO	International Organization for Standardization
IT	Information Technology
LAN	Local Area Network
LN	Logical Node
MAC	Media Access Control
MD	Message Digest
MEHARI	MEthod for Harmonized Analysis of RIsk
MITM	Man-In-The-Middle
MMS	Manufacturing Message Specification
MTU	Master Terminal Unit
NIDS	Network-based Intrusion Detection System
NIST	National Institute of Standards and Technology
NSM	Network and System Management
NTP	Network Time Protocol
OCTAVE	Operationnally Critical Threat, Asset, and Vulnerability Evaluation
OSI	Open Systems Interconnection
OT	Operation Technology
PICOM	Piece of Information for COMmunication
PLC	Programmable Logic Controller
RBAC	Role-Based Access Control
RNRP	Redundant Network Routing Protocol
RTU	Remote Terminal Unit
SAS	Substation Automation System
SAT	Site Acceptance Test
SCADA	Supervisory Control And Data Acquisition
SCSM	Specific Communication Service Mapping
SGAM	Smart Grid Architecture Model
SGIS	Smart Grid Information Security
SIEM	Security Information and Event Management
SNMP	Simple Network Management Protocol
SRA	Safety, Reliability, Availability
SV	Sampled Value
TCP	Transmission Control Protocol
TNR	True Negative Rate
TPR	True Positive Rate
VLAN	Virtual Local Area Network

Introduction

As industrial infrastructures are more and more reliant on automation and interconnected networks, the frontier separating those two worlds (corporate VS. field plant), known as the "air gap", has melted away. Information and Communication Technology (ICT) has been pervading every day more deeply industrial plants where operations are run through many digital systems and networks, giving what is now called Operation Technology (OT). These Industrial Control Systems (ICS) primarily ran proprietary protocols on closed networks, thus bringing security through obscurity and isolation. Technologies and usages have evolved and today this intrinsic security does not exist any longer, though, because of the ever wider use of standardized protocols and the market pressure for making information available. This digital or cyber layer added upon the physical system offers new services and capabilities in terms of both local control and global, possibly remote, management. But it also brings its own failure vectors and vulnerabilities. Thus, this double dimension of such industrial systems makes them face risks peculiar to the physical process / system along with cyber risks... and also risks specific to cyber-physical systems due to strong interconnections of both parts.

This new paradigm is true for all industrial domains but particularly worrying for Critical Infrastructures. Critical Infrastructures (CI) are assets essential for a society or economy, and whose failure, breakdown or misuse may impact highly sensitive aspects such as security, health, environment... The power grid is a CI. It encompasses all facilities for electricity generation, transport and distribution. Recently, the term "smart grid" appeared to designate the electrical infrastructure being more and more enriched with a digital layer, which allows functions digitalization, remote and global management and some intelligence.

Two events are worth mentioning while considering the cyber-physical characteristic of the smart grid and especially its sensitivity to cyber incidents and attacks. The first one is known as the North East America blackout of 2003. A defective state estimator was recalibrated but the monitoring tool was forgotten and not restarted. Added to that, there was an informatics bug blocking the alarm system for one hour, thus depriving operators of any state change information. This combination of unfortunate events resulted into a cascading failure and finally this great blackout [6]. The second event is closer to us, both in time and space. On December 15^{th} 2015, cyber attackers entered three Ukrainian electricity providers' networks and ICS. They were able to remotely control field devices at substations and disconnect several portions of the Ukrainian grid in a coordinated manner. A malware was used to render some devices inoperable and unrecoverable forcing operators to switch to manual mode. The resulting power outages impacted 225,000 customers for a few hours [100]. These two blackouts illustrate how much availability, reliability and safety of the energy grid rely on the associated information and communication system integrity.

"*Communication networks and systems for power utility automation*" is the topic of the IEC 61850 standard, whose purpose is to answer a global will of technologies simplicity and interoperability. This international standard, first edition published in 2003 and revised version in 2013, is deemed as key for the smart grid deployment as it defines a common data model framework along with a protocols stack for power systems automation. IEC 61850 does not answer this cyber threat, though.

This dissertation addresses the lack of cybersecurity problem of the IEC 61850 standard. We tried to identify and understand the cyber security risks, to which IEC 61850 substations are exposed. The communication protocols of the IEC 61850 stack prove to be vulnerable, especially the one involved in electrical protection functions, which are in charge of isolating faults and preventing them from spreading, the GOOSE protocol. As a first answer, we explore a passive security measure: intrusion detection. We thus propose an extension to the IEC 61850 information model dedicated to intrusion detection in power automation systems. We then propose an implementation of an anomaly-based intrusion detection system that leverages protocol specifications and system configuration to tune detection rules. The open source network traffic analyzer Bro was used. We imagined a control system architecture resilient to GOOSE attacks based on this detection module to actively secure communications in IEC 61850 automation environments.

The main contributions of the presented work are:

- an assessment of the cyber risk threatening IEC 61850 substations,
- an extension to the IEC 61850 data object model for intrusion detection,
- an anomaly-based detection module monitoring GOOSE communication and leveraging protocol definition and system configuration,
- and a proposition of an IEC 61850 ICS architecture resilient to attacks on GOOSE communications.

This dissertation is organized as follows. *Chapter 1* defines the main concepts at stakes, introduces international standards relevant to this work and gives a detail state of the art of anomaly-based network intrusion detection in ICS. *Chapter 2* presents existing risk management approaches, from both Dependability and Information Security domains, and attempts of applying them to industrial cyber-physical systems in order to deal with the cyber security risk. *Chapter 3* starts with the cyber security risk assessment of an example IEC 61850 substation and then presents the proposed extension to the standard information model for intrusion detection. *Chapter 4* first presents an experimental study of GOOSE protocol vulnerabilities. Then come the proposed detection algorithm and its integration into Bro. It ends with a proposition of automation architecture resilient to GOOSE attacks. *Conclusion* sums up the outcomes of this work and make some propositions for further work.

List of publications

- Maëlle Kabir-Querrec, Stéphane Mocanu, Pascal Bellemain, Jean-Marc Thiriet, and Eric Savary. *Architecture des systèmes d'automatisation des postes résiliente aux attaques des trames GOOSE.* In Journées C&ESAR 2015, November 2015.
- Maëlle Kabir-Querrec, Stéphane Mocanu, Pascal Bellemain, Jean-Marc Thiriet, and Eric Savary. *Corrupted GOOSE Detectors: Anomaly Detection in Power Utility Real-Time Ethernet Communications.* In GreHack 2015, Grenoble, France, November 2015.
- Maëlle Kabir-Querrec, Stéphane Mocanu, Jean-Marc Thiriet, and Eric Savary. *Power Utility Automation Cybersecurity: IEC 61850 Specification of an Intrusion Detection Function.* In 25th European Safety and Reliability Conference (ESREL 2015), Zürich, Switzerland, September 2015.
- Maëlle Kabir-Querrec, Stéphane Mocanu, Jean-Marc Thiriet, and Eric Savary. *A Test bed dedicated to the Study of Vulnerabilities in IEC 61850 Power Utility Automation Networks.* In 21st IEEE Emerging Technologies and Factory Automation, Berlin, Germany, September 2016.

Chapter 1

Intrusion detection in smart grid control systems

Introduction

This first chapter starts with a few important definitions that will help the reader to picture the context of this work. That is the purpose of *section 1*. *Section 2* quickly introduces the main standards to consider when addressing the question of cyber security in IEC 61850 automation environments, especially through intrusion detection. *Section 3* discusses cyber security from the Operational Technology (OT) angle compared to the traditional cyber security approach of the IT domain. This chapter last section gives a state of the art of ICS-oriented anomaly-based network intrusion detection.

1.1 Definitions

1.1.1 Industrial Control System

Industrial Control System (ICS) or Industrial Automation and Control System (IACS) is a general term that encompasses all *"control-command systems" (sic)* as stated by the French Network and Security Agency (ANSSI[1]) [8]. According to the ANSSI definition, the term IACS *"designates a set of human and material resources designed to control or operate technical installations (consisting of a set of sensors and actuators).* U.S. National Institute of Standards and Technology (NIST) details a bit the technical components of IACS, which include *"supervisory control and data acquisition (SCADA) systems, distributed control systems (DCS), and other control system configurations such as Programmable Logic Controllers (PLC) often found in the industrial sectors and critical infrastructures"* [131]. And the international standard IEC 62443 about "Security for Industrial Automation and Control Systems" even goes a step further considering qualitative criteria: IACS is a *"collection of personnel, hardware, and software that can affect or influence the safe, secure, and reliable operation of an industrial process."* . It details: *"associated information systems such as advanced or multivariable control, online optimizers, dedicated equipment monitors, graphical interfaces, process historians, manufacturing*

[1] ANSSI: *Agence Nationale de la Sécurité des Systèmes d'Information*

execution systems, and plant information management systems" and *"associated internal, human, network, or machine interfaces used to provide control, safety, and manufacturing operations functionality to continuous, batch, discrete, and other processes."* [68].

These definitions are quite broad and cover many sectors (e.g. energy, transport, water supply, industry, building management), which have diverse missions (manufacturing, providing a service...) and diverse structures (one or many distributed sites, different levels of complexity...).

IEC 62443 standard and NIST guide differentiate DCS and SCADA. While the first one is a local *"control system in which the system elements are dispersed but operated in a coupled manner"*, the second one may be more widespread and cover *"loosely coupled distributed monitoring and control system"* [68]. NIST adds another distinction: a SCADA also has the purpose of collecting and transmitting field data from allover the sytem to make them available as notifications to operators at a supervision center and to feed some data historians.

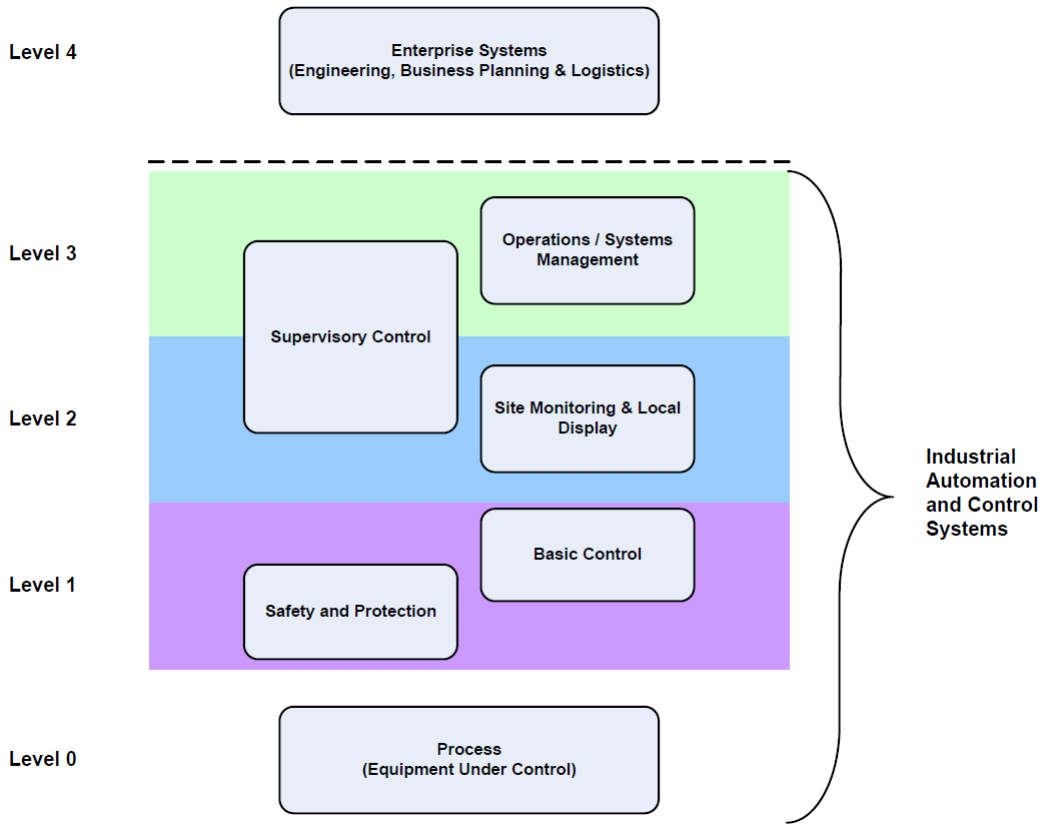

Figure 1.1: Industrial Automation and Control System reference model [68]

Figure 1.1 shows the IACS reference model used in the ISA99 series of standards and recalled in IEC 62443:1.1 [68]. The model of digital industrial systems presented in [41] echoes this international pattern and fits all kinds of IACS typologies (industrial plants,

building management systems and utilities). IACS is traditionnally split into five layers:

- **Level 4 - Enterprise Business Systems:** Systems at this level are a matter of traditional IT. They provide the functions involved in business planning and logistics. They actually are out of the scope of the industrial control system but their interfaces with it requires attention as they may be potential entry points for attackers.
- **Level 3 - Operations Management:** Activities conducted at this level relate to scheduling, optimization and maintenance. To this end, data are collected from the lower levels and analyzed off-line.
- **Level 2 - Supervisory Control:** This is where monitoring and control of the process is run. This includes data collection and processing for automatic control functions and history data bases, HMIs displaying information to operators about the process state including alarms. This level also often includes engineering stations for programming controllers of the lower layer.
- **Level 1 - Local or Basic Control:** This level's functions are involved in sensing and manipulating the physical process. The typical devices of level 1 are controllers (DCS, PLCs, RTUs, IEDs...) responsible for reading data from sensors, executing algorithms when relevant and sending resulting signals to actuators. They have a vision of the portion of the process, of which they are in charge, and they can communicate information about it or their own state to the supervisory level or receive commands from it. Safety and protection functions are implemented at this level as well. They may be as stand-alone systems or programmed in the same controllers as basic process control functions.
- **Level 0 - Process:** Level 0 is the actual physical process where sensors and actuators are set out.

1.1.2 Smart grid

The European Standards Organizations CEN and CENELEC write the following definition of "smart grid" on their smart grid-dedicated webpage: *"A smart grid is an electricity network that can integrate in a cost-efficient manner the behaviour and actions of all users connected to it (generators and/or consumers) in order to ensure economically efficient, sustainable power system with high levels of quality and security of supply and safety. Smart grids allow companies and households to produce electricity (for example – using photovoltaic panels or wind turbines) and sell it on to other consumers through existing networks."* [20]. For the U.S. Department of Energy, the Smart Grid as defined above is *"a long-term promise"* distinguishing it from what is ongoing: *"a smarter grid"* that is based on the existing grid and *"offers valuable technologies that can be deployed within the very near future or are already deployed today"* [104]. So what is called the "smart grid" is actually the historical grid enhanced with today and future technologies, especially the Information and Communication Technologies (ICT), that make possible

to provide the power grid with some "intelligence". Where monitoring and control were mechanical and local, ICT widen the field: algorithms embedded into micro-processor-based systems can perform complex operations, operators can get information about the system remotely and almost in real-time, etc.

The objectives of the smart grid identified by the U.S. Department of Energy include but are not limited to:

- make the consumer a *prosumer*, that is an informed and active consumer and possibly a producer,
- integrate all kinds of generation and storage solutions, especially Distributed Energy Resources (DER),
- automatically detect disturbances and prevent outages,
- operate resiliently to cyber attacks and natural disasters.

These ambitions imply an ever larger and more complex grid infrastructure. To help clarify the boundaries of this rather recent concept and define a common framework and vocabulary for all the European stakeholders, the European Standards Organizations CEN, CENELEC and ETSI[2] have defined a reference model: the Smart Grid Architecture Model (SGAM) framework [22], shown in Figure 1.2. This reference framework is expected to enable interoperability (e.g. through the definition of standards) as it is considered a key enabler of the smart grid. This solution and technology-independent model consists of five interoperability layers for business objectives and processes, functions, information exchange and models, communication protocols and process components. Another partitioning completes this model with two more dimensions, the *smart grid plane*: five physical domains cover the whole electrical energy conversion chain (bulk generation, transmission, distribution, DER, customer premises) and six zones represent the hierarchical levels of lower system management (process, field, station, operation, enterprise, market).

As the smart grid is a very heterogeneous system involving numerous of different stakeholders, a common and neutral reference model is essential to develop today smarter grid in a consistent and interoperable manner towards an accomplished smart grid.

[2]They are the three European Standards Organizations: the European Committee for Standardization (CEN), the European Committee for Electrotechnical Standardization (CENELEC) and the European Telecommunications Standards Institute (ETSI).

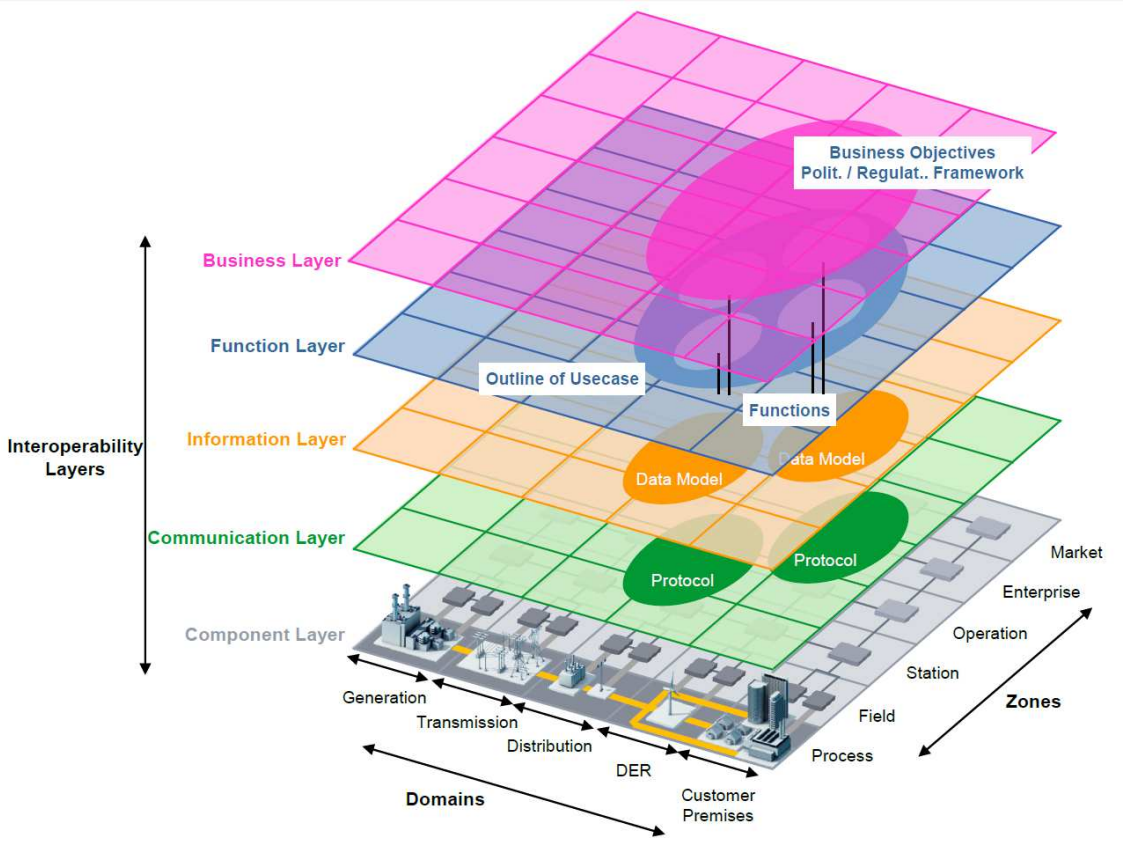

Figure 1.2: SGAM framework [22]

1.1.3 Electrical substation

A substation is part of an electrical generation, transmission and distribution infrastructure. Its function is generally to change voltage level between two portions of the grid, e.g. between a high voltage transmission line and a lower voltage distribution line. Thus, in most substations are one or more transformers. Another important role of a substation is the protection of the primary assets, that is the grid itself, e.g. to isolate incoming and outgoing lines from each other and prevent electrical faults from spreading. A substation typically consists of many interconnected elements such as transformers, busbars, switches and circuit breakers, wires, etc. that may share a common functionality and be grouped into a subpart called a **bay**. IEC 61850 glossary part describes bays as follows: *"These bays comprise a power system subset to be protected, for example a transformer or a line end, and the control of its switchgear that has some common restrictions such as mutual interlocking or welldefined operation sequences."* [65].

IEC 61850 standard defines a few typical transmission and distribution substations to illustrate some of the concepts it defines. Figure 1.3 depicts a topology example of a medium size distribution substation and a possible assignment of bay units resulting into a set of sixteen bay units.

Each bay may be managed by a generic IED called a bay controller. The bay level

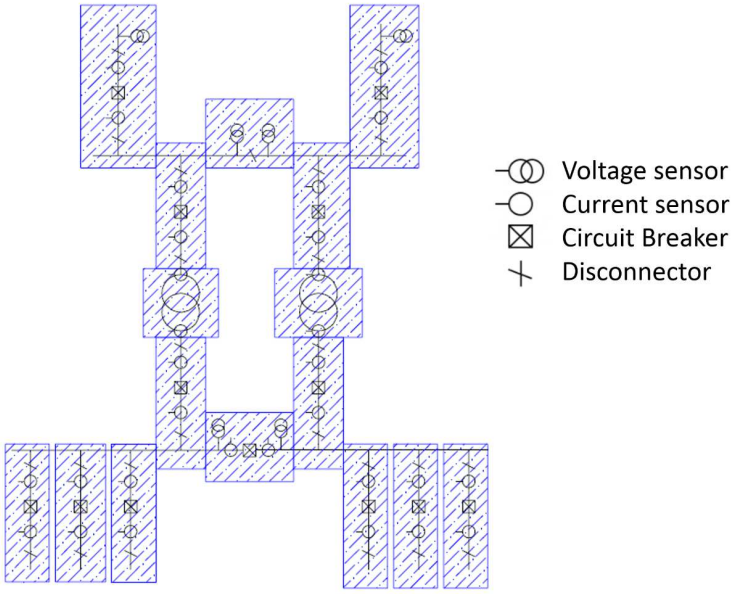

Figure 1.3: Example of bays' assignment into a typical medium-size distribution substation [73]

represents an additional control level below the overall station level. In the IACS reference model shown in Figure 1.1, level 1 would be the bay level and level 2 the substation level, process of level 0 being the power grid. What is called the **Substation Automation System (SAS)** is actually the substation IACS and includes all the IEDs and communication networks.

1.1.4 Intelligent Electronic Device

The introductory part of IEC 61850 standard defines an IED as *"any device incorporating one or more processors with the capability of receiving or sending data/control from or to an external source (for example, electronic multifunction meters, digital relays, controllers)"* [73]. IEC 61850 glossary part [65] adds that it is a *"device capable of executing the behaviour of one or more, specified logical nodes in a particular context and delimited by its interfaces"*. The terms "logical nodes" refer to a subfunction in IEC 61850 environment and is defined in section 3.3.

IEDs can be considered as specialized PLCs (Programmable Logic Controller) for power utility automation system. PLCs are programmable electronic devices designed to supplant some of the wired logic in many industrial systems. As stated in the definition above, an IED has computational capabilities and communication interfaces. It also may have integrated sensors (current and voltage transformers) and actuators (circuit breakers, switches), and binary inputs/outputs, hence being all at once an electronic multifunction meter, a digital relay and a controller.

1.2 Normative framework

Some standards are worth mentioning while considering cyber security of IEC 61850 automation systems, especially intrusion detection.

1.2.1 IEC 61850 "Communication networks and systems for power utility automation"

1.2.1.1 IEC 61850: a TC 57 standard

IEC 61850 standard [73] has been written and is now maintained by the International Electrotechnical Commission Technical Committee 57 (IEC TC 57) "Power Systems management and associated information exchange". As introduced on its webpage[3], IEC TC 57's role is to *"[develop] and [maintain] international standards for power systems control equipment and systems including EMS (Energy Management Systems), SCADA (Supervisory Control And Data Acquisition), distribution automation, teleprotection, and associated information exchange for real-time and non-real-time information, used in the planning, operation and maintenance of power systems"*.

1.2.1.2 Scope

The standardization process started in the 90'. IEC 61850 standard was first published in 2003 and a revised version was issued in 2013. Originally entitled "Communication networks and systems for substation", its scope was limited to the SAS (Substation Automation Systems), specifying communication between IEDs and the related systems requirements. Edition 2 is meant to expand the standard application areas (s.a. automation of wind power systems, hydro power systems, distributed energy resources, etc.) and hence, among other modifications and additions, it has been renamed "Communication networks and systems for power utility automation". IEC 61850 is a pillar standard to the smart grid deployment.

1.2.1.3 Objective

As stated in the standard introductory part, IEC 61850 main objective is to make IEDs interoperability a reality and generalize the use of communication technologies in power utility automation systems. As defined in the standard, interoperability is the ability for all the IEDs of a SAS *"to operate on the same network or communication path sharing*

[3] http://tc57.iec.ch/index-tc57.html

information and commands" [73]. This interoperability aims at enhancing communication efficiency among IEDs and is made possible through the use of standard communication protocols. Thus the standard's objective is to develop such communication protocols that *"meet functional and performance requirements, while supporting future technological developments"*.

1.2.1.4 IEC 61850 structure

Concretely, IEC 61850 brings two main contributions. On one hand, it defines a data object model for a common information representation and handling. Related parts are 5, 7.1, 7.2, 7.3 and 7.4. The basics are introduced in section 3.3 of this dissertation. On the other hand the standard proposes a dedicated communication architecture along with three communication protocols specific to exchanging entities and their constraints. It specifies requirements regarding syntax, semantics and performances. Related parts are 5, 8.1 and 9.2. Features of the communication architecture, including the operational protocols, are described in section 3.1.

Table 1.1 lists the main parts of the IEC 61850 standard along with their current edition number and publication year. Last column refers to this dissertation sections, in which the reader can find relevant elements of the considered part regarding our work. Not all parts were relevant to our work and we did not use the entire contents of the purposeful parts either. For exhaustive explanations, one shall refer to the IEC 61850 standard itself.

1.2.1.5 Cyber security in IEC 61850

IEC 61850 Edition 1 identifies two main threats that a SAS shall counter by implementing adequate security features [64]. They are Denial of Service (DoS), that is hindering legitimate access to and use of devices and functions, and illegitimate use, that is attempt to use the SAS in a malicious manner. For the latter, Annex A of IEC 61850-3 Edition 1 recommends that communication to and into a SAS supports authorization validation and access privileges.

Regarding implementation of IEC 61850 security functionalities, a discussion about security material available in the data object model defined by the standard can be found in section 3.3.4.

Table 1.1: IEC 61850 main parts

IEC 61850: Communication networks and systems for power utility automation				
Part	**Title**	**Ed.**	**Year**	**Section**
1	Introduction and overview	2	2013	1.2.1
2	Glossary	1	2003	/
3	General requirements	2	2013	3.3.4
4	System and project management	2	2011	/
5	Communication requirements for functions and device models	2	2013	3.3
6	Configuration description language for communication in electrical substations related to IEDs	2	2009	4.2
7.1	Basic communication structure - Principles and models	2	2011	3.3
7.2	Basic information and communication structure - Abstract communication service interface (ACSI)	2	2010	3.1
7.3	Basic communication structure - Common data classes	2	2010	3.3, App. B and C
7.4	Basic communication structure - Compatible logical node classes and data object classes	2	2010	3.3, App. B and C
8.1	Specific communication service mapping (SCSM) - Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3	2	2011	3.1
9.2	Specific communication service mapping (SCSM) - Sampled values over ISO/IEC 8802-3	2	2011	
10	Conformance testing	2	2012	/

1.2.2 IEC 62351 "Power systems management and associated information exchange - Data and communications security"

IEC 61850 first edition raised criticisms about data and communication security as very little was available (for a discussion about security material of the IEC 61850 standard, refer to section 3.3.4). Regarding its second edition, *"security issues are solved by the IEC 62351 series"*. This standard (first edition 2007) aims at enhancing communication protocols security used in power utilities as it contributes to system safety and reliability [66], as stated on the IEC webstore page introducing Part IEC 61850:7.2[4]. Thus, IEC 62351 main purpose is to *"undertake the development of standards for security of the communication protocols developed by the IEC TC 57"* [66] as it contributes to the safety and reliability of power utilities.

[4]https://webstore.iec.ch/publication/6015

1.2.2.1 Threats and security attributes identified in IEC 62351

Regarding the general security problem, IEC 62351 standard differentiates inadvertent threats, like safety or equipment failures, carelessness and natural disasters, from deliberate threats, including disgruntled employees, industrial espionage, vandalism, cyber hackers, viruses and worms, theft and terrorism [66]. Specifically for cyber security question, the standard gives the following requirements: confidentiality, integrity, availability, non-repudiation (see section 1.3.6 for definitions). RBAC (Role-Based Access Control) is proposed as a confidentiality security countermeasure. It then gives examples of possible attacks highlighting what requirement(s) would be compromised.

1.2.2.2 IEC 62351 security approach

Deploying pertinent security measures (just what should be secured and to the right degree) requires assessing the risk of the system and developing a security policy. IEC 62351 focuses on authentication, encryption of authentication keys and messages (when involved systems can handle it), prevention of playback and spoofing, tamper detection, monitoring of communications (availability of devices and resources), etc. for the IEC TC 57 protocols dedicated to power control.

IEC 62351-6 "Security for IEC 61850" defines what security measures may be carried out inside the substation perimeter as for multicast protocol GOOSE. For GOOSE applications *"requiring 4msec response times, multicast configurations, and low CPU overhead, encryption is not recommended"* [67]. According to the IEC 62351 standard, the only security measure than can possibly be implemented for such protocols is authentication of messages using digital signature with symmetric keys – HMAC (Hashed Message Authentication Code). However performance testing of such an implementation is still at an early stage (see section 1.3.3 for further discussion).

In the IEC 62351 introductory part, intrusion detection is identified as key to a full end-to-end security framework but out of the scope of the standard [66]. Part 7 of the standard still gives further detail about the IDS concept. It defines two concepts for intrusion detection: Network and System Management (NSM) whose purpose is to monitor the health of networks and systems, and Intrusion Detection System. IEC 62351 distinguishes two types of IDS: *passive* observation techniques and *active* security monitoring. *Passive* IDS are network-based IDS (NIDS) hardware monitors sniffing traffic at some points of the network and do not require to modify existing equipment. It thus makes security upgrades easier and less expensive to implement. The *active* approach involves that security monitoring shall be a design criteria for networks and control systems. Active IDS modules are host-based (HIDS) software components providing end systems with the ability to identify and send additional security information from each layer of the protocol stack to a security agent. Such an active security monitoring architecture is expected to

be able to detect intrusions at the application level... which may not be the case for a passive observation technique. Both types of IDS approaches shall implement NSM data objects and send them up to a security management unit, which carries out further analysis. NSM data include objects implemented for the purpose of security and available information from legacy systems that may provide additional awareness.

1.2.3 IEC 62443 "Industrial communication networks - Network and system security"

While considering security in power infrastructures, it is worth mentioning IEC 62443 standard (also known as ISA 99/IEC 62443) about security for IACS. This standard ambition is to build on existing cyber security standards, adapting approaches and technical recommendations to IACS peculiarities. IEC 62443 defines requirements about security policies and procedures, application of measures at the system level and development of IACS components [68]. In the IEC 62443 standard, the concept of control systems security is applied in the broadest sense as it aims at encompassing all types of facilities and systems from all industries.

1.2.4 IEEE C37.240 "Cyber security Requirements for Substation Automation, Protection, and Control Systems"

This standard presents engineering practices that may be used as sound cyber security measures of the automation, protection and control systems of a substation. However it highlights how important it is to make sure selected methods are technically and operationally feasible for every considered installation.

It identifies NIDS as one SAS cyber security requirements [62].

1.2.5 IEEE 1686 "Intelligent Electronic Devices Cyber Security Capabilities"

As stated in IEEE 1686-2013 description paragraph from the IEEE website[5], *"Security regarding the access, operation, configuration, firmware revision and data retrieval from an IED are addressed. Communications for the purpose of power system protection (teleprotection) are not addressed in this standard."* [61].

Thus, IEEE 1686-2013 does not tackle cyber security of communications.

[5] https://standards.ieee.org/findstds/standard/1686-2013.html

All these standards inevitably present some redundancies because their scopes overlap. The important thing is that they are consistent with each other. We expect IEC 62443 to become a significant reference for ICS cyber security, with other standards bringing details for specific fields, such as IEC 62351 for power domain. IEC 62443 standard is still a work in progress though and is built on IT cyber security standards, whose concepts and approaches must be adapted to OT specificities.

1.3 Comparison of IT and OT cyber security

Control systems are increasingly being integrated with non-IACS systems and applications through various communication networks, providing significant business benefits. Despite this high level of integration, there exist major differences between IT and OT communication systems and networks. Experts all agree that specificities of industrial systems make tedious if not impossible or irrelevant to transfer security tools and processes from IT to OT [38], [68], [130]. Developing tailored countermeasures is a necessity and such specificities should be put to use in this process.

1.3.1 Lifetime

IT domain knows a cycle of 3 to 5 years of renewal of informatics system stocks mainly because of rapid technological advances and users' appetence for new functionalities and greater computing capacities. Industrial control systems have a typical 20-year lifetime, or more [38]. That corresponds to the support duration and kind of oblige the user to stick to solutions prescribed by the vendor as the amount of time and resources to engage a change is quite important. Most of industrial plants thus have equipment developed and deployed at a time when no cyber threat was to be feared or no one was aware of it. This is what the terms "legacy systems" encompass.

This difference also holds to the continuity of service, the ultimate priority of most industrial infrastructures. Industrial stakeholders' attitude may be broadly summarized by the following question: "If it works, why try and risk it to not operate properly again?". In industrial environments, applying changes (e.g. application of patches) to the equipment is complicated and requires careful planning as it disturbs the process or even requires it to stop for retrofitting. Validating that applied changes do not impede the system of correctly fullfill its operational mission must be done through careful testing (site acceptance test – SAT).

1.3.2 Performances and time criticality

Data processing and transfer, and execution of functions in industrial environments answer real-time constraints. It means that delays must be controlled, which can but does not necessarily mean the fastest treatment [153]. Such time criticality does not hold for IT systems where delays are generally accepted [38]. As a comparison, the most stringent time delay requirement is of 150-200ms to supply voice-over-IP and multimedia services, while a trip signal in a SAS has an end-to-end transfer time of at most 4ms.

While latency is of the highest importance to industrial communications, IT networks are demanding on throughput, which is the condition to supply quality data services to all users [138].

1.3.3 Resources

Most IACS devices have limited computing capabilities, memory resources and bandwidth capacities [153]. This hinders encryption and authentication methods deployment in IACS as the additional latency is not compatible with the hard real-time constraints. To the present day available hardware does not have enough computation power to meet the time constraints while implementing authentication or encryption. Regarding specifically the real-time protocol GOOSE of the IEC 61850 stack, which is the main focus of this dissertation, several papers confirm this idea. A study by Fuloria et al. [44] showed that even high-end processors (such as 32-bit Intel and ARM cores) cannot encode and decode digital signatures for GOOSE and SV messages and still meet time requirements. Therefore 32-bit Intel and ARM cores are generally incapable of computing and verifying a digital signature using the 1024-bit-key RSA algorithm within 4ms, which is the end-to-end transfer time required by IEC 61850 standard for GOOSE messages (see section 3.1.2 for an introduction to this protocol as defined in the IEC 61850 standard). As an example, the authors measure a 7-8ms processing time for a RSA 1024-bit private key signature operation, as specified by IEC 62351-6 first edition, on a 1.7GHz Intel core using the OpenSSL library. And they note that such an operation can hardly be parallelized. Hoyos et al. [55] note that controllers, such as IEDs, installed in industrial environments often are fan-less, installed in closed cases to avoid dust, water or insects. Thus, power dissipation of CPU is restricted and embedded processors are generally slower than the 1.7GHz processor used in the previous study and processing times would be even longer. S. Hong et al. [54]-[53] have studied the performance degradation caused by incoming packet processing for encryption/decryption and signature purpose using IEC 61850 protocols. The authors think that embedded control devices will have to be equipped with multi-core processors to fulfill the control, communication and security functions expected from them. Their study shows that HMAC approach has the most promising performance for message authentication. This conclusion echoes the work done by Hohlbaum et al. [50] who show that use of assymetric keys for signing GOOSE and SV messages, as stipulated

by first edition of IEC 62351, does not comply with time requirements and processing capabilities of control devices. They advise to consider HMAC. In October 2009 these findings were presented to IEC TC 57 WG 15 who revised IEC 62351-6 accordingly. For Yang et al. [144], use of a MD-5 or SHA-1 algorithm to hash a checksum would be relevant for GOOSE communication, though. A hash algorithm is a one-way function computing a fixed-length bit string from data, here a message, of arbitrary size. They have measured processing time of several security algorithms applied to GOOSE messages using a 700MHz Pentium III core with 256KB cache and 256MB RAM. Their study outcomes corroborate the idea that encryption cannot be performed for GOOSE communication.

Although, performance requirements of some applications may be compatible with some encryption and authentication methods, industrial stakeholders stay reluctant to their implementation [7]. This may be explained by IACS long lifetimes and the investment to apply any change, making certain the whole system still comply with the infrastructure objectives and regulatory rules.

1.3.4 Protocols and network topologies

Automation networks are usually more constrained than traditional networks [139]. In IT networks, big amounts of data are exchanged at rather unpredictable moment, they are composed of a lot of connection points that can appear and disappear at any time. In contrast, industrial networks are often characterized by relatively fixed topologies: services and entities using the networks are known *a priori*. Industrial protocols are considered simpler or at least better specified, and a limited number of them is used in an IACS. Communication paths (who communicates with whom?...) and patterns are defined and rather regular.

1.3.5 Cyber security culture

Cyber security of IT systems is mature: awareness of cyber security risk is high while it is poor in OT sector, security testing and audits are common practices while they occasionnaly happen in case of outages of an industrial plant [38]. Also, it is not rare that an industrial device uses vendor passwords or keeps all ports open by default. Such behaviors are not accepted in IT security, while they can be met in OT.

1.3.6 Security attributes

Most differences between cyber security risk in a corporation network and an IACS ensue from the nature of feared consequences, which in the case of industrial infrastructures

may be not only financial or reputational but also related to health, safety and environment [68]. This causes a reversion of security objectives priority in IACS compared to traditional information technology systems [131], [68]:

- **Availability (A)** is the primary goal: this is the ability of an asset to be accessible to any authorized entity and to fulfill its functions at the required time (time or operations sequence) and in the required time slot. In a nutshell, it is the system readiness for proper service. This security property is particularly important in real time control systems such as protection systems, which are in charge with safety and security of persons and infrastructures (e.g. electrical protection mechanisms).
- **Integrity (I)** is the property of non alteration or destruction of data, either maliciously or accidentally. The data received by a sink must be identical to the data emitted by the source. If integrity of process data such as measurements or command signals is compromised, it may lead to a wrong system state awareness at the control entities, which may issue inappropriate or even dangerous controls.
- **Confidentiality (C)** is generally granted less priority: Confidentiality is the absence of unauthorized disclosure of information. In IACS, this security attribute may apply to stored passwords and encryption keys, process and infrastructure state information, command signals (plant recipes).

However, exceptions to the Availability-Integrity-Confidentiality paradigm may happen according to the purpose and security policy of the considered plant (for instance for intellectual property questions).

IEC 62351 standard adds the security requirement of **non-repudiation or accountability** that consists in *"preventing the denial of an action that took place or the claim of an action that did not take place."* [66]. This security attribute is particularly meaningful for forensics analysis after an event has occured to trace the whole scenario back to its causes. For ENISA, what truly rules the security in IACS environments is the Safety-Reliability-Availability model [35]. This view highlights the double meaning of availability when applied to IACS: it relates both to information data security and systems dependability, and both are important to IACS cyber security. **Safety** is the guarantee of non-catastrophic consequences for the environment (infrastructure, persons and, to some extent, society) [12]. **Reliability** is the double ability of a system to refrain from operation when it is not expected to operate and to ensure continuity of proper service [62]. While the first part of this definition relates to security, the second one is about the operational goal of dependability. This conflicting factors shall find a balance when dealing with cyber security measures implementation, especially for SAS [62], as for instance monitoring of safety measures may help prevent some attacks in electrical substations [36].

Although the debate is still ongoing about cyber security objectives in IACS, there is a consensus on dependability and security strong interconnections. Despite the well-recognized high impact that safety and security have on each other in IACS, the absence

of the safety-security link in numerous institutional publications is conspicuous [92].

1.4 Intrusion detection in industrial environments

1.4.1 Intrusion detection: concepts

Firewalls and Intrusion Detection Systems (IDS) are typical security measures continuously monitoring network traffic to judge if a packet is legitimate or not. IDSs and firewalls are not to be confused. While the latter limit access at the entry point of a network segment to prevent intrusion, IDSs check for an attack from inside it and generate alarms. Thus IDSs are able to detect intrusions originated from outside as well as inside the system network perimeter. Another difference, maybe even more significant, is that firewalls analyze a single message content, while IDS can perform analysis of both a single message and sequences of messages, thus introducing some correlation.

The definition of intrusion detection given in the introductory part of the IEC 62443 standard about IACS cyber security is: *"security service that monitors and analyzes system events for the purpose of finding, and providing real-time or near real-time warning of attempts to access system resources in an unauthorized manner"* [68]. The "system events" mentioned are the data to be analyzed and can originate from devices or from network traffic. Thus, intrusion detection systems can be **host-based** (HIDS – Host Intrusion Detection System) as a software module analyzing relevant internal data of the host devices, or **network-based** (NIDS – Network Intrusion Detection System) running as independent devices connected to strategic nodes of the network and checking traffic. Regarding detection methods, there exist two approaches: **signature-based detection**, **pattern-based detection** or **blacklisting** that checks the collected data looking for patterns characteristic of a specific attack, and **anomaly detection**, **behavior-based detection**, **model-based detection** or **whitelisting** to detect deviation from an acceptable system behavior.

Blacklisting vs whitelisting

An IDS using blacklisting relies on a database of signatures of attacks. If gathered data do not match any of this attack patterns, they are considered as normal and no alarm is generated. Such IDS generally produce few false positives as attacks are precisely described. This is true under the condition of meaningful signatures, not too tight but not too loose. Creating such rules is a complicated task. In addition, to be fully efficient, such IDS require to continuously maintain signatures databases up to date. The main weakness of signature-based IDS is their inability to detect unknown attacks.

While blacklisting consists of characterizing abnormality, whitelisting focuses on sys-

tem normal behavior. The idea is to specify this normal behavior and detect deviations from this model. Normality can be learned: the system is observed while running and machine learning algorithms extract behavioral patterns from the collected data, hold as trustworthy. Or it can rely on specifications that formally describe the system behavior. The true strength of this method is its ability to detect zero-day attacks. However, as it is complicated to exhaustively characterize the system normal behavior or as it may derive over time, whitelisting IDS typically have high rates of false positives. A criticism about the learning approach is that collected data are assumed trustworthy. If attacks actually occured during the learning phases, then false negatives may happen.

Specificities of OT systems (see section 1.3) bring some challenges when considering IDS implementation in industrial environments. False negatives are not tolerable in industrial systems, especially in Critical Infrastructures (CI), such as the power grid, because of the safety implications of security hazards, as discussed in section 1.3. And false alarms (false positives) may impact continuity and quality of service, which are of the highest priority in CIs. The cyber threat hanging over industrial infrastructures is recent compared to the IT world, and awareness of it is even more recent. Attackers and targets are still "discovering the field". Moreover, IACS typical diversity of technologies means diversity of possible attacks, which makes them difficult to characterize. System specifications exist (e.g. as process recipes, configuration files, etc.) as configuration and design are often done on engineering stations and then uploaded to the devices. They may be used and enhanced to develop IDS. Hence, the general lack of information about IACS attacks, the necessity of detecting zero-day attacks and the existence of generally well-defined system configuration specifications favor the choice of a behavior-based rather than signature-based IACS-oriented IDS.

Host-based vs network-based intrusion detection

Host-based IDS, as its name implies, is part of the host device. It thus can access internal data, from historical logs or traffic emitted and received. A good thing with HIDS is that the data manipulated are not too diverse. But the counterpoint is a limited scope. Moreover host resources (computational power and memory) are impacted by intrusion detection programs. This must be taken into consideration to make sure the primary mission of the device will not be affected.

A NIDS is set up as an autonomous device connected to the communication network in order to get the whole traffic of the system portion covered. It sniffes and analyzes communications without disturbing the rest of the system, it is a passive security measure (as discussed in section 1.2.2). NIDS performances at analyzing messages payloads are classically challenged by high network throughputs. IACS specificities such as hard real-time requirements may also give rise to further difficulties.

Limited resources of IACS and rather well defined communication protocols and pro-

files are arguments in favor of NIDS.

1.4.1.1 Performance metrics

Several metrics exist to help cast performance of an IDS algorithm that come from Classification domain [39]:

- *Number of True Positive (TP)*: is the number of true abnormal events / intrusions detected.
- *Number of True Negative (TN)*: is the number of true normal events / non intrusion detected.
- *Number of False Positive (FP)*: is the number of normal events classified as abnormal ones. They are false alarms.
- *Number of False Negative (FN)*: is the number of abnormal events classified as normal ones. They are undetected intrusions.
- *True Positive Rate (TPR)* (also called *recall* or *hit rate*): is the fraction of abnormal events that are retrieved. It is estimated as $TPR = TP/P = TP/(TP + FN)$, P being the total number of positives.
- *False Positive Rate (FPR)* (or *false alarm rate)*: is estimated as $FPR = FP/N = FP/(FP + TN)$, N being the total number of negatives.
- *Precision*: is the fraction of instances classified as true that are relevant. It is estimated as $Precision = TP/(TP + FP)$.
- *Accuracy*: is the percentage of correct classifications relatively to the total number of classifications. It is estimated as the ratio of total true detections over total detections $Accuracy = (TP+TN)/(P+N) = (TP+TN)/(TP+TN+FP+FN)$, P being the total number of positives.
- *F-score* or *F-measure*: is the harmonic mean of precision and recall $F-measure = 2 * ((Precision * Recall)/(Precision + Recall))$

Along with these accuracy metrics, others help evaluate IDS algorithm and implementation efficiency. The ones met in the literature include:

- Analysis *throughput* (in bits per second – bps or packets per second – pps): helps estimate how much data can the IDS analyze without impacting its accuracy.
- Classification decision *velocity*.
- CPU usage.

1.4.2 A taxonomy of IACS-oriented IDS

As already highlighted, industrial automation and control systems are cyber-physical systems characterized by multiple dimensions and layers. An intrusion detection system dedicated to industrial environments should consider this intrinsic diversity and heterogeneity to ensure relevance and link a deviation of the physical process to its cause at the supervision and control level. Koucham [94] proposes a taxonomy of intrusion detection approaches dedicated to IACS reflecting this diversity of views, shown in Table 1.2. His taxonomy merges two classifications from literature: (i) Zhang's [151] three-dimension division of cyber-physical systems, which encompasses computational (hardware and software) and physical components closely collaborating through communication, and (ii) Zhou et al.'s [152] distinction of the physical process from the industrial control system.

At the highest level, Koucham [94] considers two classes: *communication* and *intelligent nodes*. They correspond to the traditional distinction between network-based and host-based IDS. They also correspond to *transmission* and *processing* of data, respectively. The *communication* part includes *vocabulary* of protocols (protocol syntax and semantics), *grammar* of exchanges (communication scheduling, that is sequences of messages and timing), *structure* of exchanges (communication paths between entities or between zones), and network *telemetry* (meta-information about exchanged data such as transmission time or packets size...). The axis *intelligent nodes* is concerned with all devices having a computing role in the industrial process and are made of microprocessor, memory, communication interface and other peripherals. Features to consider are *ressources*, in terms of communication, memory and processing capabilities, and also for real-time nodes *tasks* scheduling, state and execution.

Perpendicularly to the *communication-nodes* view, Koucham [94] identifies the degree of the IDS awareness of the interaction between IACS and the physical process. Here we are interested in *control logics* performed by controllers and *control data*, that is all measurements, status information and commands exchanged between, sensors, actuators supervision, controllers... These data allow controllers to estimate the system state.

Table 1.2: Classification of IACS-oriented intrusion detection approaches [94]

Physical process awareness	**Communication**	**Intelligent Nodes**
Low	Protocols vocabulary, Grammar, Structure of exchanges, Network telemetry	Ressources, Tasks
High	Control logics and data	

Any IDS implement one or several approaches. In this dissertation, we are concerned with anomaly-based intrusion detection in communication. Henceforth, we center the

following state of the art mainly on communication approaches. Some of the presented work may also take advantage of some intelligent node aspects.

1.4.3 A state of the art of IACS-oriented anomaly-based NIDS

1.4.3.1 Vocabulary and grammar approach

Among the works about communication-oriented intrusion detection for IACS, many make use of the protocols vocabulary and grammar. Thus, rules on the messages syntax and protocol-specific semantics can be extracted, e.g. structure of messages, values of certain protocol-related fields or dependencies between fields of a single message or several messages.

Cheung et al. [27] were among the first to show and use simplicity of an industrial protocol to propose a tailored three-level IDS. Protocol level makes use of the considered protocol specificities, Modbus/TCP, to check whether packets on the wire are compliant with Modbus standard. Network level verifies communication patterns that authors characterized using the considered network segmentation and access policies. Application level is concerned with services availability and is very specific to the industrial application under study. The first two levels are implemented by specifying rules for a rule-based open source NIDS software, Snort, and specifying a formal model of the normal Modbus communication behavior. The third level is learning-based. Combining specification-based and probabilistic methods allows to mitigate their weaknesses while taking advantage of their strengths.

Lin et al. [102] also use an open source tool to develop a NIDS dedicated to the DNP3 protocol: they have integrated a DNP3 parser to Bro, an open source runtime network traffic analyzer. The authors define Bro rules to detect violations regarding DNP3 protocol. Syntax is covered as packets structure is checked, and semantics as well: DNP3 protocol define dependencies between some fields of a single packet and dependencies between many packets and authors use intra- and inter-packet inspection to ensure such requirements are observed.

A third similar open-source tool is used by Diallo and Feuillet [32] to implement their NIDS for Modbus/TCP traffic: Surricata, a real-time, multi-thread network threat detection engine developed by French security engineers. The authors present pros and contras of two possible deployment strategies of a distributed NIDS for IACS. In the case of centralized processing, IDS modules sniff network traffic and do basic preprocessing operations before sending it to the server performing the detection. Pros are the possible correlation between nodes and simple modules with easy maintenance. Weakness is that traffic is almost doubled, thus compromising scalability. Arguments in favor of a decen-

tralized IDS, where each node performs analysis locally are that little traffic is generated (if rules are accurate) and scalability is inherent. Correlation is less flexible, correlation of alerts is still possible though. Whatever the choice, modules must be installed at critical points of the network in order to collect the whole traffic, they shall have a dedicated network to disturb the industrial system as least as possible, and alerts must be transferred to operators through a supervisor. The proposed anomaly IDS makes use of Modbus protocol specifications and industrial system specificities to define Surricata rules. Features of interest include authorized communication paths, authorized function codes for each client/server couple given operating mode (normal, diagnostic, maintenance...), and for each function, authorized values ranges and readable and/or writable addresses. Values of a single packet's fields and relation between them are extracted and analyzed. The authors evaluate the performance of their approach. They estimate that one box is required for ten communication couples if a safety IACS is involved that needs message transfer time of 4ms i.e. a throughput of 500 messages per minute. In a more realistic scenario with about 100 rules and few alerts, one box may cover twenty couples.

Wu et al. [141] also adopt a distributed architecture. They want to replace a traditional perimeter firewall at the periphery of a substation by what they call a distributed firewall, namely a distribution of IPS (Intrusion Prevention Systems). Concretely, they propose to place rule-based modules at different locations of the substation thus covering internal attacks too. Each module is dedicated to the particular protocol(s) that flow over the covered links, all Ethernet-based. Main difference with the off-the-shelf perimeter firewall to be replaced resides in the Linux modules' capabilities to analyze packets down to the application layer and make use of the payload, while the initial firewall analyzed the packets only at the transport and network layers. Hence, it can be checked if one source is authorized to send specific function codes. Monitored features for implementation example of a DNP3 module are similar as for the perimeter firewall (destination MAC address, source and destination IP addresses, protocol identification, destination port) and function code. All packets are denied by default and have to meet the rules to gain access to the protected area. This active approach (preventing some packets from being transferred) is unusual in the context of electrical substations as anything that might compromised safety (e.g. protection mechanisms) is considered cautiously and generally avoided. Authors let investigation on the latency introduced by their implementation for future work.

Morris et al. [107] propose to translate industrial wireless protocols Modbus RTU/ASCII traffic into Modbus TCP traffic so it can be analyzed by Quickdraw tool by Digital Bond. Quickdraw is a preprocessor and set of Snort rules (14 for Modbus/TCP) for IACS running Modbus/TCP, DNP3 and Ether/IP protocols. Existing rules include: (i) checking packet length, (ii) detection of malformed packet by checking a specific Modbus/TCP field (not used for Modbus RTU/ASCII as this field is not used), (iii) detection of excessive busy and acknowledge exception code responses, (iv) detection of scanning of function codes and addresses. Further rules identified by authors are (v) checking traffic

volume in a given time frame, (vi) detection of multiple Modbus responses with same function code and address, (vii) detection of values that would provoke exceptions that stop the program (e.g. division by zero), (viii) detection of irrelevant values: measurements outside acceptable ranges or command values that would put the system in an inconsistent or unsafe state, (ix) detection of unsupported addresses, function codes and combinations of read/write adresses and size of data to be read or written. The attacks hence addressed are DoS (covered by rules i, ii, iii, vii), response injection (i, vi, viii), command injection (viii, ix) and reconnaissance (iv). Two configurations are possible: passive (alerts are written in logs for further analysis) or active (suspect transmissions are blocked). Latency added by the intrusion prevention configuration is 3.5ms per byte for a throughput of 9600 bits per second, that is 903ms for maximum length Modbus RTU packets.

1.4.3.2 Structure

The assumption that communication flows in IACS are strongly periodic [13] is key to the NIDS proposed by Barbosa et al. [15]. This is mainly due to polling mechanisms allowing master devices to retrieve data from field devices, that is master-slave exchanges. Cycle information of traffic flows hence typify expected communications behaviors in the proposed anomaly detector. Industrial control protocols under consideration in this work are Modbus/TCP and MMS (the protocol used in the IEC 61850 stack for server/client communication), both using TCP as transport layer. Two types of TCP connections are identified: long- and short-lived connections. Periodicity is on sent requests for the former and on the connection establishments and terminations for the latter. TCP features characterizing short connections (i.e. of duration below one second) are server address, IP protocol, server port and client address. For the long connections, they are the same plus client port. The learned model by the proposed tool, PeriodAnalyser, is a whitelist of valid commands along with their emission frequencies. MMS traffic, typical of decentralized control systems, shows more flow variations than Modbus/TCP. But Barbosa et al. [15] think that the approach is still accurate for such protocols as there are a lot of flows comprising periodic requests. According to the authors' experience of real world CIs, communication configuration information is either not readily available or incomplete. This explains the author's choice for machine learning approach. This traffic modeling approach address the limitation of previous work, telemetry-oriented, by the authors [14], in which they investigated the use of signal processing techniques such as discrete Fourier transforms and autocorrelation functions. Periodic activities were detected using number of packets or bytes sent per time interval, with no notion of semantics, thus providing little insight into which packets caused the periodic behavior.

Hadeli et al. [46] propose an anomaly-based intrusion detection system leveraging deterministic characteristics of IACS communication system. The rules are written using information from system description files, both explicitly specified and implicitly extracted

thanks to protocols specifications and experts' knowledge. Features explicitly given in IACS configuration files are the number and identification of all communicating entities, subnetwork characteristics, IP addresses, protocols used by each entity, communication paths, flows time patterns... Features inferred from the files content concern other protocols (for configuration, time synchronization or typical of the kind of considered system), ports... The authors give the examples of a setup packet file from process automation field and of a substation configuration description file from IEC 61850 power utility automation area [45]. The open source NIDS software Snort is used for implementation.

According to Shang et al. [124], identification of communication patterns through machine learning is an appropriate approach to detect anomalies involving more than one network packet. They propose an anomaly IDS that uses one-class SVM (Support Vector Machine) to model normal communication outlines. One-class SVM technique computes a hyper-plane in the feature space separating acceptable objects from anomalous objects. Extraction of feature vectors is done with a stochastic optimization algorithm (particle swarm optimization), which gives better detection accuracy and time performance than traditional grid parameter optimization. The authors conclude that their approach requires fewer support vectors, is more concise and is more likely to be generalized than traditional one-class classification. However, they apply this NIDS technique to Modbus/TCP communication of a client/server couple using only sequences of function codes to compute support vectors. Modbus/TCP is known to be a relatively simple communication protocol used for rather static exchange patterns. Would the approach of Shang et al. still be accurate, computationally efficient and scalable for more complex ICS communications?

1.4.3.3 Telemetry

Linda et al. [103] also want to exploit regular and stationary patterns displayed by the communication. The proposed IDS is based on a neural network, a supervised machine learning technique. The features extraction technique uses a fixed-length window shifted over the stream generating a feature vector, thus capturing the time series nature of packet streams. First step of learning phase is the random generation of a set of simulated intrusion vectors uniformly distributed over the window based attribute space. Second step is the training of the neural network with both normal and abnormal vectors. The considered network traffic attributes set include: number of IP addresses, maximum and minimum number of packets per single IP, average interval between packets, time length of the whole window, number of protocols, maximum and minimum number of packets per protocol, number of flag codes, number of packets with 0 window size, number of packets with 0 data length, average data length. Selected attributes, window size and neural network structure impact the IDS performances and are fixed empirically. Evaluation set up is composed of a control PC and a single PLC controlling valves, so the

traffic under consideration is indeed very simple with high regularities and a limited set of data. Further experimentation with more complex systems would be necessary.

The NIDS proposed by Yang et al. [143] uses pattern matching, characterizing normal traffic profiles from system indicators, such as link utilization, CPU usage, and login failure. Identified profiles are attached to specific operational times (day of week, week-end, holiday and so on). Monitored features are not limited to network traffic. Indeed, sources of data are existing server audit logs giving I/O flows and hardware working statistics, and SNMP (Simple Network Management Protocol) for network traffic statistics. Profiles are learned with a nonparametric, empirical modeling technique (AAKR - autoassociative kernel regression) that uses trusted observations to make predictions for new ones. Residuals obtained by comparing the observations with the model predictions allows detection of anomalous activity. Demonstration is shown on a DoS attack.

1.4.3.4 Process knowledge

In approaches making use of process and control system behavior, managed features exceed traditional security features. This echoes the strong assumption that an attacker targeting an industrial infrastructure must impact process and communication system's variables to cause damages. And therefore, attacks can be revealed by anomalies in the system behavior.

A way to evidence diverging system behavior due to intrusion is inspired by Fault Detection research field where measurable variables are monitored to estimate if the system is converging towards some critical state. Such strategies as the critical state-based approach by Fovino et al. [42], [43] and Carcano et al. [18], and the state-space models by Manandhar et al. [105] or by Singh et al. [128] require that a model of the system is available in order to compute expected upcoming state of the system and compare it with the actual measured one.

The underlying idea of the work by Fovino et al. [42], [43] and Carcano et al. [18] is that an attacker has to modify the system state in order to cause damages. The authors propose a firewall, which, while searching analyzed packets for known signatures, also updates an image of the physical system state according to the packet content. If the received packet does not put the system into a critical state, it is forwarded to its destination. The proposed detection mechanism relies on an *a priori* knowledge including the system architecture, the meaning of SCADA commands and the set of critical states. The digital representation of the system is maintained as close to the real system state using both the content of the packets flows analyzed and a master emulator that queries directly the PLCs. Every packet is characterized by a tuple consisting of its source and destination IP addresses, its source and destination ports, and specific protocol fields.

Fovino et al. [43] specify such fields for DNP3 and Modbus/TCP protocols. For instance, Modbus special fields are function code and function parameters (payload). Distance of the current state vector with critical states vectors is the criterion to block the corresponding packet and launch an alarm. The authors think that this critical state-based detection technique allows to distinguish complex attacks making use of a sequence of legitimate commands, whose individual analysis does not reveal any malicious activity.

Manandhar et al. [105] also base their approach on the similar assumption that any fault or attack directly induce changes on either voltage, current or phase variables. A state-space model of the voltage flows is used. The authors estimate the upcoming voltage value from measurements periodically fed to the central controller from sensors and meters deployed at different locations of the power system. Statistics on the residue of Kalman equations (χ^2 test) and the Euclidean distance between actual state value and the predicted one are calculated to detect attacks such as DoS (here, missing sensor data) or false data injection, but may also reveal faults. According to the authors, one of the advantages of their approach is that the χ^2 test takes into consideration all integrated effects since system start time. Thus the detector is resilient to sensors soft failures such as instrument bias shift. We wonder if this method would be efficient to detect advanced persistent threats (APT) such as Stuxnet, that cause slow long-term and subtile changes in control. Also this model needs the load profile to be constant or at least known so the voltage change can be predicted or it would be interpreted as a fault or an attack. This approach cannot discriminate a fault from an attack per se and further analysis is required to classify the alarm as malicious or genuine.

Singh et al. also use system state variables centralized by the master terminal unit (MTU) at the station level as input to their intrusion detection module. The grid architecture is modeled with PSAT (Power System Analysis Toolbox), an open source MATLAB and GNU/Octave-based software package for analysis and design of electric power systems. At every step, PSAT configuration file is updated with the dependent variables measured at the process level, that is voltage magnitude, phasor angles, active power and reactive power. The simulator performs a power flow analysis and then the resulting ideal system state is compared with the one extracted from independent variables, the measured load or control commands. Deviation between simulated and real system state helps detect any anomaly. Attacks that can be detected include a compromised remote terminal unit, either issuing wrong data to the MTU or modifying command values to actuators.

The purpose of Borges Hink et al. [49] is to compare various machine learning methods to detect cyber-attacks in power systems. The authors' framework count four phasor measurement units providing real-time power system metrics such as voltages and currents but also the status of system devices including relays, breakers, switches and transformers. Logs from control panel, Snort and relays provide further data for a total of 128 features.

Evaluation of methods show that a set of fourty relevant features is enough for accurate results, thus reducing problem dimension. Among obstacles to deployment to industry, the authors identify the non existence of processes for acquiring and maintaining in-situ training data and for learning system feedback and retraining criteria. Further research is needed.

Unlike the aforementioned works, Almalawi et al. [3] propose to infer acceptable and inadequate states from collected data thanks to an unsupervised machine learning approach. Thus, no *a priori* knowledge about the system model is required. First step of the IDS development is the identification of consistent and inconsistent states from unlabelled data based on two assumptions: (i) consistent SCADA data entries significantly exceeds inconsistent ones, and (ii) consistent and inconsistent SCADA data must be statistically different. It seems to us that a third assumption would hold: the training data set must be consistently (if not exhaustively) representative of all possible consistent states. Second step consists in extracting proximity-based detection rules for both consistent and inconsistent observations (detection rules are actually micro-clusters centroïds). The two aforementioned phases are performed off-line so there is not much concern about performance. Online inconsistency detection phase is linear with the memory size of the detection rules, in terms of computational complexity. Experimental evaluation is made on a water distribution system simulation with a virtual SCADA using Modbus/TCP protocol. The manipulated process parameters are water flow, pressure, demand, level, valve status and setting, pump status and speed. The proposed approach automatically identifies about 90% of consistent and inconsistent states. In our opinion, this is not sufficient for CI. That means that experts involvement cannot be avoided, which was an argument in favor of such an unsupervised IDS. Moreover, it seems to us that states identified as inconsistent may also be rare but legitimate states and experts' analysis is require to clear up any doubt.

Another trend is to consider sequences of observations and not only isolated features. Works by Caselli et al. [19] and Yoon and Ciocarlie [150] analyze sequences of messages while Pan et al. [115] or Skopik et al. [129] are interesting in sequences of system states.

Semantic attacks make use of the knowledge of the processes and not only the employed systems to cause damages. Caselli et al. [19] address sequence attack, a specific semantic attack, which modifies sequences of authorized ICS operations, either their order or their timing arrrangement. Individually, each event is legitimate but the alterated sequence of them causes damages. As the author want to propose an approach that does not rely on an accurate system configuration like specification-based detection, they describe ICS behavior with discrete-time Markov chains. Time-ordered sequences of events with transitions probabilities are learned from historical network communication, log entries, process variables streams. The modeling process aggregates in a single state events sharing similar semantic meaning. Transitions between states indicate order relation between them. During detection phase, Markov chains are computed (states identification and

transition probabilities calculation) from captured information along with their weighted distance with the learned reference models (either of normal or deviant behavior). Events composing sequences consist of tuples of relevant fields and features for network packets, of tuples of log attributes identifying an operation, of the value of the considered process variable or a tuple of values of linked variables. Targeted anomalies are unknown states (event's attributes have incorrect values), unknown transition (order-based sequence attack), unknown transition probability (time-based sequence attack). Authors recommend to ignore human activities as their high time variability makes them unproper for a probabilistic approach. As models are built progressively with data reading, for online detection it is necessary to wait for a comparable amount of events as there were in learning stage to get some convergence between learned chains and detection chains. Many false positives occure because of unknown transitions due to network delay. To improve detection, the authors introduce the notion of event's importance: some events are more critical than others and would more likely be exploited in attacks, thus changes on events tagged as important will trigger alarms for more stringent threshold than event tagged as unimportant. They also suggest to make use of further semantic analysis that helps to discern a suspicious but close to a valid event from a malicious event, and of stable time patterns to perform detection only on transitions with low time variability typical of automated behaviors of control process while dismissing events with high time variability related to human activities.

The IDS developed by Yoon and Ciocarlie [150] makes use of the following assumptions: predictable IACS behavior, fixed network topologies, simple protocols, regular communication patterns (on content and exchange structure). Attacks would thus exhibit diverging communication patterns. Command and data sequences are modeled by a Dynamic Bayesian Network (DBN) whose underlying probabilities are extracted using incremental (on-line learning) Probabilistic Suffix Tree (PST). PST learns a set of subsequences of different lengths, thus being resilient to some noise. The probability of a message occurence depends only on the message history of the considered connection. To address the problem of legitimate variations from the base pattern due to missing, out-of-order messages and/or sporadic tasks, the authors propose to infer missing data. It allows to significantly reduce the false positive rate, at least for high regularity of patterns in the monitored sequences: more randomness in the sequence under monitoring may instead increase false positive rate.

IDS proposed by Pan et al. [115] uses process features: phase current magnitude measured at each relay, relay status, logs of trip commands on the wire (captured by Snort), and control panel remote trip status. Signatures of genuine and malicious scenarios are learned from logs of training sessions using data mining. Signatures are temporal sequences of system states. Accuracy is evaluated with tenfold cross-validation on the 25-scenario set, 21 randomly chosen being used for training. In average three out of four zero-day attacks are detected. Misclassification is mainly due to resemblance of some attack scenarios with genuine scenarios, thus resulting in very close paths that share com-

mon subsequences. According to the authors, one of the advantages of such a path-mining IDS is its ability to process data as a stream rather than collecting them for offline analysis. Real-time classification from live system input as a track is let for future work, though.

Skopik et al. [129] address Advanced Persistent Threats (APT), carried by organizations with financial capabilities and skills, and determined to cause damages to a specific target. Such attacks may use numerous zero-day attacks, thus explaining the choice of anomaly-based IDS, and may infiltrate systems without exploiting technical weaknesses thanks to social engineering, what justifies the system behavior-based approach (process and IACS). The introduced anomaly IDS is based on self-learning approach, that is a machine learning method that executes both learning and detection phases in parallel by continuously creating hypotheses about events correlation and validating or refuting them online. This specificity endows the proposed IDS with the ability to adapt to system evolutions. Log files are collected from distributed sources (e.g. firewall, switch between corporate LAN and IACS network, SCADA...) and aggregated in a single file while keeping temporal order. Vectorization is used to compact input data into feature vectors and thus reduce computation time. These vectors may be augmented with contextual information (e.g. maintenance operation) from additional sources (not feeding the IDS). Patterns of log entities are extracted and periodically revised given their occurence to infer events that originated the considered log entries. To make sure that exceptional genuine events are integrated into the system model, the authors introduce the configuration parameter "price of a pattern". The price of a pattern is adjusted according to the log entries periodicity: rare log entries are cheaper than frequent ones. In other words, fewer rare log entries are required to buy/ratify a new pattern than frequent log entries. Hypotheses about events sequences, regarding both order and timing, are periodically created and evaluated during a certain time slot. Validated hypotheses integrate the system model. Statistical analysis infers the degree of deviation with these hypotheses and therefore with the system's normal behavior. Probability of an anomaly for a given hypothesis is tracked over short and long time periods to both detect sudden anomalies and increase alarm confidence.

1.4.3.5 Multiple appproaches

As highligted by Table 1.3, most of the works presented in this state of the art actually cover several of the approaches identified in the taxonomy of section 1.4.2.

Berthier and Sanders [16] present specification-based intrusion detection sensors for deployment in advanced metering infrastructures (AMI). An AMI is a communication infrastructure enabling information exchange between meters and utilities. Authors reject signature-based detection because of lack of information about AMI attacks and the need of detecting zero-day attacks. The necessities to rapidly understand the root causes of

attacks and to detect stealthy threats convince them to turn towards specification-based instead of probabilistic detection. Berthier and Sanders are aware of the two main limitations of specification-based detection. First, the development of specifications is expensive and tedious but the tight control over communication protocols authorized in AMI and the homogeneous behavior of meters and metering traffic enables to reduce this cost. Second, specifications are often very difficult to evaluate and verify: the authors leverage a formal verification framework combined with experiments to evaluate their IDS. The authors develop security requirements covering three constraints categories: at network, device and application levels, and all or part of the five following constraints types: (i) data: valid range of values, (ii) access: defines which program or user is allowed to access which objects, (iii) timing: interactions of processes and shared resources, (iv) resource usage: memory or network resources that a service is supposed to use, and (v) operational constraints: expected behavior of a program. The proposed security constraints are based on a threats model, an analysis of the communication protocol specifications and the expected behavior of meters, and historical training network traces.

Like Lin et al. [102], Parvania et al. [116] use Bro to implement their anomaly NIDS. Their purpose is to check Modbus/TCP communication in the fault location and isolation process of a power distribution system. But unlike Lin et al., the strategy adopted by the authors is to generate not only rules for protocol compliance, but also system communication configuration and mechanisms. Monitored criteria are IP addresses, valid defined Modbus TCP commands, master-slave sequential communication patterns, communication periodicity. The authors present a further version of this so-called hybrid control NIDS in an article by Koutsandria et al. [95]. They add to communication rules the checking of physical limits of the system, namely the consistency of the currents and voltages with electrical conservation laws and the status of circuit breakers. This comparison of measured electrical values with devices status helps detect complex attacks consisting in series of legitimate commands who collectively cause damages.

Yang et al. [146] tackles the problem of digital substation cyber security. The proposed SCADA IDS checks compliance of attributes both related with traffic structure and process model with rules set by experts from system and protocols specifications. The IDS includes three layers of verification: (i) Authorized sources and destinations at the Ethernet link layer (MAC adresses), the network layer (IP addresses) and the transport layer (ports), by themselves and combinations of them. (ii) Protocol-based detection looks at OSI model layers 2 to 7 (deep packet inspection) to identify protocols using regular expression patterns matching. Substation typical protocols include Modbus, DNP3, IEC 61870-5 series, ICCP, IEC 61850, some proprietary protocols. (iii) Behavior-based rules cover both single-packet and multiple-packet criteria. Cross-packet inspection uses an integrated database and is concerned with time-structure of the packets flows (time interval or frequency of specific packets or commands), correlation between switches states and relevant measured values, and correlation between protection functions activated by a relay and the relevant measurement data. The single-packet inspection checks that the

actual payload length corresponds to the length field value, the function code belongs to the authorized set, and the measured values fit into their expected operational range. The IDS was implemented in C/C++ as a plugin for an internal tool, Internet traffic and content analysis ITACA, a software platform dedicated to traffic sniffing and real-time IP network analysis. Extreme execution time is estimated at 254μs, which is less than the high-speed protection information data delivery time requirements given in IEEE standards for electric power substation automation, that is less than 1/4 cycle or 5ms at 50Hz [59]. Evaluation is done for a man-in-the-middle ARP spoofing attack launched from Metasploit software tool.

Yang et al. extend their approach to intrusion detection in IEC 61850 electrical power automation systems in [148] and further developed in [147]. The proposed anomaly IDS proposed is very close to our work. In this evolved version of their IDS, the authors divide the third level of verification into two distinct levels, thus resulting into a four-layer detection model: (iii) Anomaly behavior detection is concerned at the supervision level with the number of instantiated report control blocks, number of connexion requests to clients, source of settings modification, unicity of file transfer, port used for SNTP traffic, time constraints of critical commands. At the process level, IDS input features include GOOSE and SV protocols specificities as given by the standard and the system configuration. Telemetry criteria learned from practical captured traffic are also checked, including packet transfer rate per second, transfer byte size per second, length of packets, size of packets. (iv) Multi-parameter-based detection compares the data carried by process level protocols (critical commands and measurements) and MMS protocol used for reporting activity to substation level. It also checks that analog signals fit authorized ranges.

Kwon et al. [97] estimate that statistical approach is the most relevant for intrusion detection in IEC 61850 automation systems because of the large amount of data to handle. Features verified by the proposed NIDS include network telemetry metrics and protocols metrics. Classical network metrics are rates of bits per second (bps), packets per second (pps) and connections per second (cps). Metrics for IEC 61850 protocols are most recent GOOSE message timestamp, GOOSE message frequency, counter of received GOOSE messages, and MMS command type (either A1 - confirmed response or A3 - unconfirmed report). Learning data set is a one-week real traffic of a Korean substation. Evaluation is performed in a Korean smart grid test bed. To our understanding the evaluation is limited to the detection algorithm accuracy, though, using 261 five-minute-long traces of normal traffic and 27 created pcap files for most probable IEC 61850 attack scenarios. Statistical approach may be relevant for network metrics such as bps, pps and cps. However GOOSE and MMS features examined are deterministic and defined by protocols specifications and in system configuration files. We express reservations regarding validity of the whole model such as presented in this article because authors seem to not consider legitimate extraordinary events, such as electrical fault related messages, impacting GOOSE retransmission sequences and thus bps and pps rates. Also, it is not clear how the number of connections per second is evaluated.

Ten et al. [132] propose a model for an anomaly intrusion detection system dedicated to power infrastructure automation. They identify data sources to exploit as relay settings, user credentials and application logs, traffic logs, and status of running applications. Temporal correlation may allow detection of local malicious actions such as modification of relay settings, while spatial correlation may help detecting sophisticated attacks resulting of joint actions at multiple substations and control centers. Such a model is valid at a global level (SIEM) and would probably mean off-line or at least non real-time detection.

Based on this preliminary work, the authors then have proposed a blacklisting HIDS [133] and whitelisting NIDS [51]. The IDS presented in [52] by J. Hong et al. merges the findings from these previous works for IEC 61850 substation automation systems (SAS). The whitelisting specification-based NIDS checks for compliance of IEC 61850 broadcast messages with predefined rules. These rules cover compliance of communication with protocols requirements (packet structure, coherent values between fields of a single message or of many messages) and with network metrics such as rates of messages on the wire. Host-based intrusion detection is based on the assumption that system and security logs exist in the SAS devices and applications, namely user interfaces, IEDs and firewalls. Intruders' footprints can be found, such as a wrong password attempt flag, that define events sequences revealing an attack.

1.4.3.6 Intrusion detection in power system

As already written, the smart grid is one of the critical infrastructures that raise the greatest concern regarding cyber threats and thus drives a lot of research efforts. Some of the works mentioned in this section echoe this interest as the application domain is the power system: Lin et al. [102] (vocabulary and grammar), Fovino et al. [42], [43] and Carcano et al. [18], Manandhar et al. [105], Singh et al. [128], Borges Hink et al. [49], Pan et al. [115] (process), Y. Yang et al. [146] (multiple).

Four other papers specifically tackle intrusion detection in power systems implementing IEC 61850 systems and networks: Wu at al. [141] (vocabulary approach for GOOSE protocol among others), Hadeli et al. [46], [45] (structure approach for MMS, GOOSE and SNTP protocols), J. Hong et al. [52] (multiple approach exploiting vocabulary, grammar and structure for IEC 61850 broadcast protocols GOOSE and SV), and Y. Yang et al. [148], [147] (multiple approach exploiting all communication aspects and process knowledge for IEC 61850 supervision and process buses protocols).

It is interesting to mention the work of Premaratne et al. [120] as it is one of the first to tackle the question of intrusion detection in IEC 61850 substations (published in 2010). However, unlike the other works summarized in this state of the art, the authors make the choice of a blacklisting NIDS as they consider it the most effective one because of the impossibility of dealing with all genuine exceptions. Snort rules are derived from

data obtained through simulating attacks, such as a denial-of-service (DoS) attack, password cracking, and address resolution protocol (ARP) spoofing. Although the studied application is IEC 61850 substation automation system, authors do not consider any of the three protocols introduced by this standard but focus on ARP, Internet control message protocol (ICMP), hypertext transfer protocol (HTTP), file transfer protocol (FTP), Telnet. Monitored features include number of Telnet and FTP sessions, rate of ARP and ICMP packets, size of ICMP packets...

It is also worth underlying that some research works and projects address the global problem of intrusion detection in the smart grid as a whole, including several if not all layers of the SGAM (see Figure 1.2) and not only IACS. For instance, Levorato et al. [101] propose a probabilistic approach to detect anomaly in the global smart grid system. They consider a broad definition of anomalies covering malfunctions of physical entities of the grid (lines, production sites, etc.), but also unexpected or unforeseen behavior of production and consumption potentially leading to failure. Authors propose to investigate evolutions of variables such as weather conditions, consumers behavior, fossil fuel and renewable energy production and energy price. This global problem necessary means handling big amount of data from various sources and thus the choice of stochastic approaches is made. This global intrusion detection problem is beyond the scope of this dissertation, though.

1.4.3.7 Summary

Table 1.3 tries to summarize the elements of interest of the above analyses. Papers are organized in alphabetical order according to their first author's name. Second column is publication year. The third column gives the main intrusion detection approach following the taxonomy introduced in section 1.4.2. The fourth and fifth columns read information about the tuning technique and the source of tuning data respectively. Source of detection data is given in the sixth column. Column number 7 states what deployment option was chosen (e.g. centralized, distributed, stand-alone device). If a tool has been implemented, either based on existing softwares or from scratch, its name is given in the eighth column. Protocols under consideration are listed in column 9. If the IDS was developed for a specific industrial domain, it is mentioned in the tenth column. Column number 12 describes the experimental set up if there is one. And the last column gives the performance metrics the authors verified.

Table 1.3: Peculiarities of intrusion detection approaches presented in this state of the art

Authors	Year	Approach	Tuning technique	Tuning data source	Input data	Deployment	Tool	Protocols	Domain	Experiment	Evaluation metrics
Almalawi et al. [3]	2014	Process / whitelisting and blacklisting at once	Probabilistic (unsupervised machine learning)	Three datasets: one publicly available real dataset composed of sensors measurements and actuators states from a real urban waste water treatment plant, two generated by simulation set up	Raw records of SCADA data (same datasets as for learning phase)	Centralized (at supervision level)	*Not relevant*	Modbus/TCP	Water distribution	Virtual SCADA lab and simulation of a water distribution system (MATLAB)	Accuracy metrics: Recall, FPR, Precision, F-score. Classification time.
Barbosa et al. [13], [15]	2016	Communication (mainly structure, some grammar aspects)	Probabilistic (machine learning)	Traffic traces from three real-world IACS (three datasets containing more than one day of data, free of attacks, the first 30 minutes are used as training sets and the remainder as test sets)	Network traffic	*Not relevant*	No tool, only the algorithm is tested	Modbus TCP, MMS	Water treatment, electricity/gas utility	Evaluation of the proposed algorithm with a test data set composed of half of the collected data	Modeling accuracy of the algorithm (validating the periodicity assumption)
Berthier and Sanders [16]	2011	Communication (grammar, structure, telemetry) and Intelligent nodes	Deterministic	Experts' knowledge based on threats model, protocol definition, analysis of meters expected behavior, historical data	Network traffic	Distributed structure is suggested but prototype implemented for experimental purpose is a centralized system	Python-based prototype on a Linux platform	C12.22 and C12.19	Advanced Metering Infrastructures: a communication infrastructure enabling information exchnage between meters and utilities	Simulated testbed composed of three virtual machines: one emulating meters, one for the collection engine and one hosting the intrusion detector, communications are emulated using a commercial software	CPU usage, memory usage, rates of packets and bytes processed, true positive and true negative rates
Borges Hink, Morris, S. Pan et al. [49]	2014	Intelligent nodes - Process knowledge	Several machine learning techniques	Open-source simulated power system data provided by Mississipi State University	Logs from control panels, Snort, relays (including analog measurements)	Centralized detection	*Not relevant*	*Not specified*	Power system	Simulated test bed	Accuracy of the classifiers, recall, precision, F-measure
Caselli et al. [19]	2015	Communication (vocabulary, grammar, structure) and Process	Probabilistic (discrete-time Markow chains)	Modbus traffic recorded in a real infrastructure composed of PLCs, RTUs, HMI, SCADA server (4 hours for training set, 1 day for test set)	Network traffic	*Not specified*	*Not specified*	Modbus, MMS, IEC 60870-5-104	Water treatment	Evaluation of detection algorithm using Modbus traffic from real infrastructure	True positive rate and false positive rate

Table 1.3: Peculiarities of intrusion detection approaches presented in this state of the art

Authors	Year	Approach	Tuning technique	Tuning data source	Input data	Deployment	Tool	Protocols	Domain	Experiment	Evaluation metrics
Cheung et al. [27]	2007	Communication (vocabulary, grammar, structure)	Deterministic (Snort rules and formal model) and probabilistic (Bayesian network)	Modbus application protocol definition and Modbus/TCP implementation guide, network access policies, traffic (learning phase)	Network traffic	Centralized	Snort, EMERALD, eXpert Net	Modbus/TCP	Industrial process	Sandia National Laboratories test bed, whose process control zone includes but is not limited to a SCADA server and a PLC	*Not specified*
Diallo and Feuillet [32]	2014	Communication (vocabulary, grammar, structure)	Deterministic (rule-based)	Modbus protocol specification	Network traffic	Decentralized, distributed	Surricata, Mirabox (hardware embedding a Linux kernel)	Modbus/TCP	*Not specified*	Evaluation of the proposed detection sensor using Modbus traffic generated by Digital Bond's Target Service. Experiment is done considering a single function, *Write Single Register*.	Analysis throughput according to number of rules
Fovino, Carcano et al. [42], [18], [43]	2010-2012	Process, Communication (vocabulary, structure)	Deterministic: set of rules tuned by experts.	A priori knowledge about system architecture, SCADA commands and set of critical states	Packets flows and data collected by a master simulator querying field PLCs	Plugged into existing firewalls	Snort	Modbus/TCP, DNP3	Power system	Test bed for Industrial Security Laboratory. Physical emulation of the power plant processes by a dedicated electromechanical device, actuated by an ABB AC800 PLC connected to SCADA servers typical of real world .	Latency, distance analyzing time, system state update time, and memory usage.
Hadeli et al. [46]	2009	Communication (mainly structure, but also vocabulary, grammar)	Deterministic (rule-based)	Information extracted from system configuration files, explicitly specified or inferred	Network traffic	*Not specified*	Snort	MMS, GOOSE, SNTP, Modbus/TCP, RNRP	Power system, process automation	*Not specified*	*Not specified*
J. Hong et al. [52]	2014	Communication - whitelisting (vocabulary, grammar, structure), and Intelligent nodes - blacklisting	Deterministic (protocols specifications)	Protocol definition, system knowledge	Network traffic, and system and security logs	Centralized anomaly detection system collecting data from several sources over the network	Detection algorithms implemented in C language and user interface in C++ language.	Broadcast IEC 61850 protocols GOOSE and SV	Power system	Washington State University test bed including substation network with a merging unit simulating analog physical values	Computational performance, false positive and false negative rates (FPR and FNR)

Table 1.3: Peculiarities of intrusion detection approaches presented in this state of the art

Authors	Year	Approach	Tuning technique	Tuning data source	Input data	Deployment	Tool	Protocols	Domain	Experiment	Evaluation metrics
Kwon et al. [97]	2015	Communication (mostly structure and telemetry, but also some vocabulary and grammar aspects)	Probabilistic	One week of normal traffic from a real Koean substation	261 five-minute long traces of traffic from a real Korean substation and 27 pcap files for most probable attacks on IEC 61850 environments	*Not specified.* Off-line analysis in the paper.	*Not specified*	GOOSE, MMS (attacks also exploits vulnerabilities of SNTP, NTP, ARP)	Power system (IEC 61850 automation)	Experiment is conducted in a digital substation environment, part of a Korean smart grid test bed. It includes an HMI, an IED and a gateway, to which the IDS sensor is connected.	Accuracy of the algorithm: TPR, FPR, FNR, TNR, Precision, Recall, F_1-score
Lin et al. [102]	2013	Communication (vocabulary and grammar)	Deterministic	Protocol definition	Sample of traffic collected at a real power facility, and synthetic malformed network traffic	Centralized. Authors suggest to explore further a multi-DNP3 analyzers structure.	Bro	DNP3	Power system	Simulated testbed: three virtual machines for control center, site field and the DNP3 analyzer	Throughput of packets processed by the parser alone and the analyzer, that is the parser and the security policy scripts
Linda et al. [103]	2009	Communication (vocabulary, grammar, structure, telemetry), whitelisting and blacklisting	Probabilistic (neural network)	Simulated traffic	Network traffic	Off-line analysis	*Not relevant*	*Not specified*	*Not specified*	Network data acquisition set-up includes a control PC and a PLC actuating a valve, communicating through a hub, which is also used as an connection point for intrusion attemps and data acquisition	Detection rate (recall), false positive rate
Manandhar et al. [105]	2014	Intelligent nodes - Process knowledge (state estimation)	Probabilistic (Fault detection approach)	State-space model of the power system measurements (voltage, phase, active and reactive powers)	Analog measurements collected by the central controller from sensors deployed over the process	Centralized detection	MATLAB	*Not relevant*	Power system	MATLAB simulation with Matpower package	Ability to detect changes (due to faults or attacks) and detection speed
Morris et al. [107]	2012	Communication (vocabulary, grammar)	Standard Quickdraw existing set of Snort rules for Modbus/TCP	*Not relevant*	Network traffic	Stand-alone module (passive or active configuration)	Snort, Quickdraw set of rules	Modbus RTU/ASCII (wireless)	Gas distribution	Mississippi State University SCADA Security laboratory	Latency due to analysis by Snort module

Table 1.3: Peculiarities of intrusion detection approaches presented in this state of the art

Authors	Year	Approach	Tuning technique	Tuning data source	Input data	Deployment	Tool	Protocols	Domain	Experiment	Evaluation metrics
Pan, Morris et al.[115]	2015	Intelligent nodes, blacklisting and whitelisting	Probabilistic (Common path mining)	Data collected from test bed	Logs from phasor data concentrator (PDU), relays, Snort, simulated energy management system (EMS)	Centralized, off-line analysis	Snort to detect remote trip commands but no detail is given about the core implementation of the detection algorithm	Modbus TCP, IEEE C37.118	Power system	Test bed includes RTDS (Real-Time Digital Simulator), relays and phasor measurement units at process level, and PDC, historian and a network event monitor.	Classification accuracy and accuracy against zero-day attacks
Parvania [116]	2014	Communication (vocabulary, grammar, structure)	Deterministic: rules are manually specified	Communication rules (protocol definition and system configuration) and knowledge of the system specific operation procedure	Network traffic	Centralized	Bro	Modbus/TCP	Power system (fault location and isolation process)	2 PLCs emulate the master station and the actions of circuit breakers, a Modbus master simulator acts as an attacker, an instance of the NIDS.	Capability of detecting the attacker's activity in three different attack scenarios
Koutsandria et al. [95]	2014	Communication (vocabulary, grammar, structure) and Process								Hardware-in-the-loop testbed: a COTS PLC is used as a master controller that monitors a MATLAB/Simulink simulated power transformer. Traces of simulated scenarios are then fed to the NIDS instance	
Premaratne et al. [120]	2010	Blacklisting NIDS	Deterministic: rules derived manually from captured traffic both genuine and malicious	Records of ARP-based packet sniffing	Network traffic	Authors consider either to connect the IDS to a switch through port mirroring or embedd it into the gateway	Snort	ARP (also mentions ICMP, FTP, HTTP, Telnet)	Power system (IEC 61850)	A local network with an IED a Linux traffic monitor and IDS and a working station, connected to the university network through a switch. Remote site includes a Linux attack host and a working station.	Detection accuracy
Shang et al. [124]	2015	Communication (structure, telemetry)	Probabilistic (one-class SVM)	Simulated data	Network traffic	Off-line analysis, only the algorithm is tested	Wireshark to capture Modbus/TCP traffic	Modbus/TCP	*Not specified*	Schneider M340 PLC, King SCADA software for HMI, engineering stations running UnityPro	Accuracy, detection time

Table 1.3: Peculiarities of intrusion detection approaches presented in this state of the art

Authors	Year	Approach	Tuning technique	Tuning data source	Input data	Deployment	Tool	Protocols	Domain	Experiment	Evaluation metrics
Singh et al. [128]	2015	Intelligent nodes - Process knowledge (state estimation)	Comparison of power flows analysis results with actual system state at each iteration	Online system state estimation (Fault detection approach)	Measurements received at supervision level	Centralized detection	PSAT (open-source Matlab toolbox)	DNP3, IEC 101	Power system	PSAT used for both process simulation, and anomaly detection at supervision level	Ability to detect attacks in two scenarios
Skopik et al. [129]	2014	Intelligent nodes, Process	Probabilistic (self-learning)	Logs from firewall, switch and SCADA from a utility provider (training set, test set)		Centralized	*Not specified*	*Not relevant*	*Not specified*	Evaluation of the detection algorithm using collected data	Accuracy, detection rules coverage of system events, detection velocity according to evaluation time, analysis throughput
D. Yang et al. [143]	2005	Intelligent nodes (resources) and Communication (telemetry)	Probabilistic	Existing auditing systems logs	Auditing systems logs of server I/O flows and hardware working statistics	Centralized (HIDS)	Process and Equipment Monitoring (PEM) MATLAB toolbox	*Not relevant*	*Not specified*	Experimental test bed includes a simulated SCADA system consisting of SUN servers and workstations to collect data, and PEM MATLAB toolbox	Capability of detecting abnormal system status

Table 1.3: Peculiarities of intrusion detection approaches presented in this state of the art

Authors	Year	Approach	Tuning technique	Tuning data source	Input data	Deployment	Tool	Protocols	Domain	Experiment	Evaluation metrics
Y. Yang et al. [146]	2014	Communication (vocabulary, grammar, structure) and Process	Deterministic	Protocol definition and system configuration	Network traffic	Stand-alone module connected to the substation LAN switch through port-mirroring	Internet traffic and content analysis (ITACA) tool developed by the authors, Linux-based, developed in C/C++	IEC 61870-5 (also mentions DNP3, ICCP, IEC 61850 and proprietary protocols)	Power system	Test bed includes SCADA of a real PV connected to the grid. Attacker station and IEDs are simulated.	Process time and accuracy compared to state-of-the-art methods
Y. Yang [148], [147]	2016	Communication (vocabulary, grammar, structure, telemetry) and Process	Deterministic and probabibilistic (for telemetry criteria)	Protocols specifications from IEC 61850 standard, IEC 61850 substation configuration files and normal IEC 61850 traffic contents				Substation supervision protocols (MMS, COTP, TPKT, SNTP) and process protocols (GOOSE, SV, IEEE 1588)	Power system (500kV substation)	State Grid Key Laboratory of Substation Intelligent Equipment Testing Technology, China	
Yoon and Ciocarlie [150]	2014	Process, Communication (vocabulary, grammar, structure)	Probabilistic	Training dataset obtained from a Modbus network test bed composed of 2 masters and 25 slaves. Test dataset obtained by adding random abnormal sequences to it.		*Not relevant*	No tool, algorithm is tested with communication traces of a single master/slave connection	Modbus	*Not specified*	Traces captured in a Modbus network test bed (2 masters and 25 slaves)	Detection rate, False Positive Rate
Wu et al. [141]	2014	Communication (vocabulary)	Deterministic (rule-based)	*Not specified*	Network traffic	Distributed firewall	Linux machines	DNP3, Modbus, GOOSE, remote access protocol	Power system	University College Dublin test bed includes two control centers and two substations	CPU usage

Conclusion

As we focused this state of the art about ICS-oriented NIDS on behavioral approaches, the reviewed research papers globally take into account ICS specificities such as discussed in section 1.3. We find surprising that it counts about as many probabilistic approaches as deterministic ones, though. Industrial systems demonstrate several fixed or at least bounded behaviors, regarding either protocols, network topologies, communication structure or process state. In our opinion, this information, generally readily available, shall be used in ICS NIDS design. Protocol specifications, configuration files or other formalized system specifications exist and may help in tuning detection rules, even allowing to automate this task to some extent. We do not advocate to use this single approach but we believe that these characteristics are worth to be exploited in the first place and translated into detection rules.

Machine learning approaches may exacerbate interdependencies of some features, though, and thus help reduce dimension of features to check. But these algorithms are resource intensive. Of course, if the detection operation is run "backstage", at the highest of the IACS levels (3 - supervision or 4 - enterprise as modeled in Figure 1.1) and detection time is not correlated to process operation time, computational power is not at stakes. In such an approach, a centralized deployment of detection strategy seems relevant: IDS is implemented at IACS level 3 or 4, it collects data from all over the system and compute detection algorithms. But we believe that the trend is going towards distributed intrusion detection architectures, aggregation and correlation of results analyses being done at supervision level. Thus simplicity and frugality of detection approaches are valuable to us in the context of industrial systems.

Another reason that makes us prefer deterministic approaches compared to machine learning / probabilistic ones is the cost of false detection in ICS, especially false positives. As discussed in section 1.3.6, ICS priority is availability first. Deterministic approaches are very accurate for the scope they cover, but this scope may not be exhaustive. Multiple strategies thus may help balance strengths and weaknesses of several approaches.

Papers addressing specifically intrusion detection in IEC 61850 environments ([46], [52], [97], [120], [146], [148], [147], [141]) were mostly (six out of eight) published in 2014 and after, demonstrating the recency of this research field. As sumed up in section 1.4.3.6, they adopt different approaches. We must stress how close the work by Y. Yang et al [147] is to ours, it was published in April 2017, first available online in August 2016. Further detail is given in Chapter 4.

As we learned about intrusion detection, it became clear that any attempt of developping cyber security tools, including IDS, is relevant only once the risk has been assessed and evaluated. Next chapter presents risk management and assessment, their purpose and existing approaches.

Chapter 2

Assessing cyber risk of smart grid systems as cyber-physical systems

Contents

Introduction

In order to propose relevant security measures, including intrusion detection, it is important to understand the risk encountered by the system to be protected. Thus, along with the emergence of OT appears the need for methods dedicated to assessing cyber risk of CI that must combine information and process aspects. The objective is to understand interdependencies of both the cyber security risk and the dependability risk: how a cyber incident may compromise dependability attributes. Main concern arises from cyber attacks and such methods shall help understand the CI exposure and come out with security requirements specification. Literature gives many examples of initiatives proposing risk assessment approaches fitting smart grid needs.

In section 1, we briefly define the concepts of risk management and risk assessment, and their objectives. Section 2 then describes significant methods from both Dependability domain and Information Security domain, pointing out their strengths and limitations in the scope of cyber risk assessment of CI. Section 3 reviews prescriptions made by standards and governmental guides addressing cyber risk management in OT environments. In section 4, we present general remarks about risk assessment that we formulated based on our literature review. Next section, number 5, presents some works that tackle cyber risk assessment of industrial plants and smart electricity grid. The sixth section of this chapter focuses on test beds dedicated to power systems cyber security as they play an important role in the process of assessing risk.

2.1 Purpose and objectives of risk assessment

As the terms of “risk analysis” and “risk assessment” may seem interchangeable in the Dependability domain, let us clarify the vocabulary for this work. The international standard ISO 31000 [79] defines the generic risk management process regardless the nature of risks nor the industry. This process is applied by ISO/IEC 27005 [80] for information security risk management as illustrated in Figure 2.1. Risk analysis is a step of the risk assessment process (sequence of activities), which is itself part of a global and recurrent risk management process.

Risk assessment purpose is to provide the workforce with sufficient knowledge, awareness and understanding of the risks, to justify the control measures aiming at minimizing the risks and to provide regulation bodies with the required information in a global risk management process. Once the system has been defined in detail, the initial step of risk assessment is hazards identification, then comes the risk analysis, that is evaluation of likelihood and consequences (magnitude and severity) of all hazardous events, and third step is risk evaluation, which consists of comparing risk analysis outcomes with security criteria to prioritize risks regarding treatment. Final outcomes of the global risk management process are control measures to treat risk to an acceptable point, that is a trade-off between benefits of risk reduction and cost of further risk treatment [112].

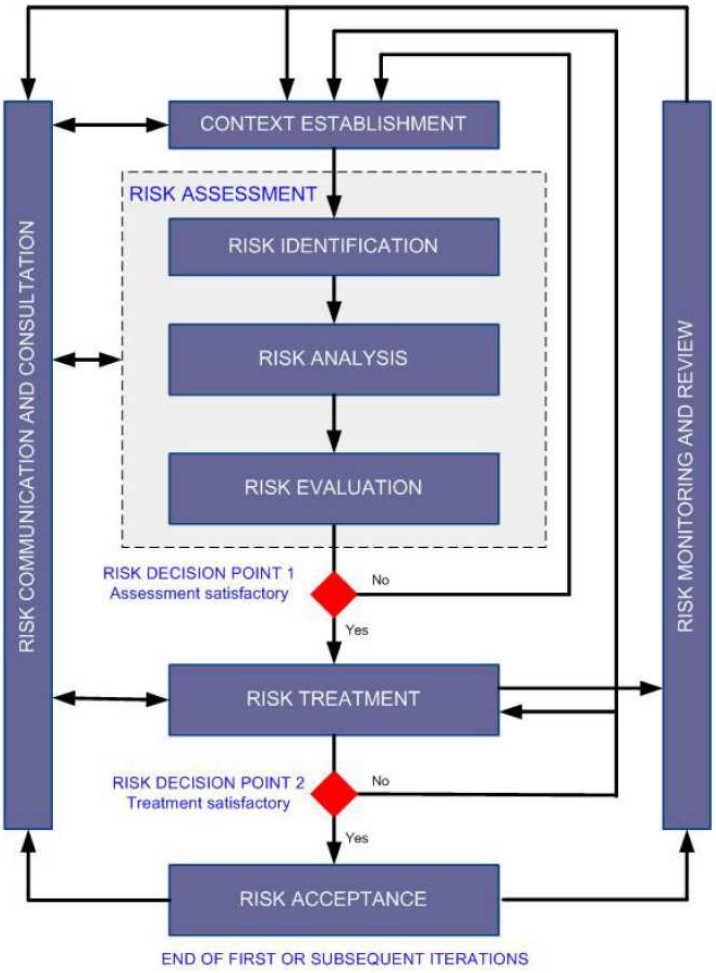

Figure 2.1: An information security risk management process from ISO/IEC 27005:2011 [80]

The need for risk assessment methods arose with the development of complex industrial systems, composed of many diverse subsystems (mechanical, electrical, pneumatic, hydraulic...). Risk assessment was meant to help make their production, operation and maintenance more efficient, that is reliable and available, and safer for people and assets. Definition of Dependability is thus the capacity for an entity to fulfill defined functionalities in defined conditions [136]. Dependability domain offers many mature risk assessment methods thanks to decades of development and practice, especially in military and industrial areas [134], [136]. Digital systems expansion is more recent and the first international standard stipulating recommendations about Information Security risk assessment was first released in 1997 (ISO13335-2 [77]). Risk management of information security is concerned with confidentiality, integrity and availability of data. Methods compliant with international norms have been used for many years in miscellaneous organizations.

While Dependability treats failures of tangible assets to answer concerns about safety, reliability and availability, Information Security considers random and deliberate alterations of immaterial resources (information) to ensure confidentiality and integrity. Al-

though, nature of assets and objectives are utterly different, risk assessment methods from both disciplines all have similarities, e.g. workflows are very alike as illustrated by Figure 2.1.

2.1.1 Dependability risk assessment

Among the most widespread Dependability risk assessment approaches, **FMEA (Failure Mode and Effects Analysis)** must be mentioned. IEC 60812 describes FMEA and its procedural steps, it also provides information on different FMEA methods [76]. It is an inductive analysis method for systematically studying causes and consequences of failures affecting the considered system components. It lets one estimate consequences of every identified failure mode of a component on the system functionalities and determine which failure modes have the most critical consequences onto relevant dependability objectives, such as safety, availability, reliability, maintainability... [136]. The analysis is complete when all failure modes of all components have been studied for every objective.

As a typical Dependability risk assessment approach, FMEA iterative process follows four main steps:

1. Defining the system, its components, its functionalities.
2. Identifying components failure modes and their causes.
3. Studying failure modes effects. This step may be completed with a criticality analysis, thus becoming FMECA - Failure Mode, Effects and Criticality Analysis. The purpose is to evaluate probability of occurence from the study of causes, and to characterize gravity of effects.
4. Conclusions, recommendations.

Regarding the risk management process depicted in Fugire 2.1, step 1 corresponds to *context establishment*, steps 2 and 3 to *risk assessment* and step 4 to risk treatment.

While FMEA is very generic, other methods concentrate on definite domains or systems. For instance, **HAZOP (Hazard and Operability Study)** is a method developed to assess dependability risk of thermodynamic systems (but used in others domains as well). It makes use of guidewords (none, more, less of, part of, more than, other) to help characterize deviations from expected operating conditions and catalog all possible causes of failures and their effects [137]. It is considered simpler than FMEA, indeed there is no need of systematically study all failure modes of each component and their effects. But it is deemed as error prone because associating guidewords to well identified portion of the system is difficult [136]. Such a method may not fit other types of systems, for which the required level of detail could not be met using HAZOP.

The **Preliminary Hazard Analysis** [136] consists of a rough first risk assessment. Experts deliver a qualitative list of possible hazards without considering any technical details nor numerous probabilities and consequences. Estimation of seriousness of the

hazards follows a coarse scale of just a few levels. HAZOP method covers risk identificaton and analysis steps of the risk management framework of Figure 2.1.

2.1.2 Information Security risk assessment

The international reference standard regarding risk management for information security is ISO/IEC 27005 [80], entitled "Information technology – Security techniques – Information security risk management". It specifies the global process of information security risk management, detailing its steps as shown in Figure 2.1. The approach follows the principle of continuous improvement, key for the implementation of an Information Security Management System (ISMS, introduced by ISO/IEC 27001 [81]): *Plan* (Establishing the context, Risk assessment, Developing risk treatment plan, Risk acceptance), *Do* (Implementation of risk treatment plan), *Check* (Continual monitoring and reviewing of risks), *Act* (Maintain and improve the Information Security Risk, Management Process). This is illustrated by the loop back in Figure 2.1. This standard does not provide any concrete method but gives guidelines. There exist several methods complying with the ISO/IEC 27005 framework, that support all or parts of the global risk management process.

The European Union Agency for Network and Information Security (ENISA) has produced and continuously maintains an inventory of risk management and risk assessment methods used in Europe[1]. Let us mention some of the most significant ones.

MEHARI (MEthod for Harmonized Analysis of RIsk) is a method developed by CLUSIF (CLUb for the Security of Information in France), an association of companies and experts from the private sector. It covers all phases of risk assessment (context establishment, stakes analysis and assets classification, risk identification, risk analysis and risk evaluation) and risk management (risk assessment, risk treatment, risk acceptance and risk communication). First version was issued in 1998 but CLUSIF continuously promotes and adapts it to evolutions of information and communication technologies and working changes, and most recent version MEHARI 2010 is compliant with ISO/IEC 27005. It is an open-source method and it comes along with a free supporting tool based on spreadsheet workbook, which contains knowledge bases and enables to conduct qualification and quantification of all elements of risk. It is suitable for medium to large structures of all sectors.

Another well established French information system risk management methodology was developed by the French government's Network and Security Agency (ANSSI) in 1995: **EBIOS** (*Expression des Besoins et Identification des Objectifs de Sécurité* - Expression of Needs and Identification of Security Objectives)[10]. Current version was issued in 2010, introducing elements specific to IACS and CI [9]. Phases of the EBIOS technique cover risk assessment and management. Used in the public as well as in the private sector, it

[1] https://www.enisa.europa.eu/topics/threat-risk-management/risk-management/current-risk/risk-management-inventory/rm-ra-methods

is compliant with major IT security standards among which the international standard for information security risk management ISO/IEC 27005 [80]. The method comes with a knowledge base including examples and advice to help build pertinent scenarios, and a free software tool. It is meant to be a simple tool enabling easy communication within the organization as well as towards business partners. As the EBIOS method is maintained and supported by a governmental agency, it may be an argument for organizations seeking certification, compared to MEHARI.

OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation) is another well known risk assessment and management method for information security by the US-CERT, first released in 1999, current version issued in 2005. It is aimed at large organizations with extensive means and skills. Experts from all business units and IT service perform a series of workshops to build the information security planning strategy based on the organization specific security operational risks. OCTAVE-S is a simplified variation of the method tailored to the needs and constraints of small entities (about 100 persons). The analysis team is composed of just a few persons with working knowledge of the important information-related assets, security requirements, threats, and security practices of the organization. Less experience of risk and security and fewer technical means are required as this version of the method is more guided. The objective of the most recent variant of the methodology, OCTAVE Allegro (issue date 2007), is to optimize the process of assessing information security risk without the need for extensive risk assessment knowledge or investment. It focuses on information assets and can be conducted in a collaborative way but is also suited for use by individuals who want to perform risk assessment without extensive organizational involvement, expertise, or input [17]. All three methodologies come with guidance, worksheets and examples.

Contrary to the previous methods, **CRAMM** (CCTA Risk Analysis and Management Method), issued by the British CCTA (Central Communication and Telecommunication Agency), now renamed Office of Government Commerce (OGC), in 1985, covers solely risk assessment (risk identification, analysis and evaluation). Current version holds number 5 and was issued in 2003. Use of the method is rather difficult without the CRAMM tool, which is a commercial product, and requires to be run by specialists. Aimed organisations are large companies and governmental bodies.

Other methods exist, developed and maintained by governmental institutions or experts associations, e.g. Magerit by Spanish government or IT-Grundschutz by the German Federal Office for Information Security. See ENISA inventory[1] for further detail.

2.2 Peculiarities of assessing cyber risk in smart grids

The multidimensional nature of the smart grid makes it challenging to assess cyber risk. On one hand, smart grid is both cyber and physical: combining methods from

Dependability and Information Security domains is at an early stage and research is still ongoing to produce tools able to grab interdependent cyber and physical vulnerabilities, so it gives to comprehend not only cyber risks and physical risks apart but combined cyber-physical risks [110]. We will present such initiatives further. On the other hand smart grid is a system of systems interconnected with each other through communication and process links. The Smart Grid Architecture Model (SGAM) developed by CEN-CENELEC-ETSI Smart Grid Coordination Group (SGCG) in the establishment of a common European reference model illustrates this complexity (see Figure 1.2 and 2.2) [22]. A cyber-attack at

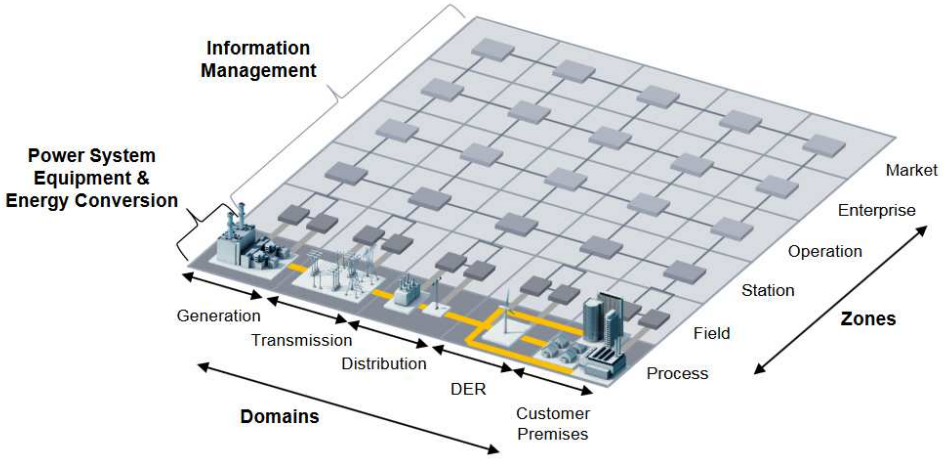

Figure 2.2: Domains and horizontal zones of the Smart Grid Architecture Model [22]

one subsystem may actually target and have consequences at another distant subsystem, that is geographically far away or from another domain or zone as in Figure 2.2. It was the case, for instance, in the cyber attack endured by the Ukrainian power grid in December 2015: attackers intruded the IT networks and gained access and control to the ICS using a malware [100]. Adequate risk assessment methods should make linkages between hazardous events clear and help to understand how an attack can propagate through the global system, compromising alternately cyber and physical assets. This is basically expected from all Dependability risk assessments as cascading effects are critical to all complex systems [111], [56].

This two points are well recognized in literature [48]. Hecht et al. [48] also stresses the wide age range of devices and technologies: a retrofit often results in new devices and technologies adjoining legacy ones. This may possibly result in additional risk on legacy systems as well as on the to-be-added ones. Understanding such implications is especially important at design time. Also, determining failure modes in Dependability and threats or vulnerabilities in Information Security is based on feedback and tests for known components and on experience of comparable components for new ones [136]. But the cyber threat does not hang over industrial items for long so there is none or few feedbacks, added to the fact that stakeholders are reluctant to communicate about cyber intrusions they encountered. Evaluating the risks may be tricky as testing may not be possible on site. There actually is a general lack of experience regarding cyber attacks on the power grid and developping tools and methods to analyze cyber attacks vectors and

consequences is another research challenge [48], [110]. Test beds are part of such solutions (see section 2.6).

2.3 Prescriptions from standards and governmental guides

US standard NERC-CIP-002-3 Critical Cyber Asset Identification: Standards mainly just give broad requirements that a risk assessment process should meet but do not explicitly prescribe any method. Thus, US standard NERC-CIP-002-3 about cyber security of the bulk electric system [114] recommends to *"identify and document a risk-based assessment methodology"*. So any method may be suitable as long as the procedure is well documented and applied.

US standard NIST.IR 7628 Guidelines for Smart Grid Cyber security: Another US government's contribution may help identify threats and vulnerabilities while assessing risk: NIST.IR 7628 [110]. It identifies extant security issues in the smart grid, both specific to smart grid (access to systems logs, trust of field devices...) and more general IT security that also apply (user authentication, intrusion detection, patch management...) but advises that organizations develop their own cyber security strategy for smart grid, including a risk assessment methodology. NIST.IR.7628 is based on many federal standards and guidelines about both security of IT infrastructures and the Electricity sector [113], [109]. It also used initial versions of the international standard ISA/IEC 62443 [5].

ISA/IEC 62443 Security for Industrial Automation and Control Systems: ISA/IEC 62443 standard builds on an international standard about IT Security, the ISO/IEC 27000 series [81], [82] and proposes refinements to it according to the differences existing between industrial automation control systems and business/IT systems [85]. It prescribes to conduct first a high-level cyber security risk assessment to get the whole picture of the system under consideration and then a detailed one of every *"zone"* and *"communication pipe"* [86]. ISA/IEC 62443 also underlines the strong interweavings between safety and cyber security as it stands that dependability risk assessment outcomes shall be input material to cyber security risk assessment. Dependability consequences of cyber hazards is out of its scope, though. As other standards, it does not mention any methodology. However, it specifies requirements that the chosen methodology shall meet for each of its steps. For instance, the IT security method requirements standardized in ISO/IEC 27005:2011 may serve [80]. When completed, ISA/IEC 62443 shall constitute an international reference offering a common language and harmonization of national and local requirements and initiatives. Especially, it aspires to establish a metric system to testify system security compliance. At the time of writing, this part of the standard is not complete yet [84]. For an overview of ISA/IEC 62443 international standard (in French), one may refer to [93].

EU efforts about Smart Grid Security: The European Union also produced, through the European Network and Information Security Agency (ENISA), a number of guides, adapting (among other sources) the three documents mentioned above to the European specific case. Especially, it produced a report [37] providing a list of proposed security measures from eleven security domains (for instance the 9^{th} - Information system security and the 10^{th} - Network security). This list is aimed at helping assets owners to identify which may be relevant to their plants after having run a comprehensive risk assessemnt, whose method is in the analysts' hands. This report shall be used in support of SGIS Framework (formerly SGIS Toolbox) risk assessment methodology, which has been elaborated by Smart Grid Coordination Group from European standardization bodies CEN-CENELEC-ETSI [24]. This methodology objective is to make decision makers aware of the risk and to help them in their choice of security measures to implement. It cannot be considered as a proper tool (and this is why its name was changed, to avoid misexpectations, see section 2.5 for details) and it is advised that CI owners use their preferred risk assessment method just tuning it to take into account SGIS Framework guidance. Some of this SGIS Framework key components are the SGAM (Figure 1.2), Security Levels definition and some typical use cases.

ANSSI Classification Method and Key Measures: Cyber security and especially cyber security of CI has been identified as critical to national sovereignty by French law No. 2013-1168 of December 18^{th} 2013, the *Loi de Programmation Militaire*. In this context, the French Network and Security Agency (ANSSI) is structuring public-private advances towards a common national cyber security understanding and coverage [30]. ANSSI-leaded Working Group on ICS cyber security published a set of documents to support the effort of ensuring cyber security level of new ICSs *"given the current threat status and its potential developments"* [8] and certify them, including *"human and material resources designed to control or operate technical installations"*.

The classification method established in [8] does not pretend to be a comprehensive risk assessment method, although ISO/IEC 27005 and EBIOS were resources to this work. The objective is to determine criticality levels (risk or impact of an attack is 1-low, 2-significant or 3-critical) of the assessed systems regarding cyber attacks. ANSSI classification method is summarized in Annex A, details are available in [8].

Determining such security classes aims to guide responsible entities in setting up relevant security measures, either technical or organizational. Key measures are identified in the document. Systems of class 2 or 3 shall be subject to a thorough risk assessment following a method of the choice of the responsible entity and well documented. Indeed the presented classification method is too simple to depict a clear view of security situation and needs, a lot of details are lost along the procedure. It is a preliminary action to a global security management policy that shall help responsible entities to predict the efforts that the certification process will require.

2.4 Remarks on risk assessment methods

Our review of literature about risk assessment made us understand some generic important reflections, common to Dependability, Information Security and even CI cyber security purposes, but it also brought out some specificities of CI cyber security risk assessment.

2.4.1 Keep in mind objectives of the study

Objectives of the risk assessment must be clearly defined prior any other consideration. And they must be kept in mind the whole process long to make sure the analysis will answer its purpose [136], which may be legal compliance, preparation of an incident response plan, description of the information security requirements for a product, service or mechanism, etc... [80].

2.4.2 Importance of context definition

Clear objectives shall help to establish the context of the study and well-defined system boundaries. Results of the analysis will obviously be strongly influenced by decisions taken during this phase [136], [40], [106], [56].

2.4.3 Consider all system states of operation

It is widely acknowledged that a system does not face the same risks depending on its state. Each operation mode has peculiarities that influence the flow of events and the consequences of risk scenarios. It is thus important to list and characterize every state of the system and run the analysis for every one of them [136], [111], [40], [93]. Regarding the smart grid, data availability is thus highly critical under certain conditions (fault occurence for instance), while it is of lower importance in stable conditions [37].

2.4.4 Inevitably arbitrary and subjective

Arbitrary and subjective judgments are part of all risk assessment methods. They thus have some incertainty that must be understood and described [111], [106]. Asking for an expert judgment during perimeter definition and further phases would strengthen final results value.

2.4.5 Continuous and long-term process

Risk assessment must be used at every phase of the system life cycle: at design time, installation time and during operation [136], [111], [37]. It thus concerns product supplier, system integrator and asset owner, one at a time or all together [83]. Risk assessment is a continuous and long-term process: a security policy should schedule risk assessment on a regular basis to follow evolution of the system environment and risks, adapt countermeasures or plan retrofit [40]. This is especially true regarding cyber security as threats and technologies change very fast.

2.4.6 Iterative process

One of the main distinctions between Dependability and Information Security risk assessment methods is that, when concerned with cyber security, it is important to evaluate intrinsic risk as well as residual risk after considering effectiveness of available security measures (MEHARI, Magerit). Dependability methods generally propose a single step of risk assessment. However all methods, from both domains, strongly advise to run risk assessment in an iterative loop until residual risk is acceptable.

2.4.7 Quantitative and qualitative methods

There exist quantitative and qualitative methods. While the first ones endeavor to measure risk numerically, the second ones classify risks more subjectively. Quantitative approaches often are appropriate for Dependability purpose as the field benefits from extensive experience and feedback and failure occurences and consequences can be numerically expressed. When regarding cyber security, methods are mostly qualitative because of a general lack of information about threats, vulnerabilities and incidents, which is due to responsible entities reluctance to communicate about it and relative recency of the field, especially for CI cyber security [122]. Moreover zero-day attacks make state of the art of CI cyber security continuously grow, thus making complicated to capitalize on past experience and historical data to quantify vulnerability exposure. Difficulties of quantifying security risk is discussed by Verendel [135] and disadvantages of probabilistic methods, that are most of the quantitative methods, are stated by Cherdantseva et al. [26]. Quantitative probabilistic risk assessment methods are still quite popular among researchers as they bring a convenient numerical estimation of risk, their meaning is comparable to qualitative indices, though.

2.4.8 Likelihood vs attack cost

In traditional Dependability risk assessment methods, risk is expressed as a combination of likelihood of occurence and severity of impact. But the concept of likelihood

is often deemed irrelevant in the context of cyber security or does not convey the same idea. What really matters is the cost of the attack and the attackers' capabilities (skills, knowledge and resources) [8]. Likelihood should then translate how difficult the attack is. Severity is still significant in characterizing risks and prioritize them, though.

2.4.9 Inductive and deductive methods

Risk assessment methods can be either inductive or deductive, given the application domain and the experience of analysts. Inductive approaches have the advantage to consider a wide range of failure or security breach causes, if not all. For CI cyber security, an argument against induction is that an event may have consequences that do not cause any harm and such an approach would imply a huge amount of work with poor result [40]. But there is also an argument against deduction: starting from feared events to go back to causes will only consider the events that come to the analysists' mind. And building a risk analysis from attacks is limiting because, as said above, new attacks crop up every day and attackers' skills and means grow every day more extensive and faster. Using them as a starting point for cyber security risk assessment implies that CI security managers will always be late compared to attackers' advances. In fact, advantage can be taken from both approaches at different steps of the assessment: deductive methods may help identify the most impacting incidents and sketch broad scenarios to figure out some possible exploits, while inductive approaches may be useful to build detailed attack scenarios from threats and vulnerabilities.

2.5 A state of the art of cyber security risk assessment methods for IACS

This section presents some works dealing with cyber security risk assessment and management in IACS environments, the four last references ([106], [99], [21] and [56]) being specifically about the power domain. The first two references [92] and [26] are research surveys reviewing academic research papers on the topic. Then, come three methods applied to IACS from both academic and industrial worlds. The two last references present methodologies addressing cyber risk of the smart grid and are developed respectively by a European standardization body [21] and by a European research project team [56].

"A survey of cyber security management in industrial control systems", Knowles et al.

Knowles et al. [92] published a survey about cyber security management in industrial control systems. They reviewed standards and guidelines from government, industry and

standardization bodies, addressing on one hand information security (abundant publications) and on the other hand control systems security (burgeoning area) that could be applied to IACS environments. Criteria used for the analysis are: (i) type of publication (standard or guidance), (ii) scope (seven topics are identified, among which risk assessment and management), (iii) metrics availability (qualitative or quantitative), and (iv) dependability-security link unambiguously mentioned or not. For the latter, the authors introduce the "functional assurance" concept: mandatory behaviors of a system and its failure states shall occur both safely and securely (see 1.3), which must be distinguished and addressed in comprehensive terms in publications focusing on IACS cyber security. As the survey revealed a dearth of practical guidance on risk management and assessment methodologies, research activity on this topic is also discussed, including application of standards and best practices to IACS environments, as well as proposals to countermeasures to cyber threats and the role of test beds in helping verifying novel risk management approaches for IACS.

"A survey of cybersecurity risk assessment methods in the context of SCADA systems", Cherdantseva et al.

Another survey was issued the same year as the one by Knowles et al. reviewing specifically cyber security risk assessment methods for SCADA systems and written by Cherdantseva et al. [26]. As Knowles et al., Cherdantseva et al. note that in the last fifteen years, the need for risk management approaches specifically designed for cyber security of SCADA systems has become every year more tangible. Their article reviews the related academic literature. Cherdantseva et al. [26] reviewed twenty-four research papers published between 2004 and 2014 and proposing methods of risk assessment for SCADA systems, covering all or some parts of the process. The considered methods are described and examined according to the aim, the application domain, the stages of risk management addressed, the key concepts of risk management covered, the sources of data for deriving probabilities, the evaluation method and the tool support. Eleven explicitly apply to the electric power domain but others may apply to smart grids as well.

Mathods are examined according to intuitive axes. First categorization is made by the level of detail and coverage of each method. It thus belongs either to *guidelines*, covering most of the risk management process stages (as depicted in 2.1) with a low level of details, or *activity-specific methods*, detailing a particular step of the process, or a combination of these two categories, *elaborated guidelines*, that is a guideline enhanced with a focus on a specific risk management process activity. An interesting remark is that context establishement, when addressed, is limited to system or network configuration and hence only technical risks are taken into account.

The second categorization axis distinguishes *formula-based* from *model-based* approaches, the latter using graphical models. Most of the model-based methods presented in the survey are attack- or failure-oriented, while a few are goal-oriented, what the authors regret

as failure-oriented approaches are by definition incomplete. Risk assessment methods are traditionally split into *qualitative* and *quantitative*. The majority of the considered methods are quantitative and more specifically probabilistic. The authors logically ponder on sources of probabilistic data as they may strengthen or question the outcomes of the risk assessment. They identify two sources: *historical data* and *experts opinion*, some of the examined methods using both. Surprisingly, Cherdantseva et al. did not find indication of probabilistic data origin in five methods out of fourteen.

As for evaluation of the methods, the authors come to the conclusion that most of the methods are not demonstrated in full nor in a sustainable rigorous manner. Most of the papers just illustrate the presented approaches through a single case study or example, which means generic and simple models or test beds. This echoes the observation acknowledged by Knowles et al. [92] that the lack of real-world validation efforts is endemic in the surveyed literature. Cherdantseva et al. value implication of experts from industry at all the risk management stages and especially to evaluate the validity of the approach and its feasibility. The feasibility may also be supported by the existence of tools, such as software prototypes, and their availability. In the papers that Cherdantseva et al. have surveyed, information about an existing tool is mentioned in only seven of them and is quite scarce with no detail about the architecture nor the user experience.

One of the conclusions of this survey [26] is that the domain of SCADA cyber security risk management still lacks maturity and there are many ways to explore in order to improve methods.

Security application of a risk assessment method from Dependability domain, FMEA

Schmittner et al. [122] propose to extend FMEA with security into a "Failure Mode, Vulnerability and Effect Analysis" (FMVEA). The objective is to cast cause-effect chains of cyber attacks on cyber-physical systems. While failure modes depict origin of faults and their consequences, threat modes describe the compromising of a security attribute (e.g. availability, integrity, confidentiality...). As an example the technique is applied to a distributed industrial measurement system, such as met in power grid. The ambition of the presented FMVEA is to assess dependability and security risk through a single process. It seems well suited for a qualitative preliminary study of cyber risk of a system. However it seems unlikely to treat interdependencies of subparts of the system and cascading consequences of a cause, which is one of the challenges of smart grid risk assessment.

APERO: an analysis method for evaluating operational risks from the field

The effort to assess IACS dependability and cyber risks jointly was also the motivation to propose the open-source "APERO" method, "Analysis for evaluating operational risks" ("Analyse Pour l'Evaluation des Risques Opérationnels" in French) [118]. This method was developed by two small French engineering consulting companies, FPC Ingénierie and NexID in order to homogenize risk analysis methods dedicated respectively to dependability and cyber-security for their industrial informatics projects. So, it is a method from the field, used by engineers for industrial applications. It is based on a functional approach so it can be used at a global level and down to the granularity deemed appropriate. The effort is put on the operational security as principles of security at interfaces are quite known and common in the authors' viewpoint. It is important to consider both, though, to keep consistency of the global approach. Security goals considered are thus availability, integrity, confidentiality and traceability, to which we add reliability as it is also covered to our understanding of the method description. This method wants to be flexible enough to be used at any stage of a project and to integrate and leverage analysts' Fknowledge.

The four steps include:

1. Modeling of the function and characterization of its objectives: the mission of the function is considered and not its implementation as the focus is on operation. Qualitative criteria are important to determine success or failure of the function outcomes. Influence of the environmental parameters shall also be studied as they outline operational contexts and system interfaces. Source material is system specifications and, for advanced projects, design and implementation documentation.

2. Analysis of possible anomalies and their severity: anomalies are characterized regarding non or partial production of the expected outcomes and their non compliance with the qualitative criteria defined during previous step. All possible dysfunctions shall be identified and considered, even unlikely ones.

3. Analysis of possible causes: the possible causes of identified possible anomalies are the true stakes of this analysis as reducing risk will require to apply security measures to these causes. They correspond to alteration of function inputs (regarding availability and integrity for operational security, and if relevant confidentiality and traceability) and to problems of design and implementation.

4. Definition of security objectives: for each identified cause, its likelihood and consequences severity help to decide whether the risk shall be treated or accepted, and whether security measures shall seek to reduce occurrence likelihood and/or limit consequences severity. The same idea holds for interfaces or for specific contexts (as identified in step 1), scenarios may help. Objective of this step is to produce all elements for relevant choices of security measures to implement.

Deliverables are defined for all steps and sub-steps when relevant. Authors stress the importance of documenting all steps of the process.

APERO method keeps things simple and very adaptable to peculiarities of the system under consideration and to expertise of the analysts team. It thus seems interesting to us for assessing cyber security risk in IACS environment, including power substations or larger portions of the electricity grids.

An Information Security method, EBIOS, to assess cyber risk of a substation

McDonald et al. [106] have proposed to assess impact of information infrastructure malperformance on the operation of the electrical distribution network using EBIOS. This work was conducted within the framework of a partnership between academic researchers and FPC Ingénierie, the consulting company author of the APERO method presented in previous section. Their choice of a method from Information Security domain is motivated by the fact, among others, that the notion of risk as introduced in EBIOS is more general than in methods concerned with fault propagation such as FMEA or HAZOP. They have applied EBIOS method more specifically to risk assessment of the information and communication system supporting a substation operation. McDonald et al. conclude that EBIOS technique has proved to be appropriate to consider both random and deliberate sources of malfunction in a single risk analysis. But in the authors' viewpoint, EBIOS turns out to be incapable of casting the dynamic dimension of operation conditions: evolving network conditions may change the risk scenarios and their consequences and cascading effects cannot be comprehended either. 2012 EBIOS knowledge bases were used as a starting point for a cyber security-oriented risk analysis of its IACS by a French critical infrastructure from energy sector. They were adapted and enriched with assessors' feedbacks and vulnerabilities descriptions by the US Department of Homeland Security [40].

"From old to new: Assessing cyber security risks for an evolving smart grid" Langer et al.

Langer et al. [99] specifically address the challenge of having both legacy systems and emerging technologies in smart grids. They propose a two-stream risk assessment process: the conceptual approach focuses on near- and mid-term developments, i.e. not implemented yet, while the implementation-based approach leads to a security audit of existing systems. The workflow ordinarily consists of four double steps: system definition, threat and vulnerabilities analysis, security risks assessment and countermeasures proposition. This method was applied in the context of an Austrian research project to a national reference architecture of distribution grid that was sketched out with the help of utilities involved in the project. Experts from industry, including utility providers,

and academia have been involved throughout the project, especially to estimate threats probability of occurence and impact.

Metrics adopted to characterize risk include impact in terms of monetary loss, customer impact and geographic range of the effect (e.g. local, regional, national). Key security controls that the authors identify are ensuring integrity and authenticity of all communications, conducting security audits, interoperability and penetration tests, implementing effective change, patch and configuration management practices, and minimizing the attack surface by deactivating unused services.

SGIS Toolbox, a European initiative dedicated to smart grid cyber security

The Smart Grid Information Security (SGIS) Toolbox [21] is a risk assessment method specifically developed for smart grids, meaning it takes into consideration both the physical process and the information infrastructures to assess cyber security risk. SGIS Toolbox was issued in 2012 by standardization bodies CEN, CENELEC and ETSI after the M/490 Smart Grid Mandate by the European Commission [21]. A newer version of it, with refined objectives, was issued in 2014 under the name of SGIS Framework [24]. Langer et al. [98] present SGIS Toolbox risk assessment method through a use case example from typical smart grid use cases developed by CEN-CENELEC-ETSI Smart Grid Coordination Group (SG-CG) [23]: Voltage control and power flows optimization applied on DER (Distributed Energy Resources). In the SGAM framework (see Figure 1.2), such a use case covers the DER domain, all zones from process to market and all interoperability layers. Thus, impact categories include laws and regulations, reputation or financial consequences, among others.

The steps of SGIS Toolbox assessment method are detailed:

1. *Identify critical information assets* involved in the control strategy and that could be exploited in a cyber attack.

2. *Estimate risk impact* of every information asset compromising in terms of security objectives (i.e. confidentiality, integrity and availability) and expressed in five *risk impact levels* (from low to highly critical). Initiating events they explored are spoofed measurements (outside thresholds, oscillated, unusually wide distribution of measurements), manipulating DER reactive power state, falsifying the current tap setting, manipulating commands sent to power equipment. Operational, legal, reputation-related and financial consequences are considered. The modelling technique used to conduct this step is event tree analysis that lets one inductively explore potential outcomes associated to an initiating event. Analysis by Langer et al. takes into account success or failure of existing power system protections, which

is not a requirement from SGIS Toolbox. Risk impact metrics used by the method include power loss and geographic range [24].

3. *Identify supporting components* to map dependencies of information assets, which helps understand cascading effects. No further detail on this step is given in [98].

4. *Estimate attack likelihood* thanks to a threat and vulnerability analysis as specified in HMG IS1 standard [25]. Only results for the use case example are given in [98] following SGIS likelihood categories (five, ranging from low to extreme).

5. *Identify security level:* risk impact level and likelihood category are summed up in a security matrix to give the *security level* of the application regarding each of the security objectives (confidentiality, integrity, availability).

6. *Determine security measures:* From *security levels*, suitable countermeasures should be induced as given in NIST 7628 [110]. Authors recognize that SGIS Toolbox gives little guidance regarding this point. Possible countermeasures are related to access control awareness and training, audit and accountability, and incident response. According to the *security levels*, countermeasures are either mandatory or up to the stakeholders.

From this implementation of SGIS Toolbox risk assessment method on a use case example, Langer et al. [98] have found some limitations to it:

- Purpose of SGIS methodology is to evaluate inherent risk, meaning without any security measure: interesting but not enough from the authors' viewpoint. It should be a recursive process to evaluate efficiency of security means deployed and assess risk they also introduce.

- According to the authors, this method is not relevant for future smart grid topologies such as microgrids or distributed generation. However, in the most recent issue of CEN-CENELEC-ETSI Smart Grid Information Security report [24], which describes SGIS Framework, the new version of SGIS Toolbox, a use case of Distributed Energy Resource (DER) control is described.

- Likelihood assessment method advised by SGIS Toolbox, HMG IS1 [25], was initially for business IT and turns out to be limited when considering critical infrastructures in the authors' opinion. Indeed, it only considers attackers' motivations and resources to define a threat level ignoring any technical dimension of the system [56].

- Information is lost through SGIS Toolbox risk assessment process: very detailed information is gathered to assess risk and it ends into a classification of rough *security levels* used as a basis to determine countermeasures. It is complicated to relate security measures to the particular threat they address and thus difficult to estimate improvement. This weakness has been answered in the SGIS Framework by introducing the concept of traceability that must allow identifying factors leading to given recommendations [24].

- When studying the impact of cyber attacks onto the power system, important power equipment assets should be identified first and then the information ones to keep the double dimension of the system, both physical and cyber.
- According to Langer et al. [98], SGIS Toolbox does not provide enough support for assessing cyber risk, making it still challenging, and it does not provide any tool for multi-stage attacks.

Despite these limitations, SGIS Toolbox tried to provide a method covering all stages of risk management (see Figure 2.1).

SPARKS project: leveraging SGIS Toolbox assessment method

SPARKS stands for Smart Grid Protection Against Cyber Attacks. One of the objectives of this three-year European research project, launched in April 2014, is to bring answers to SGIS Toolbox shortcomings and propose a "Threat and Risk Assessment Methodology", detailed in project deliverable 2.2 [56]. Researchers started by applying what they considered the most relevant risk assessment method to smart grid, that is SGIS (Smart Grid Information Security) Toolbox as described in previous section. Thus, SPARKS assessment methodology is supposed to leverage SGIS approach, building on its concepts (SGAM, risk impact levels, security levels...) and on ISO/IEC 27005 information security risk management standard [80] as this standard brings well-defined approach and terminology for risk management and is widely accepted and used in industry (see section 2.1.2).

SPARKS threat and vulnerability assessment methodology consists of four steps:

1. *Context establishment with SGAM Framework:* The system can be described for every interoperability layer, mapping the assets of different domains and zones of SGAM (see section 1.1.2). This description includes organizational, informational and technical faces of the smart grid use case under consideration.
2. *Threat identification and likelihood assessment with attack graphs:* Authors propose to examine threats and their likelihood using attack trees. Starting point of this approach is to identify and focus on the primary information assets, that is the ones with direct consequences on the physical system. Potential attack goals are then derived for each of them regarding security objectives. These attack goals are the root nodes of the attack trees beeing built. Leaves correspond to source nodes. The result is a set of attack trees constituting the attack graph of the system under study with all identified attack goals. Authors propose patterns to help deductively build attack trees according to the type of the target asset (function, component, communication link). Like in SGIS Toolbox, HMG IS1 standard is used but only to estimate a threat level and not the likelihood level. HMG IS1 standard aims at assessing an attacker's motivation and capabilities. A *threat level* out of five is then

assigned to each source node to characterize complexity of running the attack (the easier, the higher the level is) and then propagated through the attack graph up to attack goals. Likelihood is estimated for diverse attacker's profiles.

3. *Consequences identification and impact assessment:* According to the stakeholders' interests, different categories of consequences exist and the ones significant to the ongoing risk assessment shall be identified. They may be related to safety, economic losses or damages to equipment... Once consequences have been identified, they should be numerically characterized using diverse methods ranging from expert's analysis or event-tree analysis to co-simulation. Authors of deliverable D2.2 describing the methodology [56] emphasize the fact that all methods may not suit all consequences of interest and are linked to the amount of knowwledge from context establishment and effort the assessor is ready to put in. Authors advise to use a minimal-effort method such as expert's analysis as a preliminary step and then deploy more elaborated methods to picture the impact with more detail. Thanks to numerical impact evaluation, consequences then are mapped to discrete impact levels similar to risk impact levels introduced by SGIS Toolbox. These five impact categories are defined for each consequence according to context specificities. Among the methods identified in [56], some are from Dependability field such as FMEA enriched with security dimension into FMVEA described earlier [122]. Co-simulation is also presented as a way to assess impact through experimentation on a two-part set up composed of power grid simulation connected to communication network simulation.

4. *Risk treatment with Semantic Threat Graph support:* As final purpose of risk management is to set up countermeasures to mitigate the risk, authors propose to use Semantic Threat Graphs (STG) to readily link the four typical concepts of risk management: high-level threats (as given by source nodes of the attack graph from step 2), assets involved in this threat, specific vulnerabilities that are exploited by this threats and countermeasures. Figure 2.3 shows the STG model. This representation makes explicit assets and links that are implicit in attack graphs. The combination of an attack graph with STGs for all source nodes gives an overall map of potential attack paths with knowledge and protection means for relevant components.

2.6 Test beds dedicated to power systems cyber security

Test beds are useful at several stages of a risk management procedure as it allows to analyze and test control systems in a more realistic environment than computer simulation and because such tasks cannot be done in the field. During risk asessment phase, it helps to study systems vulnerabilities, to implement potential attack scenarios, and to assess their likelihood and their consequences. When countermeasures have beend defined,

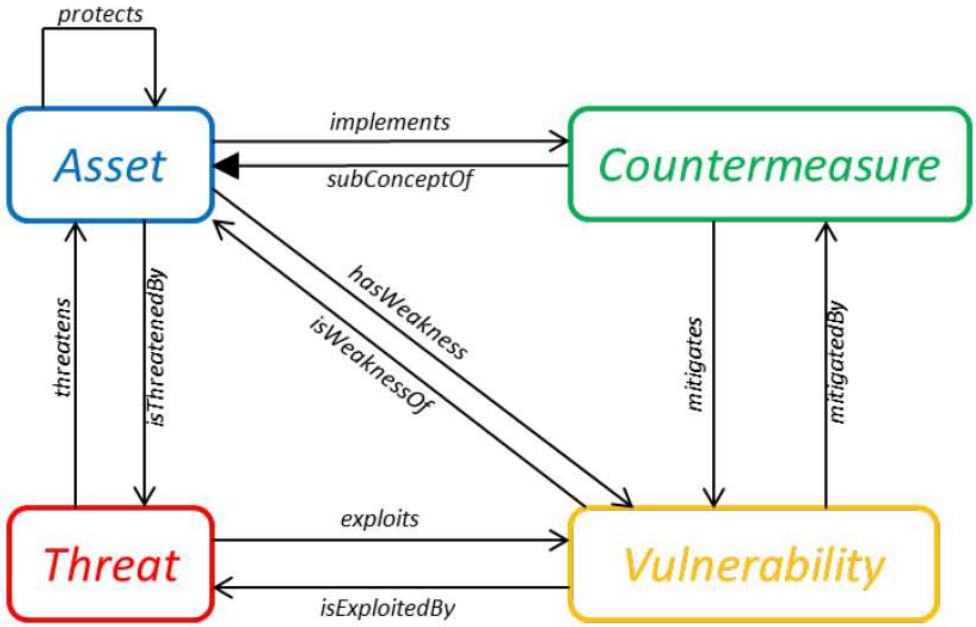

Figure 2.3: The abstract Semantic Threat Graph model defined in SPARKS Threat and Risk Assessment Methodology [56]

test beds offer a testing environment for validating technical solutions. It may also be used to train the workforce to technical and behavioral skills and to develop readiness in case of crisis situations (e.g. red team/blue team training session). Along with risk management activities of real power utility facilities, test beds are also necessary to support research activities, which overlap and reinforce each other. Such research applications, as identified by Hahn et al. [47], include vulnerability research, impact analysis, mitigation evaluation, cyber-physical metrics definition, data and model development, security approaches validation, cyber forensics, development of training scenarios. To provide an accurate cyber-physical environment, a smart-grid cyber security test bed must include three components: control, communication and power system.

One of the first IACS cyber security test beds was set up in 2003 on the U.S. Department of Energy's (DOE) initiative at the Idaho National Laboratory (INL): the National Supervisory Control and Data Acquisition (SCADA) Test Bed (NSTB), whose mission is specifically turned towards cyber security of IACS from energy sector critical infrastructures (CI). A key motivation of the NSTB program was assessment of IACS to identify and provide mitigation approaches for vulnerabilities that could put the infrastructures at risk to a cyber attack [57]. INL has several test beds for CI testing, including among others the NSTB, a Power Grid Test Bed and a Cyber Security Test Bed, who are expected to be used in cooperation to cover all CI security aspects [63]. INL Cyber Security Test Bed is a reference test bed, whose purpose is to offer an environment for tool development and simulation of attacks to IACS, and thus help industry and government to combat threats to the American CI.

Regarding cyber security of the power grid, INL is among the rare research facilities

with a full-scale system. INL operates a 61 mile, 138kV power loop with seven substations. Additionally, the laboratory owns a complete library of power simulation tools including a powerful software for grid modeling called Real-Time Digital Simulator (RTDS) [58]. In a nutshell, INL has tremendous means for a practical research integrating both cyber and physical dimensions of smart-grid cyber security.

Several more modest initiatives can be found in the literature. Power grid simulation software with hardware-in-the-loop often makes up for the impossibility of working with a real power grid system. Thus, Pan et al. [115] present a test bed with a RTDS, like the INL power grid test bed, that is used for validation of an IDS developed by the authors (see section 1.4.3.4 and Table 1.3). RTDS commercial software runs power grid simulation on-the-fly to ensure a behavior as close to a real network as possible and can be integrated with both physical and virtualized components. RTDS simulation is hardwired to real relays and measurement units to avoid time delay induced by communication between process simulation and real devices. This solution is costly, though. This second test bench is actually part of a platform developed at Mississippi State University (MSU) [108] with a double objective of teaching and research in cyber security of many critical industries.

Matlab/Simulink is a common software in academic environments and can be used for power system simulation. This is the solution chosen for the experimental framework used for evaluating the IDS proposed by Koutsandria et al. [95] and Parvania et al. [116] (see section 1.4.3.5). Simulink is used to run the virtual physical model of a transmission line. A communication module then manages Modbus communication with a PLC emulating a protection relay and performing the overcurrent protection algorithm.

The test bed presented by Yang et al. [145] is said to have the ability to perform end-to-end testing of cyber attacks and physical consequences. It has the three levels typical of an IEC 61850 substation. Process is emulated by a RTDS software and a complementary relay test set used as programmable voltage and current source. The bay level counts several IEDs, including protection relays, metering units and control devices, and a time synchronizer. At the substation level are RTU, a monitoring system, an engineering workstation. Authors realized an assessment of IEC 61850 vulnerabilities based on fuzzy testing. A fuzzer sends invalid data compared to the protocol specification to find vulnerabilities not anticipated by the protocol dsigners nor the software developers. In further work, Yang et al. [147] developed an IDS, which has been evaluated using this test bed.

Iowa State University (ISU) develops *PowerCyber*, a power system-oriented security test bed [47]. Additionally to RTDS, the process part of the test bed consists of software for non real-time power system simulation, which provides more advanced analysis capabilities but no connection to physical relays. Regarding communication elements, the test bed counts off-the-shelf IEDs and RTUs composing a substation, while another substation is virtualized. Protocols in use are IEC 61850 stack for the operational part and DNP3 for the control part. The control center has two SCADA servers, a workstation, a histo-

rian and a HMI. Further development presented by Ashok et al. [11] aims at providing the test bed with remote access capabilities. According to the authors, this shall help to answer the need for testing capabilities to a broad user community (both from industry and academia) and to build a sustainable and consistent research for improving smart grid security and resiliency. Research on vulnerability assessment and attack impacts are carried out in this environment [47].

Hahn et al. [47] identify a few other test bed development efforts, describing their interesting features and the research activities performed. Allmost all identified test beds use co-simulation for the sake of cost-effectiveness. The process is often simulated in a hardware-in-the-loop set up with real communication networks and devices, but the communication network can also be simulated in a system-in-the-loop network emulation. Research activities include cyber vulnerabilities identification, risk assessment methodologies specification, evaluation of security tools such as anomaly detection. Other examples of test beds or simpler platforms set up for intrusion detection approaches evaluation can be found in section 1.4.3.1.

Conclusion

What this state of the art evidences to us is that the term "risk assessment" actually covers a lot of different kinds of studies and activities. It all resides in the objective of such an initiative. From them will ensue the context study and boundaries of the system under consideration, the granularity of details and the amount of input data to handle, the choice of metrics characterizing the risk, the deliverables to produce according to the intended audience (e.g. regulation bodies v.s. internal diffusion only), the necessity for a repeatable procedure and thus for formalized and documented methods and tools, etc... Importance of the study objectives was highlighted in section 2.4.

Another remark is about the effort to put in. Even for complex study, it is advised to first assess risk in a broad and simple manner to identify points of interest and then refine the context and objectives for the subsystems requiring further and more thorough analysis.

It also illustrates well how valuable, if not indispensable, are experimental test beds in any attempts of assessing cyber risk of ICS. Among the benefits identified in section 2.6, let us recall that they allow to test technologies and devices and to gain experience and possibly discover vulnerabilities, they help eleborate attack scenarios and assess their feasibility and cost, they offer an environment to validate countermeasures including training.

In the following chapter, we apply this knowledge on a generic IEC 61850 substation in order to confirm intrusion detection is a relevant security measure to propose. The experimental set up at use in this work is part of the Grenoble Institute of Technology test bed for ICS interoperability and cyber security. Further detail about it and the

experimental part of this work can be found in chapter 4.

Chapter 3

Cyber security extension to IEC 61850 information model: specification of an intrusion detection function

Contents

Introduction

The IEC 61850 standard is very important for smart grid deployment as it focuses on "Communication networks and systems for power utility automation". However it gives no requirement regarding cyber security but the traditional user access control.

The main contribution of this chapter is the definition of a cyber security extension to the IEC 61850 information model, thus making possible to handle intrusion detection [89]. It is organized as follows. Section 1 gives an insight of a typical IEC 61850 communication architecture and details GOOSE protocol characteristics. A risk assessment of a generic substation example is presented in section 2. Section 3 introduces information modeling as defined by the IEC 61850 standard. And section 4 describes the proposed extension of the IEC 61850 standard for an anomaly detection function, including dedicated data objects specification.

3.1 The IEC 61850 communication

We introduced IEC 61850 standard in section 1.2.1, regarding its scope, objectives, structure and cyber security-related content. As already stated, IEC 61850 standard has two main contributions: a data object modeling approach to define the system in an implementation-independent manner, explained in section 3.3, and a concrete communication architecture including the definition of protocols mapping IEC 61850 data model. This section gives further detail about IEC 61850 communication with a specific focus on the GOOSE protocol.

3.1.1 Substation Automation System communication architecture

The IEC 61850 standard defines a communication architecture with three protocols tailored to the Substation Automation System (SAS) purpose. A SAS is logically divided into three levels according to their functional role similarly as in the IACS reference model recalled in Figure 1.1:

- **Process**: That is the lowest level of the SAS, the electrical grid to monitor, also called the primary equipment. It includes all power gears (lines, transformers, sources, charges...) including actuators and sensors.
- **Bay**: A power system is partitioned into subsets according to the protection functions to be implemented. A bay is responsible for the monitoring and protection of one of these subsets. Physically, the bay level is built up of IEDs that treat information from both the process and the supervision to possibly output commands to the lower level and reports to the upper level, and to share information among themselves. The bay level corresponds with level 1 of IACS reference model for “Safety and Protection” and “Basic Control”.
- **Station**: The uppest level has a global view of the whole substation. It centralizes data from all its bays and computes functions involving the entire monitored power system. Station level is also the entry point to the outer world e.g. for communication with a remote supervision center. It matches the levels 2 and 3 of

IACS reference model: “Supervisory Control”, “Site Monitoring & Local Display” and “Operations / Systems Management”.

Bay and Station levels constitute the control part of the IACS, the secondary system.

As shown in Figure 3.1, the protocol in use depends on the communicating entities: (i) Field sensors broadcast their measurements as **Sampled Values - SV** frames for bay level's IEDs, which need process data. (ii) **Generic Object Oriented Susbtation Event - GOOSE** protocol is for horizontal exchanges of measurements, metered values and command signals, at bay level. (iii) **Manufacturing Message Specification - MMS** (ISO/IEC 9506) is a supervision protocol for communication between substation level and bay level.

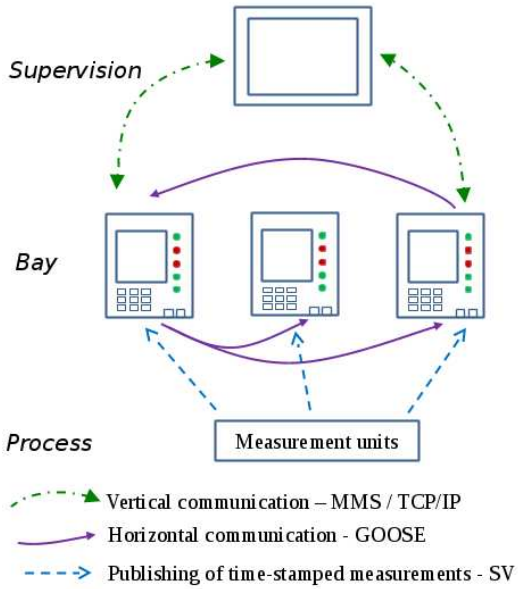

Figure 3.1: Communication architecture of an IEC 61850 Substation Automation System (SAS)

The communication requirements for an IEC 61850 SAS (as defined in IEC 61850-5 and presented in section 3.3.2) are met with the mapping of message types into the protocols stack shown in Figure 3.2. Message types and performance classes are detailed in section 3.3.2: Type 1 (Fast message), Type 1A (Trip), Type 2 (Medium speed message), Type 3 (low speed message), Type 4 (Raw data message), Type 5 (File transfer function), Type 6 (Time synchronization message), Type 7 (Command messages with access control).

MMS is an application-profile protocol suite mapped to TCP/IP fully specified in the ISO 9506 standard [78]. It is used for client/server connections. SV and GOOSE protocols directly map the Ethernet Link layer to fulfill real-time constraints of the corresponding message types. Messages are broadcasted according to a publisher/subscriber mechanism: all devices connected to the LAN see the frames but parse only the ones, that they have subscribed to. Detail about the GOOSE protocol follows.

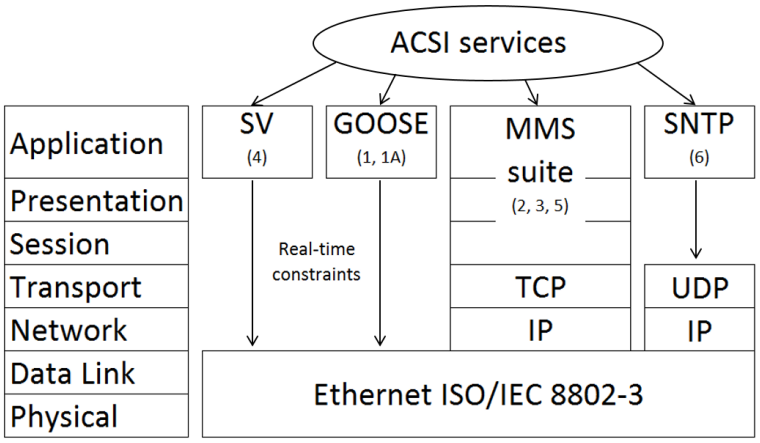

Figure 3.2: OSI mapping of IEC 61850 protocols

SNTP stands for "Simple Network Time Protocol" and is used for time synchronization of the whole system. ACSI (Abstract Communication Service Interface) services are models of information exchange defined in IEC 61850-7.2. They are specified independently of concrete implementation, which is defined in Specific Communication Service Mappings (SCSM), which makes the correspondance between ACSI services and real protocols services.

3.1.2 GOOSE protocol

3.1.2.1 Frame structure

GOOSE frame is standardized (ISO/IEC 8802-3) and detailed in IEC 61850-8.1 [72] (see Figure 3.3). The frame header is 26-byte long. It starts with the *Destination address*, a 6-byte string representing a multicast MAC address, whose value is defined by the standard. The first three bytes are assigned by IEEE with "01-0C-CD", the fourth byte shall be "01" for GOOSE and the last two bytes shall be used as individual addresses. Thus, the range of acceptable GOOSE multicast addresses is "01-0C-CD-01-00-00" to "01-0C-CD-01-01-FF".

The *Source address* is the unique MAC address of the publisher device.

Priority Tagging is used to separate time critical and high priority bus traffic for protection applications from low priority bus traffic. This parameter consists of a two-byte *TPID (Tag Protocol Identifier)* set as 0x8100, Ethertype of 802.1Q Ethernet encoded frame, and a two-byte *TCI (Tag Control Information)* defining message priority and whether a VLAN is used or not. It is a 2-byte long field, whose three most significant bits are for user priority, next bit is set to 0 and the twelve least significant bits are the VID - VLAN identifer. By default, GOOSE message priority is set to 4. Priorities allowed by

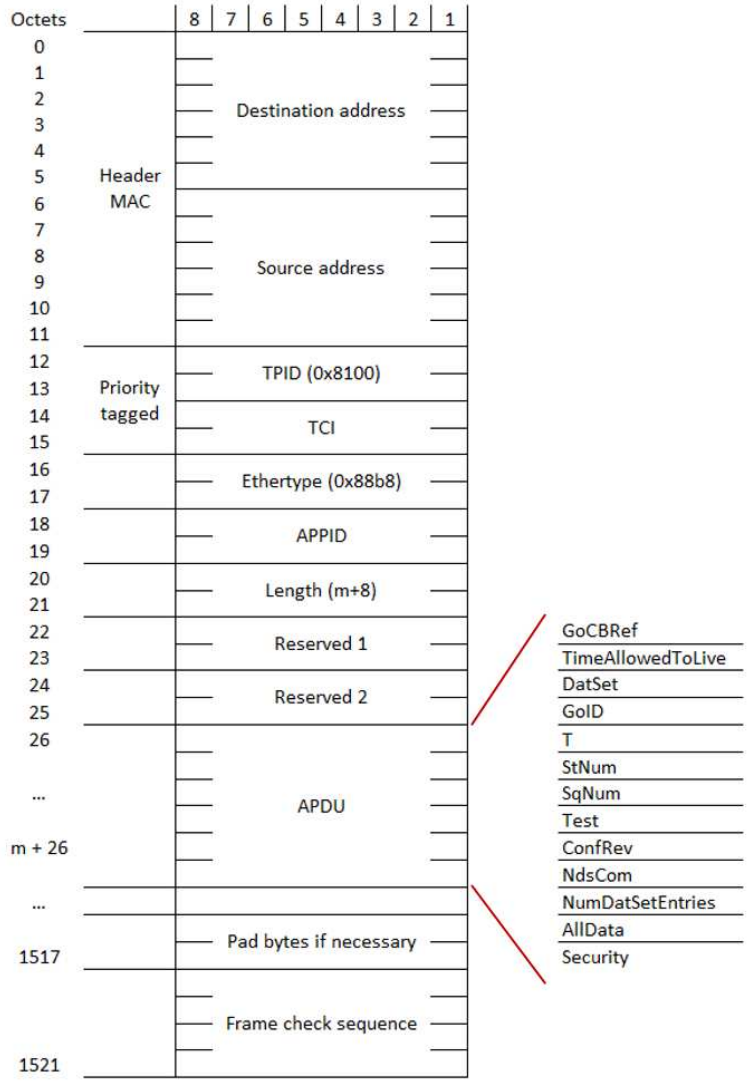

Figure 3.3: ISO/IEC 8802-3 frame structure for GOOSE communication

802.1Q are comprised between 1 and 7. 1 is for untagged frames. The highest priority set in a system shall be between 4 to 7. If use of VLAN is not supported, VID is set to 0. If supported, it is set by system configuration.

Ethertype based on ISO/IEC 8802-3 MAC-Sublayer is standardized and must be 0x88B8 for GOOSE messages.

The *APPID (Application Identifier)* is a two-byte application identifier ranging from 0x0000 to 0x3FFF for GOOSE. This field two most significant bits are 00 for GOOSE protocol, the other bits code the actual GOOSE application identifier. The standard recommends a unique application source-oriented GOOSE APPID within a system.

The variable part of the GOOSE packet starts with the GOOSE PDU *Length* field at bytes 20 and 21. It is the byte length of the PDU, including its header starting at APPID and the APDU itself: m+8 where m is APDU length and less than 1492.

The four following bytes are allocated to *Reserved 1* and *Reserved 2* fields. They are fields reserved for future standardized usage and are set to 0 by default.

The GOOSE *APDU (Application Protocol Data Unit)* contains the data sent by the transmitting application.

Last field of this frame is the *Frame check sequence*.

Parameters of the GOOSE APDU are listed in the right part of Figure 3.3:

- *GoCBRef*: *GOOSE Control Block Reference* is the name of the considered GOOSE control block, that defines transmission parameters,
- *TimeAllowedToLive (TATL or TAL)*: maximum time the subscriber waits before considering the connection lost,
- *DatSet*: *Data-Set* identifies the data to transfer,
- *GoID*: *GOOSE Identifier* is the GOOSE name, unique in the whole system,
- *T*: is the transmission time,
- *StNum*: *State Number* is a counter incremented when a variable of the data-set has changed and requires sending a GOOSE,
- *SqNum*: *Sequence Number* is a counter incremented every time a GOOSE message is generated. Reset to 0 when StNum is incremented,
- *Test*: boolean, true when in test phase,
- *ConfRev*: *Configuration Revision* is the current configuration number,
- *NdsCom*: *Needs Commissioning* indicates whether the GoCB needs an update, for instance it is true when the dataset is empty,
- *NumDatSetEntries*: *Number of Data-Set Entries*,
- *AllData*: the data-set values to transfer,
- *Security*: a field dedicated to security purpose.

Let us highlight that the *Security* field use is not detailed in IEC 61850-8.1, it is only said that this field is *"reserved for digital signature"* as suggested by IEC 62351 (section 1.2.2). Security measures are indeed recommended by the IEC 61850 standard but their implementation is up to the IED vendors.

Abstract Syntax Notation One (ASN.1) grammar in relation with the Basic Encoding Rules (BER) is used to encode the GOOSE messages for transmission on ISO/IEC 8802-3. Figure 3.4 shows the ASN.1 definition of a GOOSE PDU. The BER transfer syntax

```
IEC61850 DEFINITIONS ::= BEGIN
IMPORTS Data FROM ISO-IEC-9506-2
IEC 61850-8-1 Specific Protocol ::= CHOICE {
        gseMngtPdu          [APPLICATION 0] IMPLICIT GSEMngtPdu,
        goosePdu            [APPLICATION 1] IMPLICIT IECGoosePdu,
        ... }

  ...

IECGoosePdu ::= SEQUENCE {
        gocbRef             [0]   IMPLICIT VISIBLE-STRING,
        timeAllowedtoLive   [1]   IMPLICIT INTEGER,
        datSet              [2]   IMPLICIT VISIBLE-STRING,
        goID                [3]   IMPLICIT VISIBLE-STRING OPTIONAL,
        t                   [4]   IMPLICIT UtcTime,
        stNum               [5]   IMPLICIT INTEGER,
        sqNum               [6]   IMPLICIT INTEGER,
        test                [7]   IMPLICIT BOOLEAN DEFAULT FALSE,
        confRev             [8]   IMPLICIT INTEGER,
        ndsCom              [9]   IMPLICIT BOOLEAN DEFAULT FALSE,
        numDatSetEntries    [10]  IMPLICIT INTEGER,
        allData             [11]  IMPLICIT SEQUENCE OF Data,
        security            [12]  ANY OPTIONAL,
                                  -- reserved for digital signature
        }
```

Figure 3.4: ASN.1 encoding of GOOSE PDU

is a triplet (Tag, Length, Value). A Value may be a triplet itself. The transfer syntax is byte-based and big-endian oriented. The length field defines the length of the triplet (Tag, Length, Value).

3.1.2.2 Message transfer mechanism

The choice of a publisher-subscriber protocol allows to meet real-time requirements of protection-related messages. Such a messaging procedure has no acknowledgment for received messages. The ACSI service *SendGOOSEMessage* shall provide *"the possibility for a fast and reliable system-wide distribution of input and output data values"* as required in IEC 61850-7.2 [69]. To ensure this reliability, GOOSE protocol has a specific transmission scheme as shown in Figure 3.5. Considering a GOOSE application, when an event occurs resulting in some change in one or more variables of its dataset, a message is generated while state number *StNum* is incremented and sequence number *SqNum* is reset to its initial value 0. This GOOSE message is sent periodically at a high frequency (time period T1) the first two times and then at slower frequencies (with time period doubled at every step, T2 and T3 in Figure 3.5) until back to the frequency for stable conditions (T0).

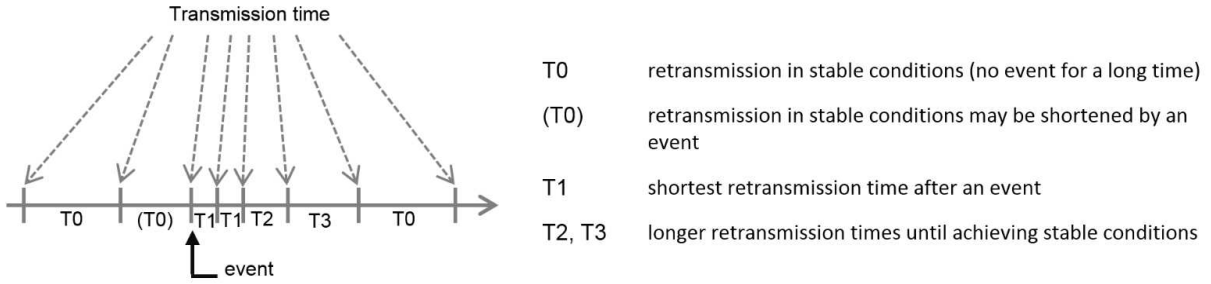

Figure 3.5: GOOSE transfer mechanism

The parameter *TimeAllowedToLive* carried by each message informs the subscriber of the maximum time to wait before considering the association as lost. T0 value shall be half the *TimeAllowedToLive* value, as suggested in IEC 61851-8.1 [72].

Counters status number *StNum* and sequence number *SqNum* are represented as 32-bit unsigned integers, thus having a possible value range from 0 to $2^{32} - 1$. StNum and SqNum initial values are respectively 1 and 0. When highest value is reached, the counter rolls-over back to 1.

3.1.2.3 GOOSE Control Block

A GOOSE application is configured in a dedicated GOOSE Control Block - GoCB. Communication applications control blocks are objects of the IEC 61850 data object model explained in section 3.3. They are part of the host's device logical node LLN0, which contains the generic information of the device.

Parameters of a GoCB are:

- *GoCBName – GOOSE Control Block Name*: unambiguously identifies a GoCB within the LLN0.
- *GoCBRef – GOOSE Control Block Reference*: is the unique path-name of the GoCB within the LLN0 (shall be <LDName>/LLN0.GoCBName).
- *GoEna – GOOSE enable*: is a Boolean indicating that the GoCB is enabled to send GOOSE messages when set to TRUE. When set to FALSE, the GoCB stops sending GOOSE messages.
- *GoID – GOOSE Identifier*: is a unique system identification of the application issuing the GOOSE messages (set to GoCBRef by default).
- *DatSet - Data-Set reference*: is the reference of the data-set, whose members values shall be transmitted by the GOOSE message.
- *ConfRev – Configuration Revision*: is a counter incremented each time configuration of the data-set is changed.

- *NdsCom – Needs Commissioning*: is a Boolean set TRUE if DatSet has a NULL value. It indicates that GoCB requires further configuration.

DatSet, *ConfRev* and *NdsCom* are carried by the PDU of every instance of GOOSE message.

The GoCB is fully defined with the specification of its services: *SendGOOSEMessage* (Send a GOOSE message), *GetGoReference* (Retrieve the data or data attribute of a member of the data-set), *GetGOOSEElementNumber* (Retrieve the position of a member of the data-set), *GetCoCBValues* (Retrieve the attributes of a GoCB), and *SetGoCBValues* (Set the attributes of a GoCB).

3.2 Risk assessment of a generic substation example

When working on cyber security measures, risk assessment of the system to protect is a logical starting point. It lets one get the overall picture of the situation prior to working on counter measures. It will help to be pertinent and put efforts on the right elements and up to the right degree of protection. Purpose and principles of risk assessment have been discussed in Chapter 2.

Cyber risk covers all types of information hazards: genuine incidents as well as malignant abuses. This dissertation focuses on cyber attacks targeting power grid applications as we seek to catch on their material consequences prejudicing dependability attributes of the grid such as safety, availability, reliability (see section 1.3.6). To grasp the effects of local malicious intrusions on the local as well as on a more global scale, it may be interesting to assess risk of the following systems:

- at the lowest granularity, the substation,
- a barely wider view including two substations and the communication link between them,
- a view including the substation, a supervision center and the link between them.

Outputs of risk assessment restricted to a substation may be inputs of the two others.

We limit the scope of the present study to a simple and generic substation, that is the smallest granularity of these three perimeters. The objective is to identify and study malicious compromising of communication that may prejudice a substation and its optimal and safe operating. We adopt an approach close to APERO (see section 2.5) as it is based on the system functional specifications and quite simple. Concretely, we got inspired by the three first steps of the APERO method (definition of the system missions, analysis of anomalies and analysis of causes) to grab a preliminary and generic picture of the cyber risk faced by IEC 61850 susbtations.

3.2.1 System definition and context establishment

The IEC 61850 standard defines typical realistic substations ranging from small to medium and large-size topologies for both transformation and distribution substations. We consider a small transmission substation depicted in Figure 3.6 with a simple topology along with a typical protection scheme and associated control functions. Details about this substation example were found in parts 1 and 5 of the IEC 61850 standard [73], [74]. Note that Figure 3.6 is illustrative and not exhaustive: only part of the equipment is shown. Regarding the SGAM framework (as defined in 1.1.2), the system under analysis is included into the perimeter delimited by "Transmission" domain, "Process, Field and Station" zones and "Component and Communication" interoperability layers. It consists

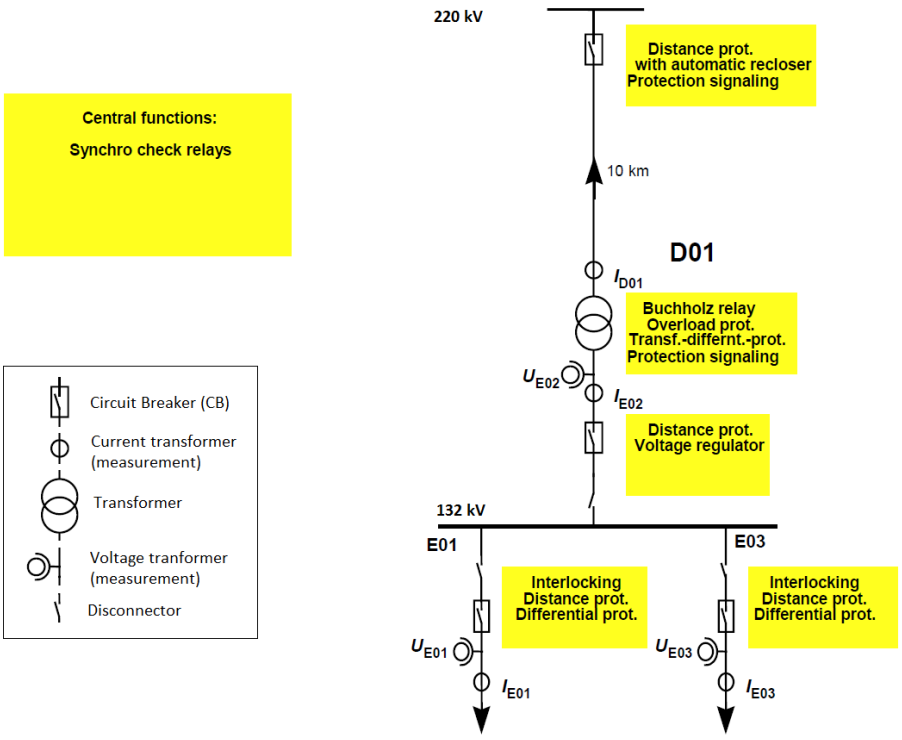

Figure 3.6: Substation of type T1-1 (transmission, small size, first topology) with allocated functions

of a single busbar with one incoming and two outgoing feeders, and one transformer. The substation level automation is limited to a remote control gateway and a simple HMI, that is a simple alphanumeric screen showing alarms and switch positions, allowing basic operation. The bay level includes control of all switches (circuit breakers and disconnectors) through IEDs. Let us assume one IED per group of functions (yellow rectangles in Figure 3.6). The standard does not mention measurement units: we assume that some of the sensors are directly integrated into IEDs and that other are measurement units that pass information on relevant IEDs over the process bus.

To support the substation functions, a station-wide communication bus is required for handling all types of messages [73]. All devices then have one Ethernet port connected

to a single switch, this bus assuming the three roles of process bus, bay control bus and substation control bus (as defined in section 3.1.1). Regarding communication network, the only substation **interface with the outside** is a gateway. Physically the local HMI also gives access to the substation automation but risks related to a fraudulent use of it is out of the scope of this analysis.

Given this description, the double cyber and physical characteristics of the substation is obvious. The physical part corresponds to the primary system: the portion of the power grid covered by the substation, including all cables, transformers, bus bars, circuit breakers, switches, etc. It basically supports electricity transmission and distribution, including adapting voltage level between two grid sections. The cyber components form the secondary system, the SAS, in charge of control, supervision, protection and monitoring of the primary equipment. It includes all supporting communication network and digital devices. It does not appear on Figure 3.6. This describes the **perimeter** and the **mission** of the system whose cyber risk is to be assessed.

Physically, the assets to protect are the physical components of both the primary and secondary systems. Functionally, we want to protect the substation automation, especially the electrical protection functions implemented in the SAS. They are programmed operations involving one or many IEDs that guarantee safety and security of the equipment and persons by isolating faulty parts of the electric grid to prevent destruction of the goods and injuries of people.

We consider the **states of operation** whose definition is given in IEC 61850-5 and taken from CIGRE – Technical Report, Ref. No. 180:

- *Normal*: Basic control and supervision tasks (parameter, measurands, commands).
- *Abnormal/alert*: Transformer overload, alert protection (overload, start/pick-up, some alarm and events).
- *Emergency/fault*: Action of protection (trip, alarms, events).
- *Post-fault*: Collection of fault information (fault parameters, disturbance records).

In the present risk assessment, for the sake of simplicity, we merge abnormal/alert state with emergency/fault state, as we stay generic and we do not want to go deep in electrotechnical considerations.

Functions of the assessed substation are listed on Figure 3.7. For a brief explanation of them, refer to Appendix D.

Several of them involve **communication**. The communication flows are shown in Figure 3.7.

- The function *Distance protection with automatic recloser* of the 10km long line is distributed among the two IEDs at both ends of the section. IED_{E02} is responsible

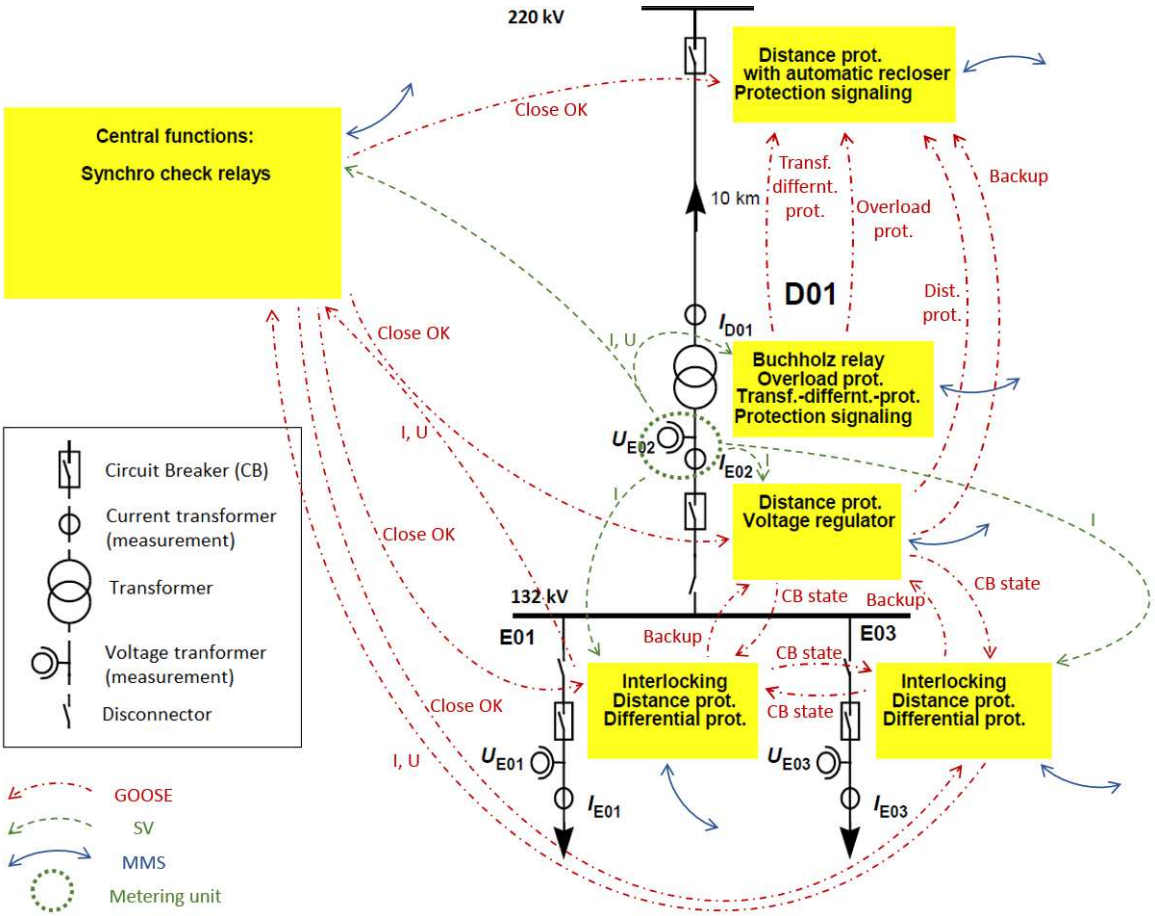

Figure 3.7: Substation of type T1-1 with communication flows

for computing the function using measurements from sensors U_{E02} and I_{E02} , which may result in sending a signal to IED at the entry point of the substation through GOOSE communication to trip CB_{D01}.

- The function *Interlocking* may either be fully hosted in a single IED or be distributed among several IEDs, depending on the switches involved. In the second case, GOOSE communication ensures signal transfer between the IEDs. In substation T1-1, we assume that the Interlocking function implemented in IED_{E01} and IED_{E03} needs the states of their related CBs and disconnectors and the states of CB and disconnector controlled by IED_{E02}. Switches states are transferred by GOOSE frames.

- The same remark holds for *Transformer differential protection* that may be fully implemented in a single IED or distributed among several IEDs depending on their measurement and switching capacities. Also, this protection function requires two current measurements on both sides of the transformer. We assume that current transformer I_{D01} is part of IED_{D01} and that I_{E02} is integrated into a standalone metering unit (along with the voltage tranformer U_{E02}, which is not used for this protection function). IED_{D01} sends tripping signals to IED_{E02} and the IED at the

entry point of the substation to open CB_{E02} and CB_{D01} respectively.

- The two *Differential protection* functions implemented in IED_{E01} and IED_{E03} both rely on their own measuring of respectively I_{E01} and I_{E03}, and on the current value I_{E02} measured by E02 metering unit. Both IEDs are subscribers to the SV application publishing this current value.
- The *Synchro-check* function is based on electrical measurements, meaning that it receives SV messages and GOOSE. It also sends signals to IEDs through GOOSE communication to authorize or forbid the closing of switches in E01, E02 and E03 zones.
- All IEDs have client-server communication with the HMI at station level and with a remote supervision center, outside the substation perimeter through the gateway. This link (blue arrows in Figure 3.7) is used for reporting to the supervision and transferring control signals to an IED. We only consider the portion of this link being inside the substation perimeter, bounded by the gateway.
- We assume an additional protection function that is not listed in Figure 3.6, *Backup protection*. In case of a breaker failure resulting in the persistence of a fault, neighbor breakers shall trip (see section 4.1.3.1 for a detailed explanation). Such a protection has need of GOOSE communication for tripping or blocking signals. Concretely, Backup protection of CB_{E01} and CB_{E03} trips CB_{E02}, while Backup protection of CB_{E02} trips CB_{D01}.

Qualitative criteria to decide of the functions outcomes success or failure include:

- The actual fulfillment of the function: Is the final state of the device the expected one? For instance, a circuit breaker state may be either open, closed, undefined or in failure mode.
- Reliability[2]: the "ability to perform as required, without failure, for a given time interval, under given conditions". In the specific context of industrial systems cyber security, a slightly wider meaning is: the double ability to actually run when called upon (dependability) and not at inopportune moments (security).
- Availability[2], that is the "ability to be in a state to perform as required".
- : Safety as regarding system integrity and physical security of people and assets.
- And in general, dependability-related criteria, dependability[2] being defined as the "ability to perform as and when required". As electrical protection functions have strong real-time constraints, questions to be answered include: Was the objective of the function met in the required time delay? And at the required moment compared to other events (synchronization)?

[2]Definitions given by Electropedia: The World's Online Electrotechnical Vocabulary, http://www.electropedia.org/iev/iev.nsf/d253fda6386f3a52c1257af700281ce6?OpenForm

3.2.2 Risk identification

As already mentioned, we focus on cyber risk and more precisely on *operational* cyber risk (as opposed to *IT* security: as an example, we do not consider exploits of OS, see section 1.3 for disambiguation). Therefore, we only analyze functions relying on communication. Also as our system is quite limited, we focus on technical consequences: dependability (reliability, availability, safety) as defined in previous paragraph. Legal, environmental, financial, brand image... aspects are beyond the scope of the risk assessment of this simple and generic substation example.

We start by listing all the feared events that we may think of as potentially harmful.

The automation system under consideration is rather simple and thus the list of feared events is relatively short. With the help of an electrotechnical expert and based on the literature addressing cyber security aspects of IEC 61850 power system automation, we brainstormed several undesirable events. As we realized that some were actually steps in possible scenarios leading to other events, we restricted the identified risks to the following short list:

- inappropriate breaker operation (tripping or closing),
- blocking of a breaker action (tripping or closing),
- blocking of a protection function,
- loss of information,
- modification of information,
- false information injection.

3.2.3 Risk analysis

After this rough risk identification comes the sorting. Considering the functional zones of the substation (D01, E01, E02, E03 and the global substation) one at a time, we try to establish potential links between them, to describe the consequences of the occurence of a single of these events or a combination of them according to the state of operation. We also try to trace back feared events to potential malicious cyber causes. This approach, both deductive (describe consequences) and inductive (understand causes), helps us build our risk analysis. Concretely, we confront this list to each of the operational zones to analyze what scenarios may cause these events and what consequences they may have.

Zone D01

D01 zone is a critical section of the substation since it is the power flow "entry point" of the substation. CB_{D01} is normally closed. In *normal state of operation*, inappropriate tripping of it would mean loss of power to the whole substation. It may be caused by a spoofing message from the local HMI or the remote supervision center carrying a trip signal to IED_{D01bis}. As client-server links, they can be the target of a MITM attack, or the gateway relaying messages with the remote supervision center may be compromised. Or the trip signal may be carried by a false GOOSE message masquerading either IED_{D01} whose protection functions rely on the tripping of CB_{D01}, or IED_{E02} as it is involved in the Distance protection function and CB_{E02} Backup protection. It also may be tripped by a GOOSE message genuinely published by IED_{D01} or IED_{E02} deceived by false measurement SV messages from E02 metering unit, thus inappropriately triggering Overload or Transformer differential protection, or Distance protection respectively.

In a state of *emergency/fault* requiring CB_{D01} to trip to fulfill the Distance or Backup protection, an intruder may exploit the broadcast communication the same way but this time in order to prevent protection from operating properly. False GOOSE messages may indicate normal state of CB_{E02} while the Backup protection shall be active and GOOSE messages with CB_{D01} trip command shall be published, thus it keeps CB_{D01} closed. If the GOOSE application being compromised is the one involved in the Distance protection, the false GOOSE message indicates a normal state while a fault is actually occuring. Or tampering with mesurements transferred through SV communication may let IED_{E02} unaware of a fault on the line and thus it will not launch the Distance protection function. Therefore, the fault would not be isolated and may propagate in the substation and even outside of it, causing damage to equipment.

In *emergency/fault state of operation*, meaning CB_{D01} has genuinely tripped, IED_{D01} or IED_{D01bis} must report to local HMI and supervision center (Protection signaling function). If this information was lost or altered, local and remote supervisions may not be aware of CB_{D01} state as open in answer to a fault. It may lead to an erroneous system state estimation and wrong decisions by operators: if they believe the system in a normal state, there is no reason to take actions of any kind. This may result in fault propagation (perhaps outside the substation boundaries).

Zone E02

In *normal state of operation*, CB_{E02} is normally closed. Tripping would cause loss of energy for the busbar and outgoing feeders. As previously it can be launched by a MITM attack mimicking a message sent by local or remote supervision. It also may be tripped by a signal carried by a false GOOSE message injected by an intruder and pretending to be published by IED_{E01} or IED_{E03} to simulate a Differential protection, or to trigger Backup protection for a fake failure of CB_{E01} or CB_{E03} respectively.

As for CB_{D01}, in a state of *emergency/fault* requiring CB_{E02} to trip to fulfill the Distance or Backup protection, an intruder may inject false GOOSE messages indicating normal state of CB_{E01} or CB_{E03} whilst in failure state in order to prevent protection from operating properly. CB_{E02} stays closed, which may mean fault propagation and equipment damage.

Zones E01 and E03

In *normal state*, at least one of the two circuit breakers CB_{E01} and CB_{E03} is closed, possibly both, depending on the global system state (beyond the scope of the system we study) and dispatching operations being run. Both these circuit breakers may be opened by a false command signal through the client-server communication (MITM on links with local or remote supervision). It would cause the loss of power to feeder E01 or feeder E03 respectively but that would not be really critical since the power system is operated in such a manner as to compensate for the loss of one equipment. However if both were lost together, consequences would be more critical. We cannot elaborate further on them as we do not know what the grid is outside the boundary of our system.

Interlocking is supposed to prevent such an action. But since this function relies on GOOSE communication from IED_{E02} to IED_{E01} and IED_{E03}, from IED_{E01} to IED_{E03} and from IED_{E03} to IED_{E01}, an intruder may disturb or hinder its operation by injecting false GOOSE messages.

In *emergency/fault state of operation*, a first level (as opposed to Backup protection) being genuinely launched, an intruder may trigger the Backup protection of the corresponding CB by injecting a false GOOSE message. The consequence would be the loss of a wider portion of the grid than strictly necessary to isolate the fault. Thus both outgoing feeders may be lost, while only one was in a faulty state if its Backup protection is inadequately triggered. And if the Backup protection of its backup CB (CB_{E02}) is also launched, that is the whole substation that would be lost instead of just a single outgoing feeder.

The substation level

Information is a critical asset to the correct and safe operation of the substation, whatever the state of operation. We mentioned how an intruder can alterate information flows for a specific purpose. Loss of information is another way to disturb the substation operation or potentially cause damage. It may be the option chosen by an intruder not as knowledgeable than in the previous scenarios. The two options to cause loss of information are:

1. MITM attack may allow the intruder to drop packets to the substation level HMI or to the remote control center through the substation gateway and feed them with replay of previously intercepted message. Recipients will have a biased awareness of the system state.

2. DoS is more radical and less subtle. By overflooding the communication network, the intruder prevents all messages to reach their destination on time. System state is no longer observable to any of the applications, including supervisions but all the IEDs as well. It means the loss of all communication-based functions.

Regarding the *post fault state of operation*, manipulation and loss of data prevents the collection of fault information. MITM attack on the client-server links or a DoS attack of the network would hinder reporting by field devices to supervision.

Another risk that may apply to several substation components is modification of configuration files loaded in IEDs to change protection thresholds or communication settings, if the service is available from remote supervision through client-server communication.

Tampering with measurements necessary to the Synchro-check function resulting in inappropriate control signals may result in connecting two portions of the network not in phase/not at the same frequency or voltage level may cause a breakdown, electrical arc.

3.2.4 Conclusion

All the scenarios described above are based on the compromising of the integrity of communications, except the DoS technique that compromises the information availability. The specific attack targeting system configuration files compromises confidentiality of their content. Confidentiality may also be compromised in a reconnaissance attack, whose objective is to gather information prior the actual end-goal attack or perhaps to blackmail the utility managers. The scenarios affecting the operating of protection functions challenge the system safety, availability and reliability (as defined in section 1.3.6).

Several risk assessment methodologies (like SPARKS method described in section 2.5) advocate the use of event tree or attack graph to cast the whole chain of events, highlighting their dependencies that may be limiting or aggravating factors. In this simple and generic case, such a representation of this risk analysis outcomes may look as Figure 3.8. This graph is the synthesis of the detailed attack scenarios described previously.

The first feared event is inappropriate tripping of a circuit breaker while in normal state of operation, resulting in power loss of the substation, partially or as a whole. Availability and reliability are compromised. Possible causes encompass, regarding only communication:

- Man-in-the-middle attack on MMS communication with local or remote communi-

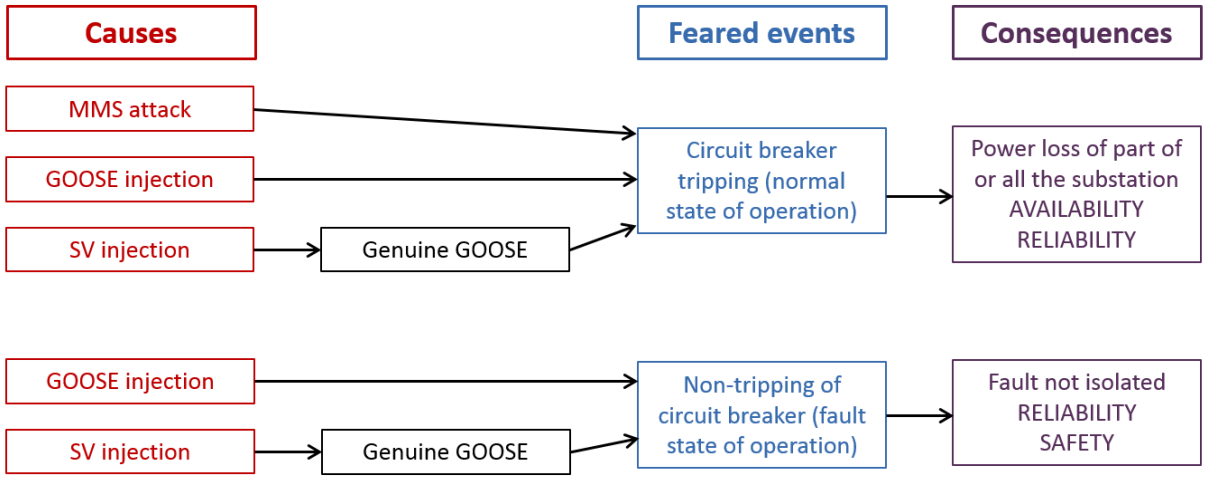

Figure 3.8: Attack graph example

cation.

- GOOSE injection attack.
- SV injection attack: fraudulent message is processed by targeted IED, which genuinely generates a GOOSE message launching circuit breaker tripping.

GOOSE or SV injection attacks may result in totally different consequences in another state of operation. Thus in an emergency state of operation, where tripping is expected, they could prevent it and result in spreading of an electrical fault thus compromising reliability and safety.

While not essential for one to grab the global picture of operational cyber risk of our simple case study, for more sophisticated applications and larger and more complex systems, such tools are helpful to strengthen the analysis process and formalize the outcomes.

Last step of risk assessment is risk evaluation, that is linking occurence probabilty and consequence severity. In cyber security, we rather talk of attack cost than probability: the more means the attack requires, the less probable. The present analysis wants to stay generic and evaluating is neither relevant nor feasible as this fictitious case study lacks environmental information. Further knowledge about system configuration, applications implementation, hardware specification would be necessary. However, evaluating whether the attacks are feasible and under what conditions is valuable. Especially experimentation may be useful to precisely characterize risks and complete the task of risk assessment. The considered attacks are mentioned in literature (see section 4.1) and we demonstrated the feasability of attacks on GOOSE communication, experiment is presented in section 4.1.3.

3.3 The IEC 61850 data object model

The first main contribution of the IEC 61850 standard is the definition of a data object modeling for functions and services with a rigorous naming convention to provide interoperability.

3.3.1 Object oriented information structure

The modeling approach of the IEC 61850 standard is to decompose the application functions into the smallest entities, defined from the application angle. These entities are called Logical Nodes (LN). Several LNs build a Logical Device (LD), which is implemented in one Physical Device (PD), the highest degree of the data object model introduced by the standard. These data elements have attributes and if need be sub-attributes, etc... For Data Attributes, Data and LNs, the IEC 61850 standard includes a catalog of objects considered as necessary for protection and control/monitoring of the power grid.

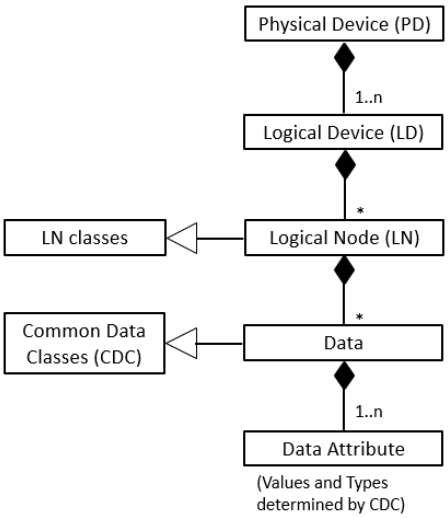

Figure 3.9: IEC 61850 Data Object Modeling

It is possible to create new objects to develop new functionalities as explained further (see section 3.3.3). Naming of data objects is standardized as well, especially for LNs that are the core components of IEC 61850 model. This is supposed to make interoperability possible. IEC identified thirteen groups of LNs according to their main purpose specific to power utility common functionalities (Protection “P”, Control “C”, Interfacing “I”, Switchgear “X”... cf. IEC 61850:7.4 for the whole list [71]). Each group defines mandatory and optional data attributes for LN description. A LN’s name is thus built as follows: Group designator / Three-letter abbreviation of function / Instance ID. For instance, MMXU1 belongs to “Measurement” group (M) and designates an operative measurement unit (MXU), first instance (1). It can possibly have an optional prefix.

When enriching IEC 61850 data object model with LN types, vendors do not always strictly follow naming rules. To make sure names of new objects are explicit enough, we do not follow this conventional naming either in this dissertation.

3.3.2 PICOM (Piece of Information for COMunication)

The second contribution of the IEC 61850 standard is about communication in terms of syntax, semantics and performance (i.e. a protocol). Communication architecture proposed by the IEC 61850 standard is described in section 3.1.1. Here, we will introduce the logical link concept, that is an abstract communication connection at the information model level.

In the IEC 61850 communication model, information is exchanged between LNs. Logical connections are defined between a source LN and a sink LN to transfer a specific data element. Such connections are fully described by PICOMs, Pieces of Information for COMmunication. A Logical Node can be considered fully specified only when related PICOMs are defined. The PICOM concept was introduced by the CIGRÉ working group 34.03 as *"a given data element or block of data on a given logical path with a given communication attribute"* [29]. Glossary part of IEC 61850 standard completes this definition as follows: *"PICOM is a description of an information transfer on a given logical connection with given communication attributes between two logical nodes. It also contains the information to be transmitted and required attributes for example performance. It does not represent the actual structure or format of the data that is transmitted over the communication network."* [65]. In the global IEC 61850 model, PICOM can be considered

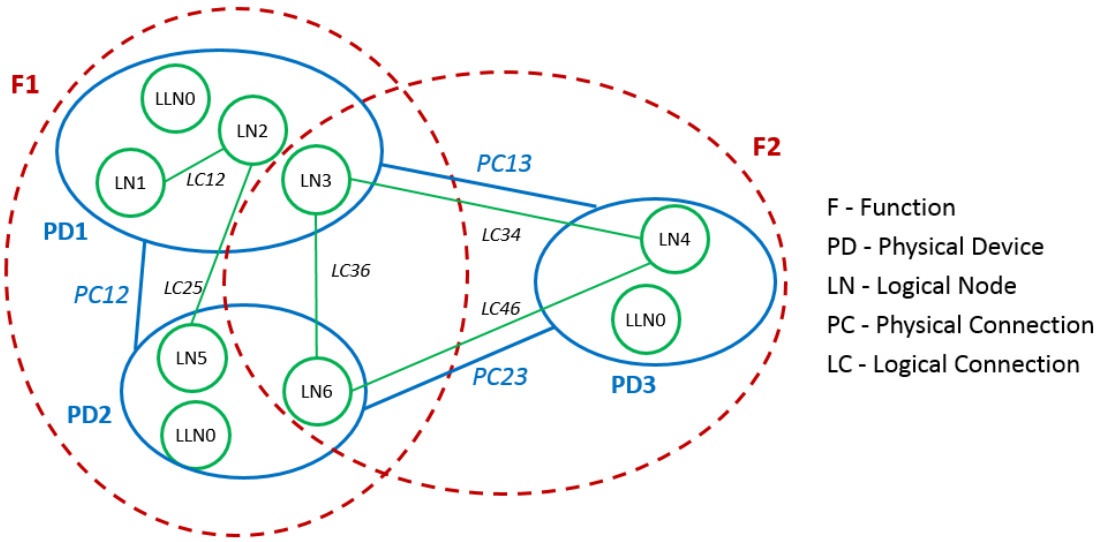

Figure 3.10: IEC 61850 Logical link concept

as a class for information transfer between a source LN and a sink LN, namely a logical connection as illustrated in Figure 3.10 [74]. LNs are the constitutive bricks of functions.

Logical node LLN0 is a "device" LN containing all generic information about the device and that does not refer to any function but that may be required to be communicated to fulfill the function. Such information includes IED's nameplate or self-supervision. Logical connections rely on physical connections between devices.

PICOM components are either characterizing the information content: data name, data type, data length and value, or the communication requirements: source and sink names, priority of transmission, data integrity, method or cause for transmission, transfer time requirements. Some of those attributes must be covered by any message when others are to be considered at configuration time or for data flow calculations. IEC 61850 standard defines seven message types based on a grouping of the performance related PICOM attributes. One important criterion for the choice of message type is end-to-end transfer time, that is time elapsed between data emission from sending function to data reception by receiving function, including times of communication processors of both source and sink devices and the network transfer time. Message types may be subdivided into perforamnce classes: control and protection PICOM messages may be of performance classe P1 – typically a distribution bay with low requirements, P2 – typically a transmission bay, or P3 – typically a transmission bay with top performance features [74]. Hence, the PICOM message types are as follows:

- *Type 1 - Fast messages*: It usually contains simple binary code for data, a command or a simple message. Receiving IED is supposed to react immediately. Trip commands having very demanding time requirements, a "Trip" sub-type is defined with stringent total transmission time: 10ms for performance class P1 and 3ms for performance classes P2/3. Sub-type "Others" concerns important messages other than trip commands. Total transmission time shall be less than 100ms for performance class P1 and 20ms for performance classes P2/3.
- *Type 2 - Medium speed messages*: As fast messages, they are simple messages whose originating time is important but transmission time is less critical. Total transmission time shall be 100ms maximum.
- *Type 3 - Low speed messages*: They usually are more complex messages for e.g. slow speed functions, transmission of reports and logs, reading or setting system data. Total transfer time shall be less than 500ms.
- *Type 4 - Raw data messages*: They are aimed for digital measurement units to send voltage and current values. Total transmission time shall be less than 10ms for performance class P1 and 3ms for performance classes P2/3.
- *Type 5 - File transfer functions*: They are typically used to transfer large files for recording, settings, etc. Their bit length is generally equal to or greater to 512 bits. No limit is specified as total transmission time is not critical. It may be greater than 1000ms.
- *Type 6 - Time synchronization*: Requirements are given in terms of time accuracy in the global system. For further detail, see IEC 61850-5 [74].

- *Type 7 - Command messages with access control*: They are mainly thought as messages from a supervision center outside the system. Thus, they are similar to Type 3 messages enhanced with access control. The standard acknowledges that in some cases Type 1 messages may be required. However, as discussed in section 1.2.1, IEC 61850 standard is unclear about security mechanisms for time-constrained communications: authentication is recommended but does not precise any further detail about it.

Table 3.1: PICOM types and associated transmission times

Message type		Performance class	Transmission time (ms)
Type 1 - Fast messages	Trip	P1	10ms
		P2/P3	3ms
	Others	P1	100ms
		P2/P3	20ms
Type 2 - Medium speed messages			100ms
Type 3 - Low speed messages			500ms
Type 4 - Raw data messages		P1	10ms
		P2/P3	3ms
Type 5 - File transfer functions			1000ms or greater
Type 6 - Time synchronization		*requirements in terms of accuracy*	
Type 7 - Command messages with access control		*Type 3 or Type 1 messages*	

Choosing adequate message types will improve global performance of the automation system. PICOMs are related to the application layer and do not represent the actual format and structure of the data over the physical network. This is the abstract level of the communication process in an IEC 61850 environment and it is totally independent of its real implementation.

3.3.3 IEC 61850 data model extension rules

As stated in the introductory part of the IEC 61850 standard, "*the purpose of the standard is neither to standardise (nor limit in any way) the functions involved in substation operation nor their allocation within the SAS*" [73]. And hence, IEC 61850 standard lets one define new functions but this must be done following some rules to ensure the interoperability between all the components of the global system [142]. An IEC 61850 function is then defined by:

1. *Task description*: a formal description of the function task and its context of execution.
2. *Starting criteria*: the reason of the function launching.

3. *Result / impact*: output of the function.

4. *Performance*: total requested response time to guarantee, process time, possibly also accuracy of synchronization or other criteria.

5. *Function decomposition*: decomposition into subfunctions, that is into LNs.

6. *Interactions with other functions*: data exchanged with other functions formalized by description of PICOMs.

Function decomposition results into its LN description, standard LNs or new ones. Data elements given in the IEC 61850 catalogs should be used. If none is appropriate enough, the model can be enriched with new data elements required to fulfill the task of the function following a stepwise extension process [75].

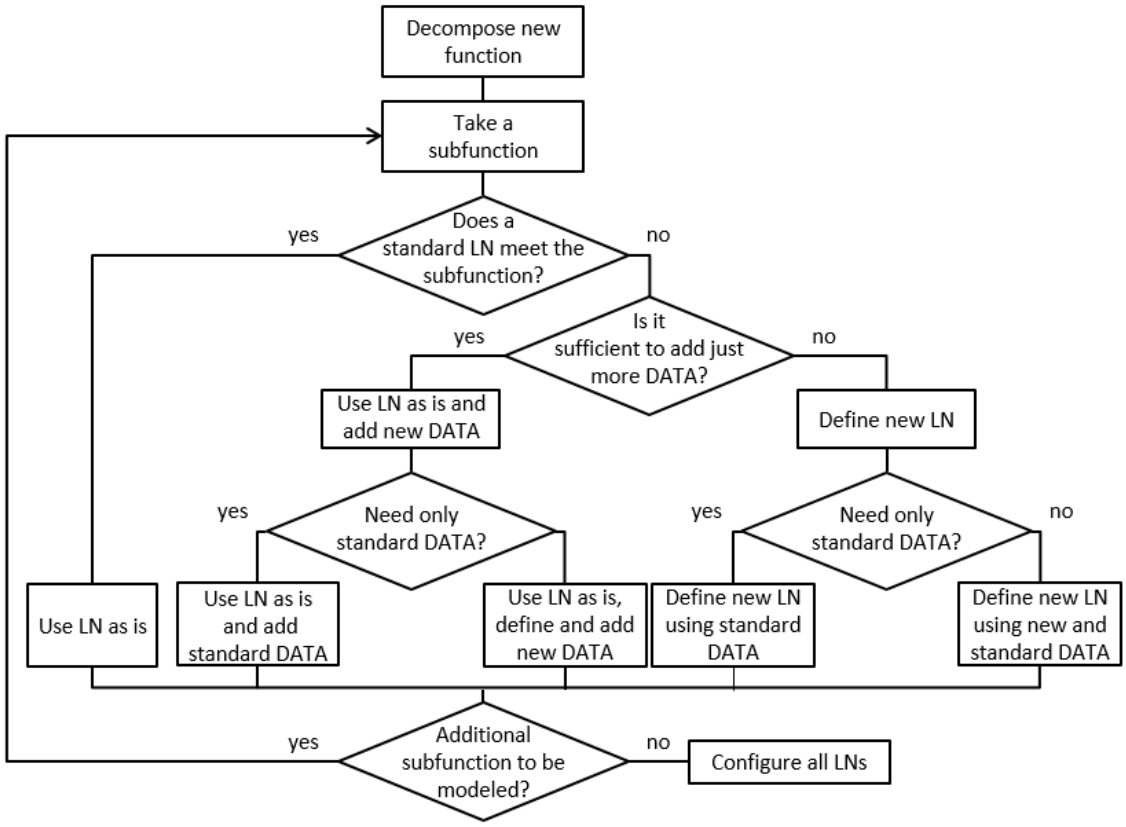

Figure 3.11: IEC 61850 function extension flowchart

Figure 3.11 shows the workflow of IEC 61850 information model extension. To build a new functionality, the first step is to decompose the function into subfunctions. If a LN of the standard meets the characteristics of the subfunction, use it as is. Otherwise, if adding existing or new data to an existing LN is not sufficient, then create a new LN. The same flow can be run to define all the data of a LN along with their existing and new data attributes.

A LN specification is formalized into a brief task description and a table inherited from the "Common LN Class" which is composed of Data (mandatory and optional), Data Sets, Control Blocks (for configuration of setting, reporting, logging and communication) and Services (GetLogicalNodeDirectory, GetAllDataValues). The Data part classifies the LN data objects into six categories: Common LN information, Status information, Settings, Measured values, Metered values (computed by the LN itself), Controls. A LN may have data attributes of one, many or all of these categories.

Describing logical connections completes the functionality specification as PICOMs provide information about semantics, type of data, performance requirements and logical connection path.

3.3.4 IEC 61850 security-related information material

As demonstrated by the extension workflow described above, it is necessary to be aware of security material available in order to expand the IEC 61850 data object model with cyber security objects. The IEC 61850 standard defines a few security-related functions:

- **Access security management**, which purpose is to monitor human access from HMI users to operational functions.
- **System Security Management** function. Its definition is supposed to make it generic and global thus "*monitoring and providing all activities regarding security violations*". But the second sentence of its description states that it "allows the control and supervision of the security of the system against unauthorized access and loss of activity" [74]. It seems that security was initially thought regarding mainly authorization and service privileges, "loss of activity" may allude to DoS. However this function decomposition into logical nodes (LN) is quite generic: it is composed of the device LN (LLN0), interface LNs ("Human Machine Interface" IHMI, "Remote Control Interface" ITCI and "Remote Monitoring Interface" ITMI), "Generic Security Application" LN (GSAL) and "Alarm Handling" LN (CALH)[1]. Result shall be awareness of the security level at any time plus a possible immediate blocking of sensitive functions. From our point of view, blocking shall be considered very carefully as it must not hinder availability of the system.
- **Alarm management** is responsible for raising alarms to an operator and letting him deal with them (acknowledge and clear them). Data may be related to the process state or the automation system itself. An alarm has many attributes: source, cause, alarm acknowledged or not, urgency and gravity. Alarms may not be related exclusively with access security violations but also with compromising of dependability, physical security, etc, anything that "should be taken into consideration by the operator".

[1] Details about LNs IHMI, ITCI, ITMI and CALH can be found in Table C.1. For a description of LN GSAL, refer to Appendix B or to IEC 61850-7.4 [71].

Two other functions are worth mentioning, although they are not exclusively for security. The first one is "Network management", whose basic task is network node identification. Status of all physical devices and links and all logical nodes and links is known, as well as data traffic between all links. The second one is "Event management", which continually collects, processes and records all events, including systems state changes, process state changes and control actions.

Considering these definitions, an option for developing further cyber security functionalities would then be to define new sub-functions to complete "System Security Management" LN decomposition. However, it seems to us that cyber security measures are very diverse and keeping them separate would make the model more comprehensive.

In this work we propose to specify a security function dedicated to anomaly detection in a communication system. This is a passive operation which means that this functionality does not impact the system directly but only returns alarms and information about the system behavior. There also exist active security functions, which have to take action according to the situation, such as intrusion protection. The security material of the IEC 61850 information model could be extended with such other functions. Considering that further security extensions are possible, we can imagine two options on how to deal with the "System Security Management" function. It can either be kept as a global security management function monitoring all security specific sub-functions, doing correlations, dealing with reporting to SIEM (Security Information and Event Management), etc. However this is not how IEC 61850 data model is built: there is no hierarchy between functions, only interactions. Or, second option, it can be used as a basis for every security function, that is adding necessary LNs, possibly replacing inappropriate ones, to "System Security Management " specification for designing these security functions. Structure of "System Security Management" function is traditional: interface nodes, an application node and an alarm node. We keep this basic skeleton for our cyber security function just replacing the generic application node GSAL by new nodes specific to anomaly detection.

3.4 A complementary set of IEC 61850 data objects for intrusion detection

Previous section was meant to give a substantial introduction to the concepts of IEC 61850 data object model relevant to our purpose, which is to build an intrusion detection function consistant with this model. We explained the information structure, described the workflow to consistently specify new data objects and outlined existing security material. Next step is to define the intrusion detection function to be added to the IEC 61850 information model and specify all necesary data objects as to fit this model characteristics.

We focus on GOOSE protocol as our risk assessment of a generic IEC 61850 substation (section 3.2) stressed that broadcast communications are particularly critical to safety and availability of substation mission. Also, these communications (GOOSE and SV)

are vulnerable as they cannot support neither encryption nor signature (see discussion in section 1.3.3).

3.4.1 Definition of a network-based anomaly detection function

The concept of intrusion detection is well known since the definition of an intrusion detection model by Denning in 1986 [31]. In the context of IEC 61850 automation system, the model must be defined with a good understanding of the standard specifications and with respect to the extension procedure. Figure 3.12 shows our network-based anomaly detection function model as an independent device in an IEC 61850 architecture. LNs of the model are associated to the IDS modules for a clear picture of the function. They will be explained in section 3.4.2. Anomaly detection model as depicted in Figure 3.12 is

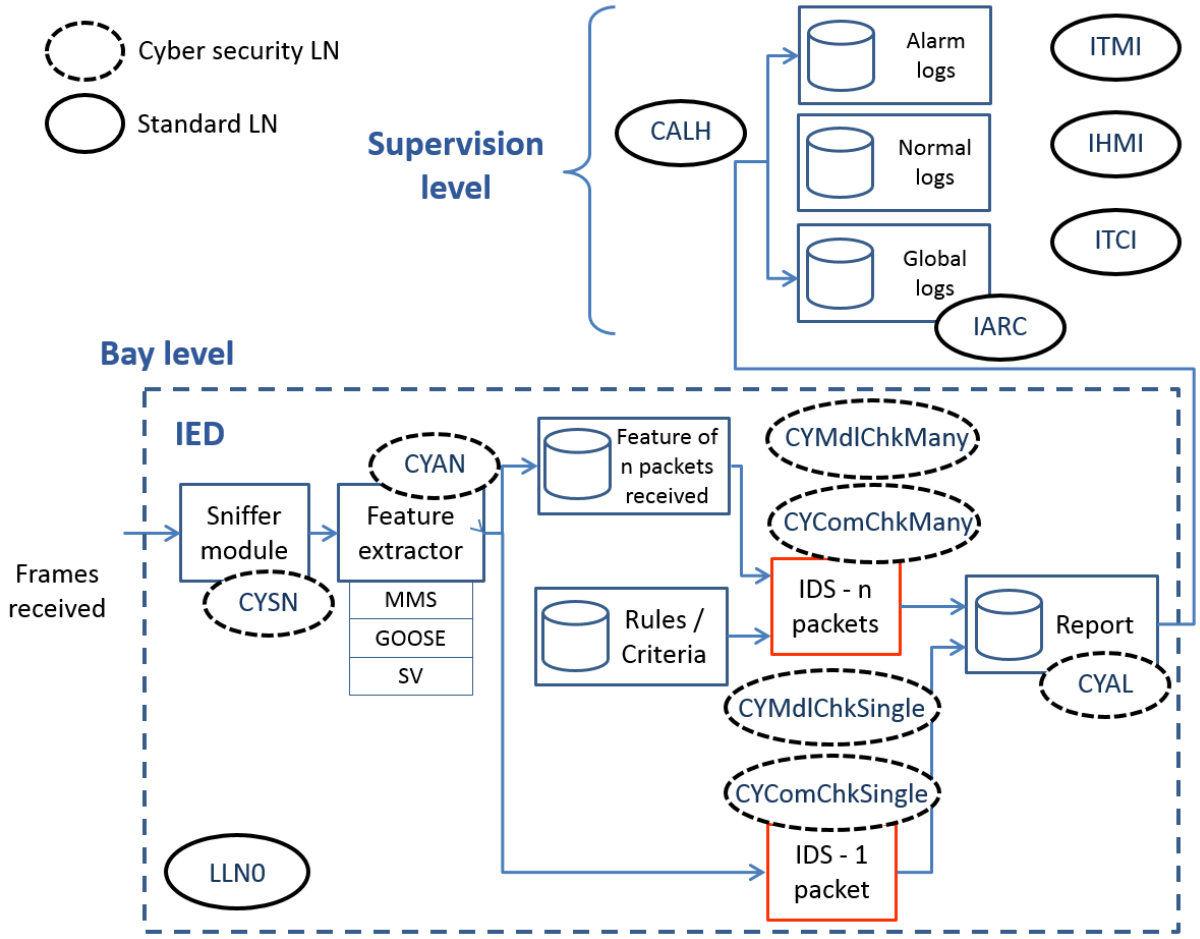

Figure 3.12: Intrusion detection model as an autonomous IED

classically structured. First step of the detection process is gathering data, in the case of a NIDS such data are the monitored traffic. Relevant features must then be extracted and verified. Verification concerns protocol- and behavior-based criteria. Some of them can be controlled on the basis of a single packet (syntax, semantics, values of certain fields...)

but others require comparison between two or more packets (reception frequency, counters incrementing...). Details are given in section 3.4.2. The IDS must generate alarms when anomalies are detected and publish logs to keep track of its analyses. Such logs may then be used by complementary cyber security monitors.

Following the information model extension rules, we defined our "Network-based Anomaly Detection" function:

- **Task:** The anomaly detection function allows detecting anomalies in the system behavior. It analyzes input frames for feature extraction to check them for compliance with the communication model (syntax and semantics, and exchanges) and the system model. Results of this analysis are recorded and alarms are generated in case of deviance from the normality that has been specified.
- **Starting criteria:** Reception of a frame.
- **Result or impact:** Results of the analysis are stored in logs and alarms are generated if need be, that is when an anomaly is detected.
- **Performance:** As written in section 1.3, availability is of the upmost importance when regarding security and dependability objectives of ICS. This is especially true for systems dealing with electrical protection. Thus, accuracy of the detection function is a major performance criterion as false positive are intolerable (see section 1.4.1.1). Another performance criterion is processing time (or decision time). As alarms are raised when anomalies are detected, it seems relevant to consider that the faster they are processed, the better. Concretely, we can refer to the performance requirement of the standard function "System security management": "*The security supervision function should be as comprehensive as possible. In case of breached security, blocking should be issued immediately (10 ms). Any alarm should be provided within the human operator response time (about 1 s).*" [74]. And production and sending of analysis logs may take more time.
- **Function decomposition:** As given in Figure 3.13 and explained in next section 3.4.2.
- **Interactions with other functions:** As mentioned in previous section 3.3.4, IEC 61850 catalog of data objects counts a few functions more or less correlated to cyber security. This function will be linked to "Alarm management". It may also cooperate with "Network management", "Event management", "System security management", "Access security management" for contextualization of its analysis. As we already wrote, it seems to us that a SIEM, or an equivalent monitoring function, shall deal with the correlation of cyber security information from all relevant functions, as anomaly detection is just one among many cyber security measures and connecting their outcomes may help get the global picture. The role of a SIEM is to collect data generated by all IDS modules / functions and by other relevant sources (seucrity related such as firewalls, or network architecture-relates such as switches...),

make some correlation, formalize information to be comprehensively displayed to the operator, and deal with historians for further analysis.

3.4.2 Function decomposition

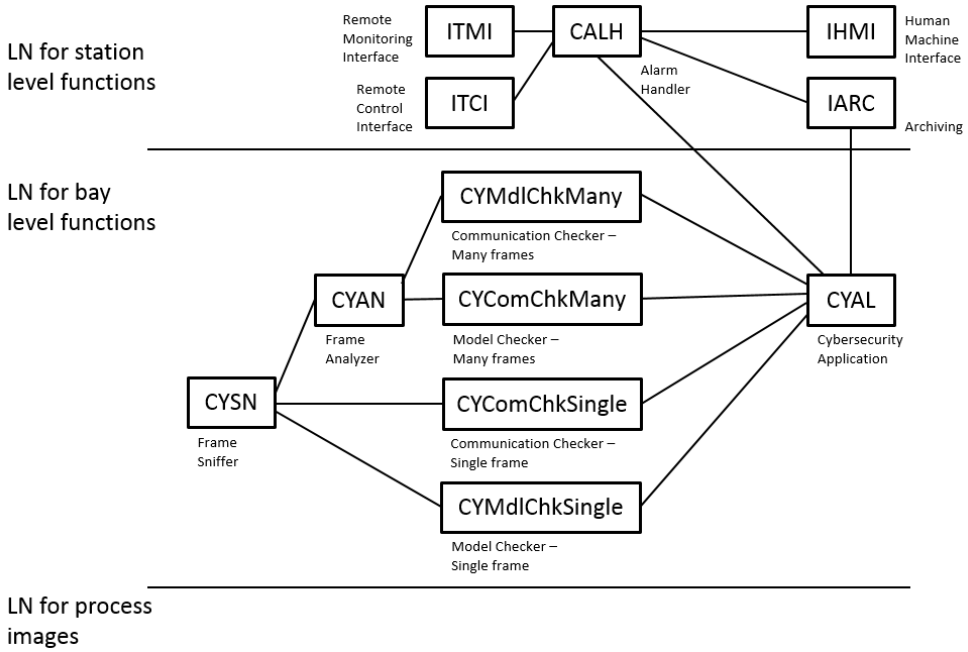

Figure 3.13: Decomposition of the anomaly detection function into interacting LNs on the different SAS levels

The "Anomaly Detection" function is built from many LNs. Some of them are introduced by the standard. The ones which designators start with "CY" were made up for our cyber security purpose. Again, such a group designator "CY" for Cyber security does not comply with the standard naming convention as it shall be just one letter. None of the remaining letters seemed pertinent to us, so for sake of clarity, we chose a two-letter designator to explain our work. Same motivation led us to make checkers names as explicit as possible, thus making them longer than the conventional four-letter length and using capital and lower case letters. These choices concern this dissertation only and may be reconsidered later.

The generic function decomposition is given in Figure 3.13. The "Frame Sniffer CYSN" LN intercepts all messages from the network, recording their content along with reception information. Relevant data are transferred to single frame checker modules, "Communication Checker – Single Frame CYComChkSingle" and "Model Checker – Single Frame CYMdlChkSingle" for communication- and model-related criteria respectively. In parallel, sniffed frames and their interception information are sent to a "Frame Analyzer CYAN" LN whose role is to compute relevant features for a multiple-packet inspection. This cross-packet inspection is realized by the "Communication Checker – Many Frames CYComChkMany" and "Model Checker – Many Frames CYMdlChkMany".

Communication checkers are responsible for verifying conformance with protocol specifications and communication settings of the automation system. We focus on GOOSE protocol for describing the proposed function decomposition, the procedure for SV would be similar, considering specificities of this protocol. Details about GOOSE protocol and IEC 61850 communication architecture in general can be found in section 3.1.2. Let us quickly sketch the main GOOSE protocol specificities here. GOOSE protocol operates on the process bus and is used for information exchange between bay level devices. GOOSE frame structure is standardized (ISO/IEC8802-3 Ethertype 0x88b8), given in the IEC61850 standard [72] and recalled in section 3.1.2.1. The GOOSE transmission mechanism is also to be considered. GOOSE protocol is mapped on the Ethernet link layer which means it is multicast, it is a publisher – subscriber messaging procedure and there is no acknowledgment. So to make sure information has reached its destination, GOOSE has a specific transmission mechanism. When a variable from the defined data set has changed (event), a GOOSE message is generated. The same GOOSE message is sent periodically at a high frequency first and then at a slower pace until back to stable conditions. State and sequence counters are used to keep track of message succession. Single-packet analysis is thus concerned with the following parameters: packet structure, source address, destination address, message identifiers (Application Identifier APPID, GOOSE Control Block Reference GoCBRef and GOOSE Identifier GoID), number of data entries... Significant criteria for multiple-packet verification are message succession counters (Sequence and State Numbers SqNum, StNum), transmission time, reception time, frequency of messages...

Model checkers must verify application-related criteria such as control operation or value range for the single-frame module and correlation of successive commands or state information for the multiple-frame one.

Results of these analyses are sent to the "Cyber security Application CYAL" LN for aggregating and formatting before being forwarded to the substation-level standard-defined LNs related to interfacing and archiving as shown in Figure 3.13. This function decomposition distinguishes the three operation levels of the SAS. Our cyber security function has no LN for process image since it deals with communication network and does not act on the process as it is a passive function. Links between LNs are for logical connections corresponding to one or many PICOMs. Urgent alarm PICOMs do not appear on this diagram: if one of the checking LNs detects a serious anomaly it should send an alarm to an interfacing node for operator awareness or to a prevention system. It also transmits information to "CYAL" for reporting. Further analysis may be run by another dedicated function, comparable to a SIEM. One can argue that checkers verifying a single or many frames may have been kept as one module, that the "Frame Analyzer" LN is not necessary. We made such choices for sake of simplicity. Let us highlight that the proposed specification, as well as any standard LN specification, does not directly involve implementation choices which are up to the manufacturer. Conformance of the programming with the information model is certified afterward.

3.4.3 Extension to the IEC 61850 information objects catalog for network-based anomaly detection

Following the flow of the extension process (see section3.3.3), we have fully specified our anomaly detection function covering all the data layers from LNs to basic types and PICOMs.

LN specification: CYComChkSgl

Following the template of LN tables given in the IEC 61850 standard, Table 3.2 gives data attributes of "CYComChkSgl". "Communication Checker – Single Frame" specification is identical to the one of standard LN devoted to security, "Generic security application - GSAL" [71], except for the status information data attributes. For the sake of clarity, the data objects that we have created for the purpose of our cyber security function are in italic type. These ones correspond to the security criteria to check, which are related to authorization, access control, service privilege and inactive associations in GSAL case. LN "Communication Checker – Single Frame" is composed of the mandatory Common LN information data attributes (Mode, Behavior, Health, Name plate) and an optional one (Operation counter resetable), and Status data attributes that we created. Attribute type is chosen among the Common Data Classes (CDC) defined in [70] or may be created if none is relevant. Such CDCs are templates of data, with a list of attributes and their specificities. CDC "Security Violation Counting" (SEC) seems to suit most of our cyber security data objects, details are given in Appendix B. LN textual description was given in section 3.4.2 about function decomposition and is recalled in Table C.1. This node shall be used to check conformance of a single frame with protocol specification and communication configuration.

This LN shall monitor violations of the message integrity regarding protocol specification and the automation system's communication configuration. Protocol specification violations may concern the frame structure as defined in international standards. For GOOSE protocol, it would mean answering questions such as: are all the protocol fields present in the frame structure?, are they in the right order?, are their values in the defined range?, are fields coherent with each other (when relevant, e.g. field "number of data entries" and the actual number of data entries carried by the frame)?, etc. Communication configuration violations are about who is supposed to send and receive what information. Still for GOOSE protocol, questions to answer would be: do the source and destination addresses exist in the configuration?, is the association of a given source address and a given destination address actually configured?, are message's identifiers defined and are they all coherent in a single frame (APPID, GoCBRef, DatSet, GoID)?, are the value of static configuration-related fields as defined in the communication configuration (e.g. time allowed to live)?, etc. Each of the criteria to check makes a data element of the LN so the security violation is clearly identified when occuring. They all are of type "Security violation counting". This common data class has two attributes that can be used to give

further detail about the violation: “d - Textual description of the data” and “addInfo - Additional information that may give further clarification as to the last detected violation” (see Appendix B).

PICOM specification: PICOMs with source LN CYComChkSgl

PICOMs are categorized according to their functional purpose: Operational data transfer, Parameter transfer, Informative transfer [29]. This last category is for “*data communicated for post-mortem display, monitoring, archiving, statistics*” and would fit analysis results sent by checking modules. Tables 3.3 and 3.4 give the definitions of PICOMs with source LN CYComChkSgl as outlined in section 3.3.2. All alarm PICOM have the same PICOM profile “Event/Alarm“ 3.3.

Table 3.2: CyComChkSgl Class Table

CYComChkSgl (Communication Checker - Single Frame)			
Data Attr Name	Attribute Type	Explanation	M/O
Common Logical Node Information			
Mod	INC (Controllable integer status)	Mandatory Data from Common LN class	M
Beh	INS (Integer Status)		M
Health	INS		M
NamPlt	LPL (LN name plate)		M
OpCntRs	INC	Resetable Security Violation Counter	M
Controls			
NumCntRs	INC	Number of counter resets	M
Status Information			
StructAlm	SEC (Security violation counting)	*Protocol criterion / Inconsistency of the frame structure as defined in the standard.*	M
ProtValAlm	SEC (Security violation counting)	*Protocol criterion / Inconsistency of the value of specific protocol-related field(s): wrong value or out of the standardized range.*	M
CoValAlm	SEC	*Protocol criterion / Inconsistency of certain fields' values with each other.*	M
AddrAlm	SEC	*Configuration criterion / Source and/or destination addresses are not configured in the system networks, or no communication link associating these source and destination addresses is configured.*	M
IdAlm	SEC	*Configuration criterion / Communication application identifiers either are not configured or are not consistent with each other.*	M
ConfValAlm	SEC	*Configuration criterion / Inconsistency of the value of specific system configuration-related field(s): wrong value or out of the acceptable range.*	M
Services			
GetLogicalNodeDirectory			
GetAllDataValues			

Table 3.3: PICOM "Event/Alarm" of source LN CyComChkSgl

Name	*Message structure alarm, Protocol value alarm, Protocol co-values alarm, Address alarm, ID alarm, Configuration value alarm*
Source and sink	As given in Table C.6.
Type of data	Binary data. IEC 61850-5 recommends type 3 (Low speed message) for alarm handling and type 2 (Medium speed mesage) for automatics. We would recommend to use type 2 or even type 1 (Fast message) for transferring alarms about communication compromising according to the criterion being compromised and to the further use of the alarm: if it is meant to launch some automatic protective action (beyond the scope of anomaly detection), the faster the information is transferred the better, and then a type 1 message would fit this need beeing simple (1 bit) and fast (1 to 10ms); if its purpose is to inform an operator or to feed a security management system collecting and processing information from many monitored point throughout the system, then a type 2 or 3 message would probably be more suitable (a few bits for 1 to 100ms).
Length of data	One bit if type 1 message, a few bytes if type 2 or 3.
Time tagged data	Yes
Cause of transmission	Spontaneous
State of operation	all states of operation as defined by CIGRE and given in section 3.2.1
Priority of transmission	High
Data integrity	High
Time requirements	Overall transfer time of 1 to 10ms if type 1, 10 to 1000ms (which corresponds to operator reaction time) if type 2 or 3.

Table 3.4: PICOM "Analysis report" of source LN CyComChkSgl

Name	*Diagnostic Data/Analysis report*
Source and sink	As given in Table 3.5.
Type of data	Type 5 is the most appropriate for transferring results of he analysis run by CYComChkSgl (up to 512 bits, overall transfer time of 1000 to 5000ms).
Length of data	Up to 512 bits.
Time tagged data	Yes
Cause of transmission	Spontaneous and request.
State of operation	all states of operation
Priority of transmission	Normal or high depending on time criticality of further processing if available.
Data integrity	High
Time requirements	Overall transfer time of 1000 to 5000 ms depending on time criticality ot further pricessing if available.

Table 3.5: PICOMS of source LN CyComChkSgl

LN	**PICOM Name**	**Source**	**Sink 1**	**Sink 2**	**Sink 3**	**Sink 4**	**Sink 5**
Communication Checker - Single Frame		***CYComChkSgl***					
	Message structure alarm	*CYComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Protocol value alarm	*CYComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Protocol co-values alarm	*CYComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Address alarm	*CYComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	ID alarm	*CYComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Configuration value alarm	*CYComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Analysis result	*CYComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI

Conclusion

As written in section 1.2.2, the IEC 62351 introductory part identifies intrusion detection as key to a full end-to-end security framework (but out of the scope of the standard). Assessing cyber security risk of a generic example substation, we have confirmed that intrusion detection is a passive security measure that would be pertinent to implement as part of an IEC 61850 cyber security policy. To answer the IEC 61850 standard lack of cyber security material, we have specified an intrusion detection function compliant with the IEC 61850 data model. We did so thoroughly following the extension rules given by the standard to ensure that interoperability would not be compromised by extending the model. In accordance with the IEC 61850 standard approach, the data objects defined in this work give abstract specifications of the intrusion detection function. Concrete implementation of it is out of the scope of the standard. We propose such a concrete implementation in the following chapter.

Chapter 4

Anomaly detection in GOOSE communication

Contents

Introduction

Compared to the previous chapter where we proposed an extension to IEC 61850 specifications for dealing with intrusion detection on GOOSE protocol, this chapter presents more concrete and experimental contributions. As part of a substation risk assessment (see section 3.2), it was necessary to test the available IEDs against malformed messages (not compliant with the IEC standard) and to demonstrate feasibility of intrusions [90].

This is the topic of section 1. Section 2 explains what information can be extracted from configuration files to help tune detection rules. Section 3 details the proposed detection algorithm [88] while section 4 presents the integration of a GOOSE parser into Bro. A proposition of an IEC 61850 ICS architecture resilient to GOOSE attacks concludes the chapter [87].

4.1 Cyber vulnerabilities and exploits in GOOSE networks

4.1.1 IEC 61850 attacks in the literature

Literature about vulnerabilities assessment and attacks testing in IEC 61850 environments are rather scarce. When existing, papers on this topic are usually not very detailed. The main reason is that authors do not want to disclose information that may be used for ill-intentioned purpose. They rather communicate their discoveries to the vendors in order for them to patch these vulnerabilities

Hong et al. [51] generate several types of attacks on both IEC 61850 data-link layer protocols, that is GOOSE and SV as well. They include replay of intercepted frames, message modification, injection, generation and DoS.

Attacks may also exploit vulnerabilities of MMS, the client-server protocol of the IEC 61850 stack: Kang et al. [91] investigate a Man-In-The-Middle (MITM) attack on MMS. The attacker hijacks the connection between the two victims, who keeps "believing" they communicate with each other while the attacker actually intercepts all messages of the link, modifies them and relays them. The authors explain how the attacker can launch several types of attack based on the MITM mechanism, including eavesdropping, data modification or injection, and DoS attacks.

Other protocols than the strict IEC 61850 stack (GOOSE, SV, MMS) are also used in IEC 61850 automation systems. Premaratne et al. [119] investigate packet sniffing, password cracking and DoS attacks in IEC 61850 substations. Protocols under consideration are ARP, FTP, HTTP, ICMP and Telnet.

Yang et al. [145] presents a study of IEC 61850 substations vulnerabilities based on fuzz testing approach. The test bed presented in [145] is also used to explore some of the attacks mentioned above (malformed packet attack, MMS DoS attack, ARP spoofing, MITM attack) and it additionally considers attacks such as reconnaissance attack, configuration files tampering or exploit of known OS vulnerabilities. *PowerCyber* test bed presented by Hahn et al. [47] was also the opportunity to assess vulnerabilities of software and communication protocols involved in smart grid technologies. The attacks they investigate include command injection and DoS. They also propose a scenario exploiting a combination of attacks targeting several mechanisms of the substation to study their

physical impacts. Both test beds were briefly introduced in section 2.6.

One of the most critical vulnerability of the IEC 61850 protocol stack concerns the GOOSE transfer mechanism. Exploiting this vulnerability allows an intruder to inject false GOOSE frames in the communication network. This false GOOSE frame injection attack is mentioned in several papers under different designations: poisoned GOOSE [96], GOOSE message spoof attack [55], malformed packet attack [145]. Description and demonstration of feasibility follow.

All these vulnerabilities and attacks may be used by an intruder to generate inappropriate or damageable behaviors of the system, thus compromising the optimal fulfillment of operation or causing damages. Possible inappropriate and dire consequences of such intrusions are studied in the risk assessment of a typical substation presented in section 3.2.

4.1.2 False GOOSE frame injection attack

GOOSE protocol presents vulnerabilities that can be exploited to generate a false frame injection attack. As shown by Hoyos et al. [55], an intruder only needs an access point to the LAN of the target substation to publish OSI-layer 2 frames, that is GOOSE frames in an IEC 61850 environment. These GOOSE frames will be interpreted as valid by the subscriber as long as the intruder follows GOOSE protocol specifications. Such an attack is possible because GOOSE messages are unencrypted and unauthenticated, due to latency constraints of real-time IEC 61850 communication (as discussed in sections 1.2.2 and 1.3.3).

As an answer to GOOSE replay, IEC 62351-6 [67] recommends the use of a security extension with HMAC signature (see section 1.2.2) and it additionally gives other recommendations about how a subscriber shall process received GOOSE frames to accept or discard them:

- The subscriber shall discard GOOSE frames whose timestamp exceeds a two-minute interval with its own clock.
- The subscriber shall discard GOOSE messages whose current status number StNum is smaller than StNum of the previous message by the same sending application. This holds if there has been no roll-over or TimeAllowedToLive timeout.
- In case of a StNum roll-over or a TimeAllowedToLive timeout, StNum shall be reset to its initial value 1.
- Upon initialization, the starting StNum shall be 0.

Figure 4.1 reproduces the timeline of GOOSE transfer mechanism explained in section 3.1.2.2 adding injection opportunities for the intruder.

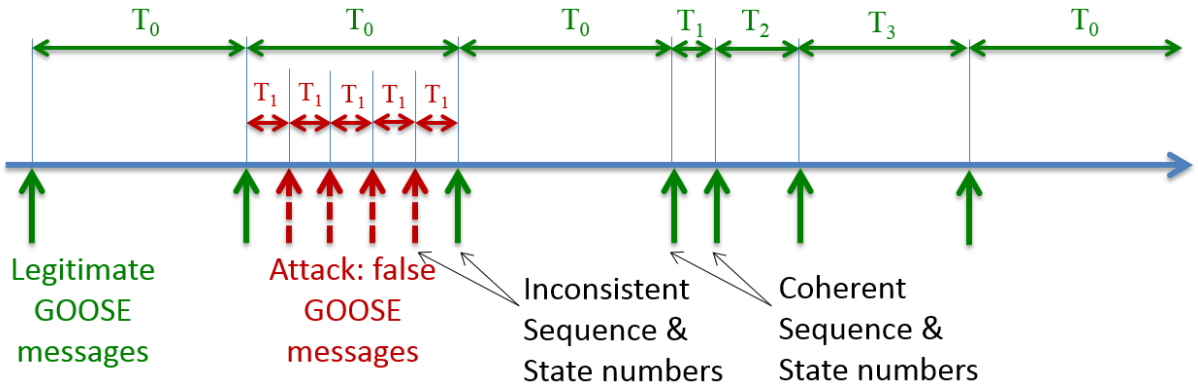

Figure 4.1: False GOOSE injection mechanism

Exploiting specificities of the subscriber processing algorithm of received GOOSE frames described above, an intruder can launch attacks broadcasting falsified GOOSE frames. Such attacks include:

- Denial of Service (DoS) attack: The attacker broadcasts one GOOSE message with a very high StNum. Once it has been processed by the subscriber, it is expected that further legitimate messages with a StNum less or equal to it will be discarded. In a more sophisticated version of this attack, the intruder can sniff GOOSE messages flowing over the network and extract current StNum for the target GOOSE connection. Then he increments this value by just one to forge the counterfeited GOOSE frame.
- Flooding attack: The attacker sends a series of GOOSE frames with increasing StNum values. Sending rate shall be high, that means delay between two messages shall be less than T1, the shortest transfer time after an event has occured. Thus, the attack is expected to be successful even in the worst case (events occur continuously and legitimate publisher broadcast messages with a time period of T1). It is expected that at some point fraudulent StNum will be greater than the legitimate one and flood of fraudulent messages will take over the sequence of the legitimate ones. In a basic version of this attack, the intruder can send messages the quickest as possible. In a sophisticated version he can sniff the GOOSE traffic and infer the temporal scheme of the target GOOSE application and tune the sending rate accordingly. As the previous attack, it results into a DoS.
- False data injection attack: This attack requires to sniff and parse GOOSE traffic and change values of the variables of interest in the data set, while maintaining the sequences of counters and timers, to forge and broadcast the fraudulent GOOSE frame. Processing the false GOOSE frames is expected to launch unrequired control functions at the subscriber side.

Lack of confidentiality (no encryption) of the GOOSE protocol allows to read genuine

GOOSE frames flowing over the network and the lack of integrity check (no authentication) enables the spoofed GOOSE frames to be processed by the subscriber. In the three variants of the mechanism, the subscriber processes a StNum greater than the current genuine one, it thus forces it to discard following legitimate GOOSE frames. As a consequence, the attacker causes at least a DoS and even takes control of the subscriber IED if he has enough knowledge about the system configuration (note that GOOSE messages carry the data set variables but not their meaning, the subscriber is supposed to know GOOSE messages content). Demonstration of these false GOOSE injection-based attacks can be found in the literature: DoS variants are presented by Kush et al. [96], while Hoyos et al. [55] demonstrated the feasibility of a semantic GOOSE attack to take control of an IED.

In this dissertation we focus on the false data injection. In practice, the attack script follows eight steps: (i) sniff packets, (ii) identify GOOSE frames using Ethertype 0x88b8, (iii) parse them using ASN.1 and BER specifications, (iv) identify the target GOOSE application, (v) modify the value(s) of interest in the data set, (vi) adapt StNum, SqNum and timestamp T to keep consistency with on-going legitimate sequence of messages, (vii) encode the frame using ASN.1 and BER, (viii) and send it through the physical port. Details of a GOOSE frame structure and encoding are given in section 3.1.2.1.

Hoyos et al. [55] implemented such a script using *Scapy* and made it available online[3]. *Scapy* is a packet manipulation program written in Python and using the C/C++ library for network traffic capture, libpcap. It is able to sniff, decode, forge and send packets of a wide number of protocols. GOOSE is not one of them. This is why we had to implement the ASN.1/BER specifications of GOOSE protocol to perform steps (iii) and (vii) of the attack script, respectively decoding the captured frame and encoding the fraudulent one.

4.1.3 Feasibility demonstration of the false GOOSE injection attack

4.1.3.1 Electrical protection and selectivity/discrimination

The role of electrical protection is to stem breakdown, to contain it and prevent it from spreading and causing a cascading failure. Protection is realized by SAS whose protection relays continuously monitor the state of the supervised electrical components and isolate them when they are subjected to serious disturbances such as short circuits. Protection mechanisms cannot prevent disturbances from occurring, they aim at limiting their impact instead. Their main purpose is to protect people from electrical accidents and power assets from damages (a three-phase short-circuit on medium-voltage bus bars can melt up to 50 kg of copper in one second), and to provide service continuity [123].

[3] https://github.com/mdehus/goose-IEC61850-scapy

Selectivity is key to electrical protection, it is essential for maintaining service continuity. It consists in localizing and disconnecting the fault part of the power grid, and no more, while maintaining under power the greatest part of the architecture [4]. This is done by opening the circuit breaker (CB) immediately upstream to the fault and that CB alone. There are many selectivity methods, among which the two main are time-based and logical or communication-based selectivity. In Figure 4.2, the fault on transmission line A is observed by both protecting relays A and B. If the protection mechanism follows time-based selectivity, the relay the closest to the fault, A, is supposed to open its associated CB A if the fault is persistent. If relay B still observes the fault after a configured time-delay, it means that CB A has failed to trip and relay B opens CB B. In a wider application there could be C and D relays/CBs with longer time delays. Logical selectivity ensures a quicker isolation of the affected power assets because it does not rely on programmed time-delays: relay A sends a command to its upstream relay B to prevent it from tripping CB B. If the CB A fails to open and the fault still exists, relay A stops sending its blocking command to relay B, which opens CB B.

In conventional protection systems, such logical selectivity blocking commands are transmitted from relay to relay through copper wires. In IEC 61850 design, these hard wired command signal exchange is replaced by a high speed inter-relay communication GOOSE.

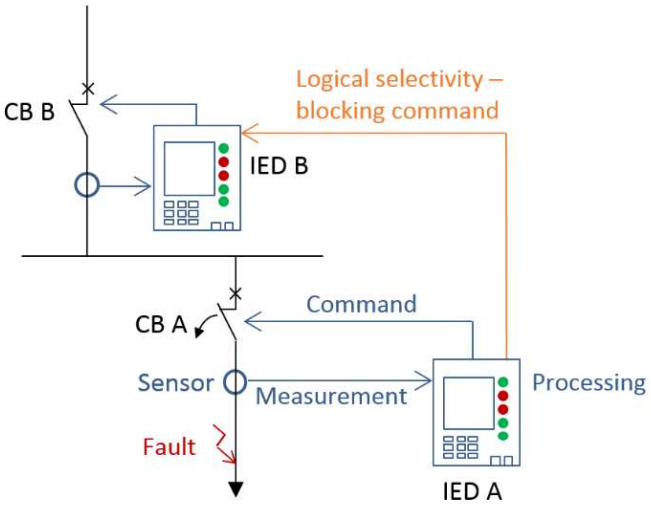

Figure 4.2: Logical selectivity principle

4.1.3.2 Test bed description

Ense3 Grenoble Institute of Technology, together with GIPSA-lab (Grenoble Images Speech Signal and Control laboratory), has developed an experimental platform dedicated

to ICS interoperability and cyber security, G-ICS Sandbox (GreEn-ER[1] Industrial Control Systems Sandbox)[2] . The presented test bed comes as a part of G-ICS and is shown in Figure 4.3. Its objective is the study of cyber security in IEC 61850 communication networks and systems for power utility automation. It includes all typical components of a SAS:

- Ethernet network for supervision-to-IEDs and inter-IEDs communications.
- Off-the-shelf IEDs from diverse vendors: Current experiments use a bay controller, an overcurrent protection relay, a transformer protection relay and a feeder protection relay from two vendors. Other IEDs are available but have not been operated yet.
- Engineering workstations with configuration tools.
- Supervision applications and Human-Machine Interfaces (HMI).

Regarding the process, that is the power grid, we do not have access to any real infrastructure (neither real-world one nor small-size laboratory one). But the process must be part of such a test bench dedicated to cyber security of the automation systems controlling it to comprehend possible impacts of cyber risks on the physical infrastructure. We thus made the choice of a feasible and affordable solution: hardware-in-the-loop simulation where electrical architectures are simulated by a software platform but still controlled and monitored by real off-the-shelf IEDs. Of course, such a solution helps understanding the system behavior but cannot give a complete representation of components interactions. A STM32-based I/O (Input/Output) card was developed by GIPSA-lab to ensure signal conversion between simulation and IEDs. Simulation software communicates with the I/O card over UDP for sending and receiving both analog and binary values to and from the IED physical connections. The card is wired to the IED binary I/O and to its analog measurement modules.

Regarding cyber security tools, an attacking computer is connected to the network to sniff high-speed real-time communication and launch false data injection and spoofing GOOSE attacks, and another computer runs our anomaly detection module.

4.1.3.3 Protection scenario

Let us consider a simple distribution substation from the typical substation topologies used as reference in the IEC 61850 standard [73]. These are classified by types (transformation or distribution) and size (small, medium, large) to be representative of worldwide substations. Figure 4.4 shows the considered distribution single-line diagram with an

[1] *Grenoble Energie - Enseignement et Recherche* – Grenoble Energy - Teaching and Research
[2] https://persyval-lab.org/en/platform/g-ics-sandbox-green-er-industrial-control-systems-sandbox

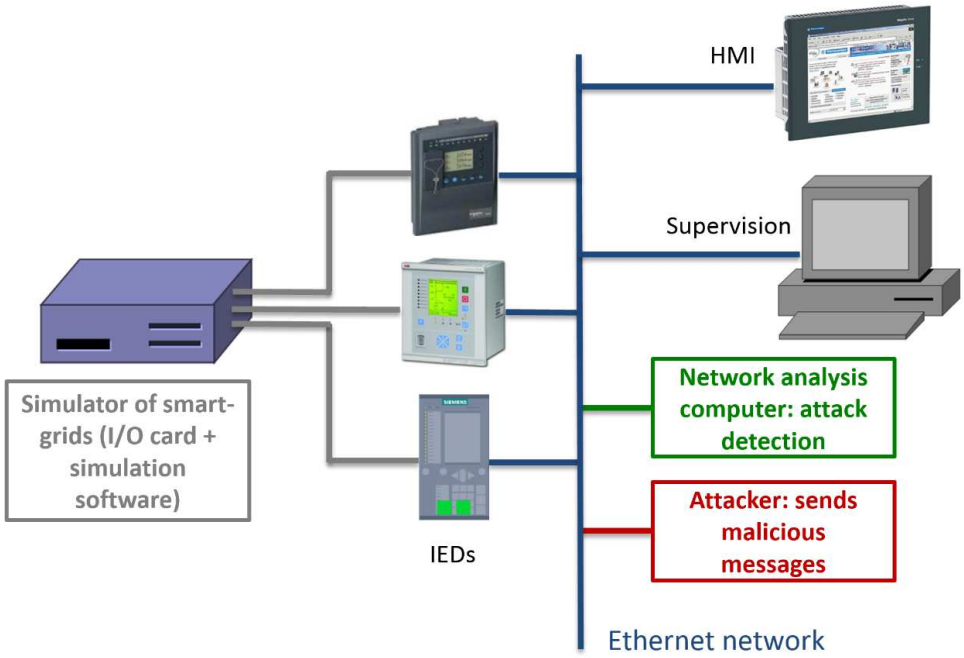

Figure 4.3: IEC 61850 cyber security test bed

overcurrent protection and a backup protection (i.e. breaker-failure protection). Logical selectivity as explained in section 4.1.3.1 is implemented here.

When an overload or a phase-to-phase short-circuit occurs downstream line 1, the associated protection relay IED 1 measures an overcurrent. It simultaneously sends a trip signal to CB 1, the CB directly upstream to the fault, and publishes a GOOSE message with the faulty current value and a Boolean variable to prevent CB 2 from opening. When CB 1 operating time has elapsed and fault is still present, meaning CB 1 has failed to open, or if CB 1 has an internal failure, IED 1 changes blocking Boolean variable to false. When IED 2 receives the corresponding GOOSE message, it sends a trip signal to CB 2.

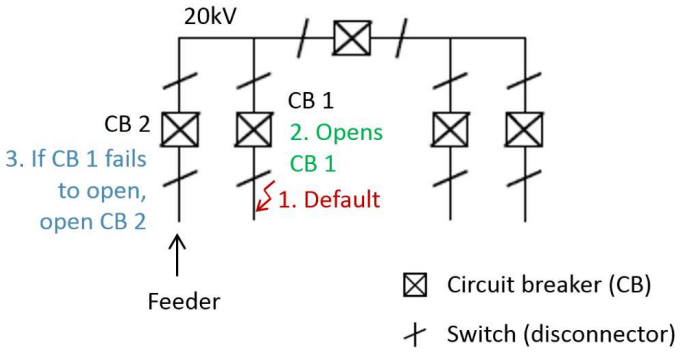

Figure 4.4: Example of a distribution substation with overcurrent protection

4.1.3.4 Risk analysis

Considering the simple distribution substation described above and shown in Figure 4.4, we focus our risk analysis onto false data injection in the high-speed Ethernet network. Such an attack can cause CB 2 to trip inappropriately or conversely to not trip when it should. In case of an inappropriate trip, the whole substation is de-energized since CB 2 protects the incoming feeder line. If there is an electrical fault on line 1 and neither CB 1 nor CB 2 trips, fault is not isolated and possible consequences are physical damages to substation components (worst possible case being destruction) and/or substation breakdown.

4.1.3.5 Attack scenario

A fictive attacker wants to disturb the production of a factory. His/her target is then the substation responsible for powering the factory facilities, which topology is shown in Figure 4.4. We assume that the attacker can connect to the substation Ethernet network and sniff or send packets. We also assume he/she knows the substation GOOSE messages configuration. The attacker is thus able to read GOOSE messages, forge new ones and inject them on the network for targeted IEDs to read them and use their malicious content. Attacker's objective is to de-energize the facility. He/she sniffs GOOSE packets of the substation until the situation of an overcurrent on line 1. He/she then injects GOOSE messages with the genuine current value (greater than configured overcurrent threshold) and Boolean variable "CB 1 Failure" as TRUE while its genuine value is FALSE. Attack timeline is depicted in Figure 4.5. Once IED 2 has read an attack GOOSE message it denies following genuine GOOSE flow because of mismatching message counters (see section 4.1.2).

Our protection scenario is simulated on Matlab, current values and CBs states are sent to IEDs over UDP initially and then when changing. IED commands to CBs are also transferred to the simulation in UDP packets. The described protection and attack scenarios are run. Figure 5 shows the resulting GOOSE communication captured with Wireshark protocol analyzer. First column is capture time. Source column gives the MAC address of publisher IED: the "1c" address is IED 1's and "1a" is IED 2's. Green messages from IED 1 corresponds to stable conditions with no fault and pale blue messages from IED 2 asserts CB 2 state is closed. Orange messages from t=7.938807s to t=8.589623s evidences an overcurrent with operating delay for CB 1 still going on. Attack message is highlighted by the red rectangle. It consequences is that IED 2 opens CB 2 and sends this new CB 2 state in the blue GOOSE messages. The orange message with capture time t=9.231743s is the next genuine message from IED 1. But it is too late, CB 2 has already been open and the substation is not powered any longer.

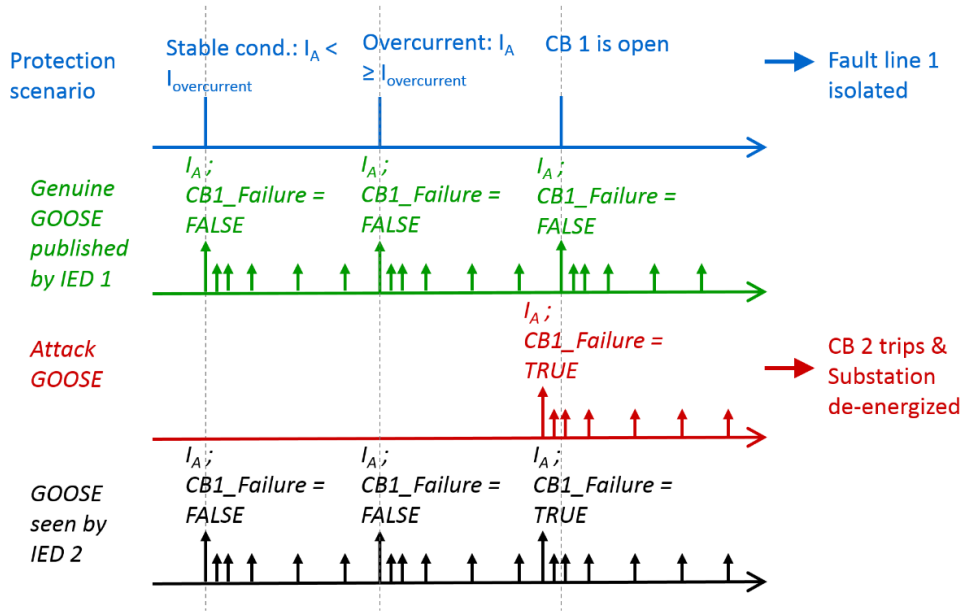

Figure 4.5: Figurative timeline of protection and attack scenarios

Figure 4.6: Wireshark capture of the GOOSE traffic during attack scenario

4.1.4 How off-the-shelf IEDs do react when receiving malformed GOOSE frames?

The false GOOSE data injection that we described above exploits a vulnerability of the GOOSE transfer mechanism. The implementation we made of it assumes that the attacker has good knowledge of the targeted system and crafted the frame to inject in order to cause a specific consequence.

An attacker with no or limited knowledge of the system at all and who wants to disrupt or damage the power utility may try the following strategy: send GOOSE frames with random values. This section thus explores how off-the-shelf IEDs, available in our test bed, deal with malformed GOOSE frames.

4.1.4.1 Security features of the available IEDs

We have checked the security features offered by three SIPROTEC 5 devices by Siemens, and two REC 650 and a REF 615 by ABB, as presented in technical documentation. Both vendors support authenticated and TLS encrypted communication between IED and engineering software tool as recommended by IEC 62351 [126], [125], [1]. User authentication with password can be configured for full access to the relay from engineering tool for both and interfaces for ABB device (local and web HMI, client-server MMS communication). As stated in Siemens documentation, files to be transferred from and to a SIPROTEC 5 relay are digitally signed, crypto chips are used for securely storing public keys for signature verification.

Both vendors follow IEEE 1686-2013 requirements regarding security logins and logouts: successful and unsuccessful access attempts are recorded into a separate nonvolatile audit trail for the administrator to read from configuration tool and on-site operation panel. Both vendors give generic cyber hygiene recommendations about global communication architecture with bounded security zones separated by firewalls and about antivirus and updates of the engineering computers. GOOSE communication being critical to the safety of the system, alternative modes with pre-defined values can be configured in case of communication losses or syntax inconsistency of messages. This is true for both vendors [2], [127]. Moreover both vendors give the opportunity to perform diagnosis on GOOSE traffic. Subscriber IED checks many parameters of GOOSE message header and in case of mismatch with what is expected, or also in case of communication losses or timeouts, messages are discarded and default values are used instead [2], [127].

4.1.4.2 Testing

The GOOSE protocol as defined in the IEC 61850 standard is vulnerable as we demonstrated. The objective of this study is to feed a real off-the-shelf IED with GOOSE frames, which do not fit the framework defined by protocol specification, and see how the device

reacts. The idea is to check whether the invalid (data) and malformed (structure) GOOSE frames are discarded or processed by the IED because acceptance of an invalid GOOSE frame may potentially be a vulnerability for an intruder to exploit and launch an attack. IED under study implements GOOSE protocol as specified in IEC 61850 Ed.1, which is as described in section 3.1.2.

Considering an application, typical and expected values of all fields are collected and defined using the standard specification and the project configuration files. For each of the GOOSE frame fields (see section 3.1.2), invalid values or length regarding the protocol definition are tested. Test cases are described in Table E.1.

For implementing the test cases defined in Appendix E, the substation automation project requires: at least two GOOSE applications published by one IED, at least one GOOSE application published by another IED, a dataset with at least two objects. The electrical protection project configured for the attack scenario implemented in section 4.1.3 fulfil these conditions as it counts two IEDs (7SJ82 and 7UT82), which publish respectively one and two GOOSE applications.

Using Wireshark, we captured a sample of the genuine traffic. Using *GHex*, a simple binary editor that lets users edit a binary file in both hex and ascii formats, we created *pcap* traces for each test case. A trace is generally composed of five genuine frames and the sixth is the one to test. A *Scapy* script helped send the traces on the network.

The IED under study maintains internal statistics about GOOSE communication. For each GOOSE application that it has subscribed to, two counters are interesting for our purpose: "RxCounter" is incremented each time the IED processes a GOOSE frame for the corresponding application, and "RxMismatch" is incremented when the received GOOSE frame has an invalid parameterization. So after a test case has been run, we connect to the IED through the configuration software and check these two counters to conclude whether the malformed frame was discarded or accepted.

The greatest majority of the tested frames were discarded by the subscriber IED. But this study has some unexpected outcomes that are worth to be mentioned.

If the *Source address* field does not have the expected value for the corresponding GOOSE application but matches another IED MAC address configured in the project (including the subscriber address), then the frame is processed and the data it carries are used. If the fake MAC address value is not part of the project, the frame is discarded.

A frame with a *TimeAllowedToLive*, *DatSet*, *GoID*, *Test*, *ConfRev* or *NdsCom* different than the configured values is always processed. It means that the IED only uses the fields *Destination address* and *APPID* to identify the GOOSE application and use its content. Regarding the *GoID*, it is acceptable as this field is said to be optional in the protocol specification and thus a frame with no *GoID* at all is actually accepted by the subscriber IED.

Whatever the value of the fields *Reserved 1 and 2*, the GOOSE frame is accepted, even

when they are missing in the frame structure. They are not optional, though.

An inacurrate time stamp T compared to the subscriber clock does not prevent the frame to be processed and used, its value being either well in the past or in the future. This is in contradiction with IEC 62351 recommendations about GOOSE processing at the subscriber side as detailed in section 4.1.2.

These inconsistencies in this implementation of the GOOSE protocol are regrettable to our opinion as they may ease the task of a potential intruder.

Conclusion

GOOSE communications are vulnerable. Monitoring them and check their behavior is as expected may help securing them. As discussed in section 1.3, IACS show characteristics that can be leveraged for intrusion detection. One of them is that communication system configuration is well defined and often readily available through configuration files. IEC 61850 control and automation systems have several configuration files written in a dedicated XML-based language, Substation Configuration Language – SCL, and that provide such interesting information.

4.2 Communication information from SCL configuration files for automatic rule generation

4.2.1 SCL configuration files

The IEC 61850 standard specify a configuration file format for a formal description of the switchyard topology, the substation automation and the communication system, and the relations between them. This format is key to the interoperability that the standard aims at, as it enables a compatible way to exchange descriptions of IEDs capabilities and SAS between engineering tools of different vendors.

The language defined by the standard is called **Substation Configuration description Language - SCL** and is based on XML v1.0. The scope of SCL covers models of the primary system structure, the communication system, the application communication (ACSI), each IED (data object model), instantiable LN type definitions (SCL provides templates of really implemented LN and data object types), and relations between instantiated LNs and their hosting IEDs on one hand, and with the primary system functions on the other hand.

SCL allows to specify IED and communication system configurations in a serialized form and a standardized syntax, and with a semantic following the IEC 61850 data ob-

ject model defined in parts 5 and 7-x of the standard (see section 3.3). SCL files are organized into three submodels following a hierarchical structure as shown in Figure 4.7: (i) *Substation*: provides a functional view of the primary system and its topology, (ii) *IED*: describes automation devices and their data object model, and (iii) *Communication system*: contains communication-related objects and describes connections between IEDs.

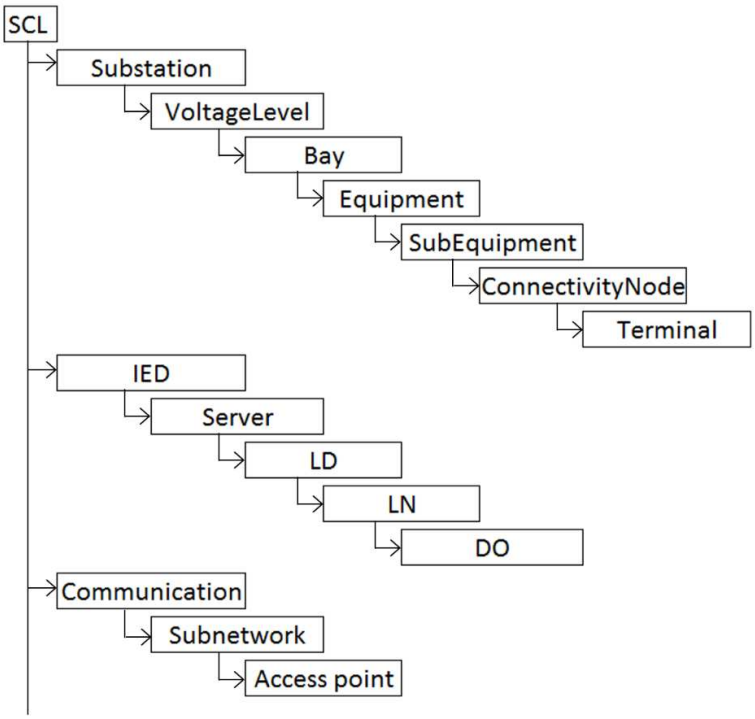

Figure 4.7: SCL file structure

Each SCL file starts with a *Header* section identifying an SCL file and its version, and may end with an additional section defining templates of really implemented LNs and data objects, *DataTypeTemplate*.

4.2.2 Types of SCL files

Different file extensions enable to cover all the engineering purposes of SCL data exchange:

- **IED Capability Description - ICD**: An ICD file describes the capability of a device and must be provided by the manufacturer. It shall contain an *IED* section and may contain optional *Communication* and *Substation* parts. If defined, the *Substation* section provides the binding of LN instances to physical entities, thus describing a predefined functionality.
- **System Specification Description - SSD**: An SSD file describes the single line diagram of the substation along with the required LNs. It shall thus have a *Substation* section and templates and definitions of the needed data types and LNs.

- **Substation Configuration Description - SCD**: This file describes the whole substation configurations, thus containing sections for all IEDs, a substation section, a communication section and data types definitions.
- **Configured IED Description - CID**: A CID file is the configuration file uploaded into an instantiated IED of a real substation, with current address and communication parameters and variable names. This file format enables information exchange between the IED configuration tool and an IED.
- **Instantiated IED Description - IID**: As a CID file, an IID file contains only the configuration of a single IED of a project with an IED section, a communication section that gives only the IED's communication parameters, the IED's data type templates and an optional substation section with the binding of the IED's LNs to the switchyard. This type of file is transferred from the IED configurator to the system configurator.
- **System Exchange Description - SED**: This file format describes the possible and actual connections between several projects and is exchanged between the system configurators.

The last two file formats were introduced by the second edition of the IEC 61850 standard.

4.2.3 Extracting GOOSE communication-relevant information from SCL files

The idea of automatically producing intrusion detection criteria from system description files for IACS environments is rather straightforward as their format is well defined and they contain a lot of relevant information regarding security. Hence, it is not surprising that we found in literature an example of such an approach. Article by Hadeli et al. [46] presents a prototype tool to generate configuration for missing/delayed traffic detector and firewall using the system description files as input.

According to the description of existing types of SCL files given in previous section (4.2.2), Substation Configuration Description file appears to be the most appropriate. It specifies the whole system, giving the global communication system's parameters and also the configuration of all communication applications implemented by each IED of the project. The Substation Configuration description Language is defined in IEC 61850-6. In this section we compile definitions of sections and elements relevant to GOOSE communication.

Regarding GOOSE communication, relevant information can be found in two places of the SCL file. The *Communication* section defines the topology of the global system network and the characteristics of protocols used. If a logical device implements a GOOSE application, the content of the data set is defined in corresponding *LDevice* sections.

Communication section

Communication section defines all logical connections (*SubNetwork*) and IEDs' access points connected to them (*ConnectedAP*) and that may communicate using the *SubNetwork*'s protocol without a router. A *ConnectedAP* is identified by the two following attributes: its IED name (*iedName*) and the name identifying the access point within the IED (*apName*). For each *ConnectedAP*, addresses and configuration parameters of communication applications (e.g. GSE or SV applications) are defined. Parameters of a GOOSE application such as addressing and connection timing are defined in a *GSE* control section. Attributes identifying a GSE application are (i) *ldInst*, the instance identification of the LD within the IED, on which the control block is located, and (ii) *cbName*, the name of the control block within the LLN0 of the LD ldInst.

Thus elements found in this section are: *VLAN-ID*, *VLAN-PRIORITY*, *MAC-Address* and *APPID* as defined in section 3.1.2.1, and *MinTime* and *MaxTime*. VLAN-ID and VLAN-PRIORITY combined form the TCI field carried by a GOOSE frame as defined in section 3.1.2.1. *MAC-Address* corresponds to the Source Address from GOOSE frame header. It is the broadcasting address of the considered GOOSE application. *MinTime* is the maximal allowed sending delay on a data change in ms. *MaxTime* is the source supervision heartbeat cycle time in ms. Within this time, a failure of the source shall be detected by the client. These parameters correspond to transmission times T1 (event) and T0 (stable conditions) respectively as defined in the GOOSE transfer mechanism explained in section 3.1.2.2. The value of the GOOSE PDU field *TimeAllowedToLive* thus can be calculated as 1.5 times *MaxTime* (see section 3.1.2.2).

Figure 4.8 shows the *Communication* section of an example SCL file. The corresponding system counts only one subnetwork named "PN/IE_1". Two access points are connected to this subnetwork: access point "E" of IED named "SIP1" and access point "E" of IED named "SIP". They both are involved in a GOOSE application and in each of the two access point sections is defined a *GSE* section. Logical device *CB1* of IED SIP1 hosts a GOOSE control block named *Control_ Dataset*. The corresponding GOOSE application identifier is *0002*, messages are broadcasted using the MAC address *01-0C-CD-01-00-01* with a cycle time of 1s (*1000*ms) in stable conditions or *5*ms in case of event. The second GOOSE application of the system under consideration is hosted by logical device *CB1* of IED SIP and is named *Control_ Dataset*. The corresponding GOOSE application identifier is *0001*, messages are broadcasted using the MAC address *01-0C-CD-01-00-00* with a cycle time of 2s (*2000*ms) in stable conditions or *10*ms in case of event. VLAN is not used and GOOSE messages priority is set to the default value *4* for both applications.

LDevice section

The SCD file counts as many *IED* sections as there are IEDs in the system. The *IED* section describes the IED configuration: access points, instantiated logical devices, logical

```
<Communication>
  <SubNetwork name="PN/IE_1">
    <Private type="Siemens-MasterId">137988709285889</Private>
    <Private type="Siemens-Application">GOOSE 7SJ82 to 7UT82||0001||PriorityLow||10|2000|000|4</Private>
    <Private type="Siemens-Application">GOOSE 7UT82 to 7SJ82||0002||PriorityMedium||5|1000|000|4</Private>
    <ConnectedAP iedName="SIP1" apName="E">
      <Private type="Siemens-Application-GSEControl">GOOSE 7UT82 to 7SJ82|CB1|Control_Dataset</Private>
      <Address>
        <P type="IP" xsi:type="tP_IP">10.10.20.5</P>
        <P type="IP-SUBNET" xsi:type="tP_IP-SUBNET">255.255.0.0</P>
        <P type="IP-GATEWAY" xsi:type="tP_IP-GATEWAY">10.10.255.254</P>
        <P type="OSI-AP-Title">1,3,9999,23</P>
        <P type="OSI-AE-Qualifier">23</P>
        <P type="OSI-PSEL">00000001</P>
        <P type="OSI-SSEL">0001</P>
        <P type="OSI-TSEL">0001</P>
      </Address>
      <GSE ldInst="CB1" cbName="Control_Dataset">
        <Address>
          <P type="VLAN-ID" xsi:type="tP_VLAN-ID">000</P>
          <P type="VLAN-PRIORITY" xsi:type="tP_VLAN-PRIORITY">4</P>
          <P type="MAC-Address" xsi:type="tP_MAC-Address">01-0C-CD-01-00-01</P>
          <P type="APPID" xsi:type="tP_APPID">0002</P>
        </Address>
        <MinTime unit="s" multiplier="m">5</MinTime>
        <MaxTime unit="s" multiplier="m">1000</MaxTime>
      </GSE>
    </ConnectedAP>
    <ConnectedAP iedName="SIP" apName="E">
      <Private type="Siemens-Application-GSEControl">GOOSE 7SJ82 to 7UT82|CB1|Control_Dataset</Private>
      <Address>
        <P type="IP" xsi:type="tP_IP">10.10.20.6</P>
        <P type="IP-SUBNET" xsi:type="tP_IP-SUBNET">255.255.0.0</P>
        <P type="IP-GATEWAY" xsi:type="tP_IP-GATEWAY">10.10.255.254</P>
        <P type="OSI-AP-Title">1,3,9999,23</P>
        <P type="OSI-AE-Qualifier">23</P>
        <P type="OSI-PSEL">00000001</P>
        <P type="OSI-SSEL">0001</P>
        <P type="OSI-TSEL">0001</P>
      </Address>
      <GSE ldInst="CB1" cbName="Control_Dataset">
        <Address>
          <P type="VLAN-ID" xsi:type="tP_VLAN-ID">000</P>
          <P type="VLAN-PRIORITY" xsi:type="tP_VLAN-PRIORITY">4</P>
          <P type="MAC-Address" xsi:type="tP_MAC-Address">01-0C-CD-01-00-00</P>
          <P type="APPID" xsi:type="tP_APPID">0001</P>
        </Address>
        <MinTime unit="s" multiplier="m">10</MinTime>
        <MaxTime unit="s" multiplier="m">2000</MaxTime>
      </GSE>
    </ConnectedAP>
  </SubNetwork>
</Communication>
```

Figure 4.8: Communication section of example SCL file

nodes and data, communication services offered, and default and configuration values. An IED has at least one server that enables access to its LDs and LNs through access points. GOOSE settings of a publisher LD is found in *LN0* section for the LLN0 logical node dedicated to supervision and control of the LD. The element *DataSet* lists the data to be transmitted by mean of GOOSE and *GSEControl* element provides the GOOSE control block as defined in 3.1.2.

DataSet section is identified by a *name* attribute, unique in the LN where it is defined. It contains a sequence of *FCDA* elements, that is functionally constraint data or data attribute of this IED to be in the data set. A functional constraint is a property of a data attribute defining the services that may be applied to it; there are defined in IEC 61850-7.2 [69]. The order of the FCDA elements defines the order of the values in the

message. The elements has the following attributes:

- *ldInst*: The LD where the DO resides.
- *prefix*: Prefix identifying together with lnInst and lnClass the LN where the DO resides.
- *lnClass*: LN class of the LN where the DO resides.
- *lnInst*: Instance number of the LN where the DO resides; shall be specified except for LLN0.
- *doName*: A name identifying the DO (within the LN) as standardized in IEC 61850-7.4. If doName is empty, then fc can contain a value, selecting the attribute category of all DOs of the defined LN.
- *daName*: The attribute name. If empty, all attributes with functional constraint given by fc are selected.
- *fc*: All attributes of this functional constraint are selected. As defined in IEC 61850-7.2 "*the functional constraint (FC) shall be a property of the DataAttribute characterizing the specific use of the DataAttribute*". Possible constraint values are specified in IEC 61850-7.2 [69].

The field *AllData* carried by the GOOSE PDU is composed of the ordered sequence of the values only. All the attributes identifying data objects are not conveyed.

Each GOOSE application for publishing of the host LD data values shall have its dedicated *GSEControl* section. Its attributes are:

- *name*: The name identifying this GOOSE control block, unique within the LD.
- *desc*: A description text. Optional.
- *datSet*: The name of the data set to be sent.
- *confRev*: The configuration revision number of this control block.
- *type*: Default type is GOOSE.
- *appID*: A system wide unique identification of the application to which the GOOSE message belongs.

Concretely, these attributes are used to forge some of the fields of the GOOSE PDU as defined in IEC 61850-7.2 [69]. *Name* attribute is used to build the *GoCBRef* field: "LDName/LLN0.GoCBName" with GoCBName being *name* attribute. Attribute *datSet* is the DataSetName in "LDName/LNName.DataSetName" with LNName being LN0, thus

resulting into the *DatSet* field. If the data set the *datSet* refers to is empty, Boolean field *NdsCom* in GOOSE PDU will be set to TRUE, meaning the GOOSE control block requires further configuration. *ConfRev* of the GOOSE PDU is equal to the *confRev* attribute value defined in the SCL file. *GoID* is equal to the *appID* attribute value from SCL file. The number of *FCDA* elements gives the value of the GOOSE PDU field NumDatSetEntries.

```
        <LDevice desc="CB1" inst="CB1">
          <Private type="Siemens-MasterId">775fea0a-d58a-4a93-88e6-3877ba7438e0</Private>
          <LN0 desc="General" lnClass="LLN0" lnType="SIPROTEC5_LNType_LLN0_LDevice_Generic_14_1" inst="">
            <Private type="Siemens-MasterId">66fdc0e7-1b11-4b78-843e-472d54e9d431</Private>
            <DataSet name="Dataset">
              <FCDA ldInst="CB1" lnClass="PTRC" lnInst="1" fc="ST" daName="stVal" doName="MonSglF1" prefix=""/>
              <FCDA ldInst="CB1" lnClass="PTRC" lnInst="1" fc="ST" daName="q" doName="MonSglF1" prefix=""/>
            </DataSet>
...
            <GSEControl name="Control_Dataset" appID="SIP/CB1/LLN0/Control_Dataset" confRev="1" type="GOOSE" datSet="Dataset"/>
          </LN0>
```

Figure 4.9: Excerpt of a LDevice section of example SCL file

Figure 4.9 shows an excerpt of the *LDevice* section CB1 (Circuit Breaker, instance 1) of IED SIP from Figure 4.8. The data set transmitted by this GOOSE application has two items. They are the data attributes *stVal* (Status value of the data) and *q* (Quality of the data attribute value) of the data object *MonSglF1* belonging to LN "Protection trip conditioning" *PTRC*. Both data attributes have the functional constraint *ST*, involving that the data attribute "shall represent a status information whose value may be read, substituted, reported, and logged but shall not be written" (definition of the FC ST in IEC 61850-7.2 [69]).

Conclusion

Most of the fixed fields of GOOSE frames (as defined in section 3.1.2.1) can be extracted from the SCD configuration file, either directly (that is explicitly mentioned: Destination MAC address, APPID, GoID, ConfRev) or indirectly (that is computed from other fields values: TCI, GoCBRef, TimeAllowedToLive, DatSet, NdsCom, NumDatSetEntries).

Source MAC addresses do not appear in any of the configuration files. Only IP addresses of IEDs connected access points are given (see Figure 4.8). Thus, in order to automatically extract GOOSE communication specificities of the system and tune the IDS, it is necessary to have a list of corresponding IP and MAC addresses.

As mentioned in section 3.1.2.1, some fields are fixed by GOOSE protocol definition (TPID, Ethertype). During operation, Test field is set to FALSE. It may be turned to TRUE by engineers for testing periods but intrusion detection would not occur then. Remaining fields are dynamic and their values are set at publishing time (Length, T, StNum, SqNum) by the application.

4.3 Detection of corrupted GOOSE frames

The tamper detection or intrusion detection has been determined as a required security measure for all three IEC 61850 protocols by clause 6.10.2 of IEC 62351-1 [66].

The intrusion detection approach for GOOSE communication that we propose in this work covers multiple dimensions of the ICS-oriented IDS taxonomy introduced in section 1.4.2. The rules proposed in this work cope with communication aspects, including *protocol vocabulary*, *grammar* and *exchanges structure*. We do not make use of *telemetry* criteria (even though data objects of the complementary cyber security set specified in Chapter 3 can help handle such criteria). Process awareness is not addressed by the following rules.

In this section, we use the terminology defined in the section 3.4 about the proposed complementary IEC 61850 data object model dedicated to cyber security. The idea is to make the connections between the data object model concepts and the concrete detection approach as explicit as possible. Thus the names of alert types are the names of the Status Information data attributes of communication checkers LNs specified in section 3.4.3.

Communication checkers are based on the GOOSE protocol specification presented in section 3.1.2. Values specific to the project under study are extracted from SCL configuration files as explained in previous section 4.2.3. The *Communication checker - single frame* is basically concerned with intra-frame inspection regarding protocol definition, communication links, communication system configuration and communication application (GOOSE Control Block) parameters. The *Communication checker - multiple frames* supervises sequences of messages in terms of time intervals and order. Between the two of them, all fields of a GOOSE frame as described in section 3.1.2 are covered except *Ethertype* that is used by the parser to identify GOOSE frames. If it is not equal to GOOSE Ethertype 0x88b8, the message is discarded.

This section describes only the detection rules and no implementation, which is discussed in next section 4.4.

4.3.1 Filters

Each GOOSE application of the project under consideration has its own *filter* in the NIDS data base. It lists all the features peculiar to this GOOSE application, extracted from the system configuration files, and used as the trust basis to check compliance of the frame with the detection specifications. Filters also have further parameters, used for inter-packet inspection:

- Delay between publishing times of the current message (n) and the previous one (n-1): $\Delta T_n := T_n - T_{n-1}$.

- State number of the last message: $StNum_{n-1}$.
- Sequence number of the last message: $SqNum_{n-1}$.

For each GOOSE message parsed, the corresponding filter is identified using the value of GOOSE identifier field *GoID* from its PDU.

4.3.2 Communication checker - single frame

The following detection rules apply to fields of a single parsed GOOSE frame, covering four dimensions: protocol definition, communication links, communication system configuration and GOOSE application.

Protocol definition

Some fields of the GOOSE frame header have fixed values, given by the protocol specification.

- Is *TPID* equal to 0x8100, tag protocol identifier of 802.1Q Ethernet frames?
  if $TPID_n \neq 0x8100$ **then**
  Alert(ProtValAlm), Log
  end if
- Is the fourth most significant bit of *TCI* equal to 0?
- Are *Reserved 1* and *Reserved 2* 2-byte long and equal to 0x00? (Under the assumption that the IEDs used in the considered project do not implement authentication nor encryption mechanism.)
  if $Reserved1_n \neq 0x00 \vee Reserved2_n \neq 0x00$ **then**
  Alert(ProtValAlm), Log
  end if

The protocol defines boundaries for some fields:

- Is the *Destination address* in the range of multicast hexadecimal addresses allocated to GOOSE protocol "01-0C-CD-01-00-00" to "01-0C-CD-01-01-FF"?
- Is *Length* in the range defined by the standard (maximum 1500)?
  if $Length_n > 1500$ **then**
  Alert(ProtValAlm), Log
  end if

- Are StNum and SqNum values within the authorized boundaries: maximum value being $2^{32} - 1 = 4294967295$?

  if $StNum_n > 4294967295 \vee SqNum_n > 4294967295$ **then**
  Alert(ProtValAlm), Log
  end if

Some fields are related to the amount of data carried by other fields, in terms of byte length or number of items. Their value shall be in accordance with the actual content of the related field.

- Is *Length* equal to the actual octet-length of the Ethertype PDU (header starting at APPID, which represents 8 bytes, and APDU, which thus is 1492-byte long at the most)?

  if $Length_n \neq Length(APDU_n) + 8$ **then**
  Alert(ProtValAlm), Log
  end if

- Is *NumDatSetEntries* equal to the number of entries listed in the configuration file for the considered dataset?

  if $NumDatSetEntries_n \neq Number\_of\_item(DataSet_n)$ **then**
  Alert(ProtValAlm), Log
  end if

Communication links

- Does the *Destination address* (MAC address) comply with the communication application configuration?

  if $DestinationAddress_n \neq Filter(GoID).DestinationAddress$ **then**
  Alert(AddrAlm), Log
  end if

- Does the *Source address* (MAC address) comply with the communication application configuration?

  if $SourceAddress_n \neq Filter(GoID).SourceAddress$ **then**
  Alert(AddrAlm), Log
  end if

Communication system configuration

- Is the user priority coded by the three most significant bits of *TCI* equal to the priority value from system configuration?

  if $Priority_n \neq Filter(GoID).Priority$ **then**

Alert(ConfValAlm), Log
end if

- Is VID, coded by the twelve least significant bits of *TCI*, equal to the VLAN name from system configuration?
  if $VID_n \neq Filter(GoID).VLAN$ **then**
  Alert(ConfValAlm), Log
  end if

- The standard suggests that *TAL* be greater than (actually twice) the maximum retransmission time of GOOSE transfer mechanism (time T0 under stable conditions in Figure 3.5).
  if $TAL_n \neq 2 * Filter(GoID).MaxTime$ **then**
  Alert(ConfValAlm), Log
  end if

Control Block

- Are *APPID*, *GoCBRef*, *DatSet*, *ConfRev* equal to the values available in the configuration of the considered GOOSE application?
  if $APPID_n \neq Filter(GoID).APPID \vee GoCBRef_n \neq Filter(GoID).GoCBRef \vee DatSet_n \neq Filter(GoID).DatSet$ **then**
  Alert(IdAlm), Log
  end if
  if $ConfRev_n \neq Filter(GoID).ConfRev$ **then**
  Alert(ConfValAlm), Log
  end if

4.3.3 Communication checker - multiple frames

As highlighted by section 4.1.2 about the false GOOSE message injection attack, the GOOSE transfer mechanism as defined in the IEC 61850 is vulnerable. To detect intrusions exploiting this vulnerability, it is necessary to control the sequences of GOOSE messages in terms of time intervals as well as counters follow-up.

Time profile

The time profile of the sequence of frames for a given GoID shall comply with the minimum and maximum transfer time given in the system configuration file (as *MinTime* and *MaxTime* respectively), and with the definition of the GOOSE transfer mechanism recalled in section 3.1.2.2. When a new event occurs, a frame with the new values is generated "instantaneously", then twice after a time delay of *MinTime*, following messages

are published with doubled time delay compared to the previous one until *MaxTime* is reached. In stable conditions, GOOSE messages are emitted with a periodicity of *MaxTime*.

Order of the message sequence

Given the broadcasting nature of the GOOSE protocol, *StNum* and *SqNum* counters help to follow the connection state. Monitoring them enables detection of the GOOSE injection as demonstrated in section 4.1.2.

Algorithm 1 checks consistency of sequences of GOOSE messages regarding timing and counters.

Algorithm 1 Checking integrity of the GOOSE transfer mechanism

if $StNum_n == StNum_{n-1}$ **then**
  if $SqNum_n \neq SqNum_{n-1} + 1$ **then**
    Alert(CountAlm), Log
  else if $(SqNum_n == \{1, 2\} \wedge \Delta T_n \neq Filter(GoID).MinTime) \vee (SqNum_n \geq 3 \wedge [\Delta T_n \neq Filter(GoID).MaxTime \wedge \Delta T_n \neq 2 * \Delta T_{n-1}] \vee [\Delta T_n \neq Filter(GoID).MaxTime])$ **then**
    Alert(TxAlm), Log
  end if
else if $StNum_n == StNum_{n-1} + 1$ **then**
  if $SqNum_n \neq 0$ **then**
    Alert(CountAlm), Log
  else if $\Delta T_n \geq MaxTime$ **then**
    Alert(TxAlm), Log
  end if
end if

Conclusion

Following the taxonomy of IACS-oriented IDS introduced in section 1.4.2, the rules proposed above correspond to a Communication approach covering protocol vocabulary and grammar and structure of exchanges. They are derived from the IEC 61850 standard specifications and from SCL configuration files. Implementation a NIDS checking compliance of the monitored communications with these rules is the topic of the next section, which presents the integration of a GOOSE module into a network traffic analyzer, Bro.

4.4 Integrating the GOOSE protocol into an open-source NIDS, Bro

Integrating an intrusion detection module for GOOSE protocol into an open-source was proposed as a master thesis subject. The intern we recruited, Laurent Lê-Hébrard, had the required background in informatics security with good programming skills. The internship objectives were to choose an open-source NIDS, implement a parser for the GOOSE protocol and validate it. This section presents the outcomes of this work.

4.4.1 The choice of NIDS

The proprietary dimension of IEDs does not allow a third-party to install a HIDS of one's choice. Such a technology must be on the vendor's initiative. Other characteristics of both HIDS and NIDS are discussed in section 1.4.1 and tip the scale in favor of NIDS. For instance, IEDs are responsible for electrical protection of the system, a safety mission of primary importance and given limited resources of IACS (see section 1.3.3), implementing intrusion detection capabilities in the IEDs shall be considered with the greatest care.

A NIDS passively connects to the network and listens to the traffic, through a port mirorring connection to a switch for client-server communication or through direct connection to the LAN for multicast protocols, as GOOSE.

4.4.2 An existing open-source tool

An IDS must not introduce vulnerabilities and allow to be used as an attack vector, since its mission is to alert in case of security compromising. There are examples of vulnerabilities discovered in IDS parsers[4], though. An open-source IDS used and maintained by a broad community is expected to have been tested in a wide range of applications and environments. Even if it is not an absolute guarantee, we are more inclined to trust such an open-source tool than to implement our own one from scratch.

4.4.3 A layer 2 protocol sensor

IDS are mainly designed for typical companies networks. Their architecture and implementation are therefore generally concerned with protocols over TCP or UDP (OSI model layer 4) and mechanisms that would ease the creation and integration of sensors are often available for this kind of protocols. Some IDS even make possible the integration

[4] http://cve.circl.lu/cve/CVE-2015-0971

of protocols over IP (layer 3). However, GOOSE is a data link-layer protocol (layer 2) and fewer mechanisms are made available to develop the parser. This makes the integration of a GOOSE sensor challenging.

4.4.4 Open-source NIDS candidates

Bro, Snort and Suricata are three open-source NIDS with extensive user and developer communities. We found several examples of industrial IDS based on these tools in the literature including [33], [95], [102], [116] for Bro, [27], [46], [108], [115], [120], [121], [149] for Snort, and [28] and [32] for Suricata.

Suricata was preferred to Snort because both NIDS are comparable in their design and features but Suricata supports multi-thread, which makes it deliver higher performance [140].

We compared the two remaining candidates according to the following criteria:

- *Parser development process:* Bro has a well documented tool chain for creating a protocol parser: BinPAC[5]. Integrating a new parser into Bro is not documented, though. One has to refer to source code of available sensors. BinPAC is no longer maintained, since it has evolved into Spicy[6]. But the parser generator Spicy allows only to parse protocol over TCP or UDP, for the time being.

  Suricata is implemented in C and new parsers shall be written in this language as well. The C language is well known and enjoys a broad user community, which is a good point regarding maintainability of the project.

- *Detection rules:* Bro scripting language allows to specify complex detection rules. It is necessary to learn it to fully benefit from the features it offers, though. To this end, its detailed documentation[7] and a training web interface[8] may help.

  Suricata rules specification is not as flexible but much easier. Detection rules are written using a set of key words calling functions, written in C. Each protocol has its own set of key words. If one wants to write new rules that cannot be covered by the available key words, one needs to implement new key words/functions in C, recompile the whole Suricata project before writing the rules. This hinders a smooth integration of new rules.

- *Host security:* On one hand, the BinPAC parser development tool chain significantly reduces the opportunity of introducing vulnerabilities. On the other hand, the Bro scripting language has type inference and automatic memory allocation. So, the

[5] https://www.bro.org/sphinx/components/binpac/README.html
[6] http://www.icir.org/hilti/
[7] https://www.bro.org/sphinx/scripting/index.html
[8] http://try.bro.org/

user does not have to worry about memory management, which may be vulnerability prone while done manually.

With Suricata it is just the opposite. The developer of the new parser and its associated detection key words/functions enjoys the flexibility of C programming but shall be very cautious about security weaknesses.

- *Performance:* Given the flexibility offered by Bro to write sophisticated detection rules, for similar rules, it is expected to be slower than most of the traditional NIDS. Performance tests of Bro is not fully completed yet, they are available in the Bro performance benchmark sub-project[9].

In conclusion, Bro has been chosen mainly for two reasons: its ability to specify sophisticated detection rules and thus help detect complex attacks, and the development suite it provides to create a parser because it effectively reduces the amount of code to be supplied by the developer thus reducing the opportunity to introduce vulnerabilities and making it faster to review. Strengths and weaknesses of both NIDS are summarized in Table 4.1.

Table 4.1: Comparison of Bro and Suricata

Criteria	Bro	Suricata
Programming language	BinPAC, a high level language for describing protocol parsers, limiting the introduction of vulnerabilities	C language, which enjoys a wide user community: good for maintainability
Documentation	Detailed BinPAC suite documentation No documentation about new parser integration	Good documentation about existing parsers and sensors Documentation about new parser development is limited
Detection rules	Bro scripting language enables sophisticated detection rules Requires to get familiar with the Bro scripting language	Not easily upgradable Simple rules writing
Security of the IDS host	BinPAC development chain helps prevent introducing vulnerabilities Script language makes memory allocation transparent to the user, which is prone to vulnerabilities when done manually	Both the parser and the function for security rules are programmed with C, which requires caution
Performance	Bro scripting language allows complex detection rules but for simple rules, it is expected to be less performant than Suricata	

[9] https://www.bro.org/development/projects/benchmark.html

4.4.5 Bro packet processing chain

Bro is layered in two main components, an "event engine" and a "script interpreter" for a clear separation between mechanism and security policy. Bro operates as follows:

1. *Capture:* Bro captures the packet.

2. *Dispatch:* Bro reads the first bytes to identify the protocol and transmits the remaining bytes to the corresponding parser[10].

3. *Read:* The parser decodes the packet.

4. *Emit:* The decoded information is transmitted to an "event engine" that reduces a stream of packets to a stream of higher-level network events [117]. Event carries neutral information (what?) but no interpretation (why? meaning?). For each packet parsed, events generated, if any at all, are forwarded to relevant analyzers in the script interpreter.

5. *React:* When receiving an event it has subscribed to, an analyzer passes it to its event handler (Bro functions) to produce code to be analyzed by security script.

6. *Detect:* The analyzer executes the script commands, deciding if an anomaly has occured or not.

7. *Notify:* An alert is launched, using either emails or logs.

This Bro processing chain of network traffic stream is illustrated in Figure 4.10. Bro can parse n protocols and two analyzers are activated and reacts to the events produced by protocol 1 parser.

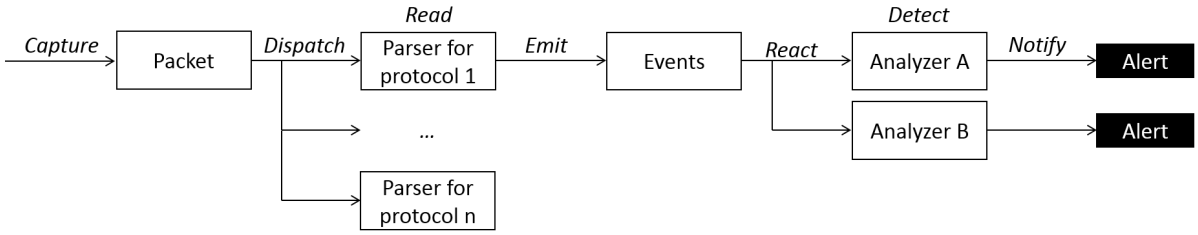

Figure 4.10: Bro processing chain

[10]Note that what is usually called a "parser" or "sensor" in literature is named "analyzer" in Bro documentation, while the "analyzer" from literature corresponds to a "sensor" in Bro documentation. In this dissertation we stick to the literature nomenclature.

4.4.6 Implementing the GOOSE intrusion detection

Considering the Bro processing chain described in previous section, integrating a GOOSE protocol intrusion detection module into Bro requires to provide a GOOSE parser and security scripts specifying anomaly detection algorithms.

The protocol parser is mostly written in the BinPAC language, the Bro parser language. As compilation of the BinPAC files generates C++ code, it is possible to integrate portions of code in C++, which is convenient to use existing C++ methods. The resulting clean and exception-friendly C++ code is then compiled along with the other parsers and the core of Bro. Since GOOSE is a link-layer protocol, coding its parser required to add code in Bro core source code, like the existing ARP protocol parser[11]. So, it is a big change. Merging it to the Bro project must be done after a thorough review and validation by the Bro project team.

Data carried by a GOOSE frame can have a recursive form with the type "array of data". An attacker may craft a GOOSE message with unusual amounts of encapsulated data: a data element of type array, whose first element is of type array, whose first element is of type array, and so on... It may cause a stack overflow of the running Bro instance. To avoid this kind of attacks, the part of the parser that handles the potential recursiveness of the messages must not rely on any recursive sets of methods. Using only BinPAC would have resulted into code with recursive methods, that is why we wrote it in C++.

4.4.7 Testing

Any merge request to the Bro project must come along with tests to verify whether the functionality has the intended effect and the new code does not compromise the stability of the existing version (non-regression tests). The idea is to make sure that no message would cause any malfunction. Among the GOOSE parser tests we provided, one displays the parser output, that is the decoded GOOSE frame fields. Its purpose is to demonstrate the parsing of correct and malformed GOOSE frames works properly. The other tests perform a "boundary-value analysis" of data items of the carried data set of types bit-string, integer, unsigned integer and array. Such an analysis is chosen when an exhaustive testing is not possible. It consists of feeding a function with input arguments taking in turn a value at the limit of the function specifications. While boundaries of a parameter are tested, the other ones take their typical values, far from their own boundaries. Data type "array" demands meticulous testing because of its recursive nature and because it was coded in C++ as explained in the previous section. The tests for "array" data type have two objectives: verify whether an item of an array can be of any type, including

[11] See Bro FAQ page at https://www.bro.org/documentation/faq.html#can-i-write-an-analyzer-for-that

"array", and check the parser robustness to malformed arrays. Details about testing are given in Appendix F.

4.4.8 Source code

The parser source code has been submitted as a pull request to the Bro project and is currently under study by the Bro team. GOOSE parser source code is available from the request web page[12].

4.4.9 Detection scripts

Bro detection scripts are written in *Bro scripting language*, an object-oriented programming language with type inference and automatic memory management.

A single event is generated for GOOSE protocol, *goose_ message* each time a GOOSE frame is parsed. The argument *pdu* contains the frame fields. The detection analyzer whose code is given by Algorithm 2 has subscribed to this event. Each time it receives it, the values of the fields *stNum*, *sqNum* and *datSet* are stored in the dictionnary *lastInfoOf* whose keys are *datSet*.

In the script example, an alert is launched when current state number is less than the previous one, or when state number is unchanged but current sequence number has not been incremented, as detailed in Algorithm 2.

4.4.10 Performance

To evaluate the performance of the proposed intrusion detection implementation, we use an instance of Bro with our GOOSE parser and the simple detection script given in Algorithm 2. The traffic to be replayed is a trace captured during the experiment demonstrating feasibility of false GOOSE injection, presented in section 4.1.3.

Among IDS performance metrics introduced in 1.4.1.1, some are relevant for this performance evaluation:

- the rate of packets dropped by the running Bro instance,
- the True Positive Rate – TPR,
- the processing time by the GOOSE parser and event generator, covering steps 1 to 5 of the Bro processing chain introduced in section 4.4.5,

[12] https://github.com/bro/bro/pull/76

Algorithm 2 Bro script for detection of the "State and Sequence numbers attack"

```
@load
# Declare a one class with two integer objects
type DataSetLastInfo : record {
    stNum : count ;
    sqNum : count ;
} ;
# A dictionnary whose keys are the datSet values
# and the values are the state and sequence numbers
global lastInfoOf : table[string] of DataSetLastInfo ;

# This detection analyzer has subscribed to the following event
event goose_ message (info : GOOSE : :PacketInfo, pdu : GOOSE : :PDU)
# Every time the analyzer reveives this event, the following instructions are executed
{
    # If a frame with the same datSet has already been analyzed:
    if(pdu$datSet in lastInfoOf)
    {
        local lastInfo = lastInfoOf[pdu$datSet] ;

        # Check stNum
        if(pdu$stNum < lastInfoOf$stNum
        {
            print fmt("State number inconsistency for GOOSE data set %s",
pdu$datSet) ;
        }
        # Check sqNum
        else if(lastInfoOf$stNum == pdu$stNum && pdu$sqNum ≤ lastIn-
foOf$sqNum)
        {
            print fmt("Sequence number inconsistency for GOOSE data set %s",
pdu$datSet) ;
        }
    }
    # Store current state and sequence numbers values
    lastInfoOf[pdu$datSet] = [$stNum = pdu$stNum, $sqNum = pdu$sqNum] ;
}
```

- the processing time by the Bro analyzer alone,
- and the total processing time.

All these five metrics are given as functions of the throughput of GOOSE messages flowing through the managed interface. The last three criteria are relevant only for throughput that Bro can handle. If packets are dropped, the detection is not accurate any more.

4.4.10.1 Experimental set up

Experimental set up is depicted in Figure 4.11. It consists of a virtual machine running three programs:

- An instance of Tcpreplay, a Linux pcap editing and replaying tool, which replays a trace of 43 161-byte-long packets, 5,000 times in a row, on loopback interface at rates ranging from 100 pps to 5,000pps. Tcpreplay generates logs with actual replaying rates in packets per second (pps) and bytes per second (bps).
- An instance of Bro with our GOOSE parser and the simple detection script given in Algorithm 2, which monitors local host interface. Time entering detection script, time exiting detection script, number of packets parsed, number of alarms are stored in a log file.
- An instance of Tshark, a network traffic analyzer, sniffing traffic on local host interface. Log gives times that packets hit local host interface.

Performance tests were run on a Debian GNU/Linux 8 64-bit virtual machine with 7.8GB memory, an Intel Core i7-5600U 2.60GHz processor. During experiment, we did not run any other program to avoid interferences, neither on the virtual machine nor the host machine.

4.4.10.2 Results

Bro crashed for a rate of 4,678pps that is 753,158bps. However, for rates between 100pps and 4,678pps, Bro did not drop any packet and performed a 100% TPR.

From Figure 4.12, we can see that Bro analysis capabilities are degraded for a rate greater than 3,700pps. Processing times and throughputs are given in Tables 4.2 and 4.3, respectively.

A GOOSE application may publish up to one packet every millisecond and a GOOSE message length is limited to 1,500 bytes at most. Thus in the worst case, a GOOSE application generates 150,000 bytes of traffic every second. These figures mean that a

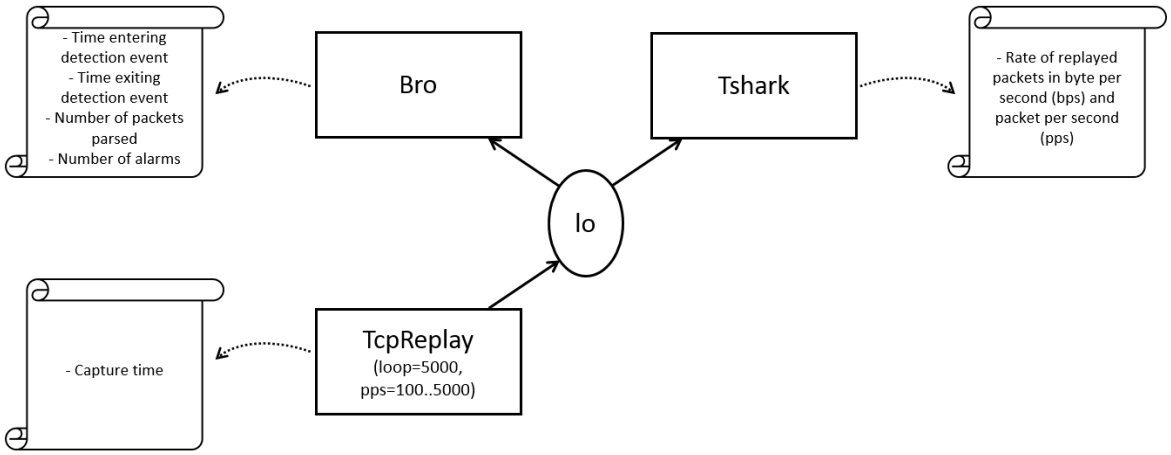

Figure 4.11: Experimental set up for Bro performance evaluation

Table 4.2: Bro processing times

Rate	**100 to 3,700pps 161,000 to 595,700bps**	**3,700 to 4,678pps 595,700 to 753,158bps**
Total analysis time	2.3ms	12.7ms

single instance of Bro, running the simple detection script given in algorithm 2 would be able to monitor up to 4 GOOSE applications in the most critical conditions (600,000bps) for an analysis time of 2.3ms. This number comes to 40 for 150-byte long messages with a publishing rate of 1,000pps.

4.4.10.3 Related works

Lin et al. [102] also make use of Bro for implementing a NIDS dedicated to ICS in power utility domain, DNP3. They obtain an analysis throughput of 9427pps with packets of a 264-byte length in average. Compared to these results, ours seem mediocre. Except the protocol, the main difference between their implementation and ours is that they created as many events as there are data fields to manage, while we created only one, whose output is a list of all the fields of interest. We wonder if that may significantly impact the Bro chain processing performance.

Another work that is interesting to mention is the IDS proposed by Y. Yang et al. [147]. The tool used for implementation is not Bro but protocols covered are GOOSE, SV and MMS. The authors claim a 100% accuracy and an analysis time of less than 0.3ms with 32 detection rules. We guess it is a time per packet but it is not specified in the paper and that would mean an analysis throughput of more than 3,300pps. Our results are poor compared to this work: analysis time of our implementation is eight times longer and we only checked a single detection rule. But the experience as described in [147] does not provide enough detail about the experimental set-up and methodology.

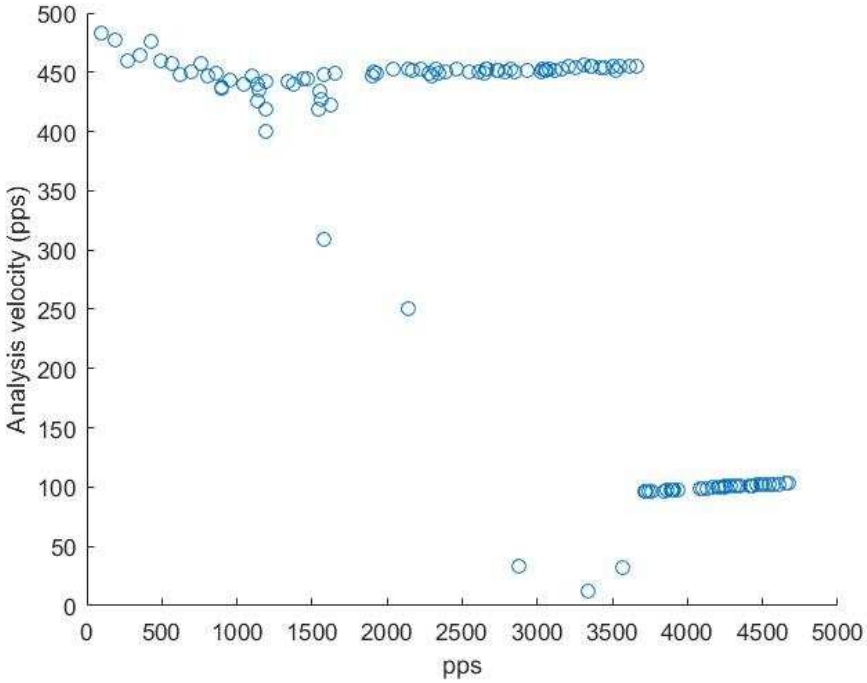

Figure 4.12: Total analysis (parsing + detection) throughput as a function of pps

Table 4.3: Bro processing throughputs

Rate	**100 to 3,700pps** **161,000 to 595,700bps**	**3,700 to 4,678pps** **595,700 to 753,158bps**
Total analysis throughput	444pps 71,564bps	95pps 15,227bps

The information missing that would be necessary for a comprehensive comparison of both works is: hardware characteristics, message size and throughput of captured traffic.

4.5 IEC 61850 SCADA architecture resilient to GOOSE attacks

One of the objective of the smart grid identified by the U.S. Department of Energy is to operate resiliently to cyber attacks and natural disasters [104]. Indeed, reducing cyber risk may be achieved either by reducing occurrence likelihood or limiting consequences. Both shall be covered but as "zero risk" do not exist, confining possible consequences regarding both their scope and severity must be a leitmotiv while designing IACS. Following this idea, we propose an IEC 61850 automation system architecture resilient to GOOSE attacks [87].

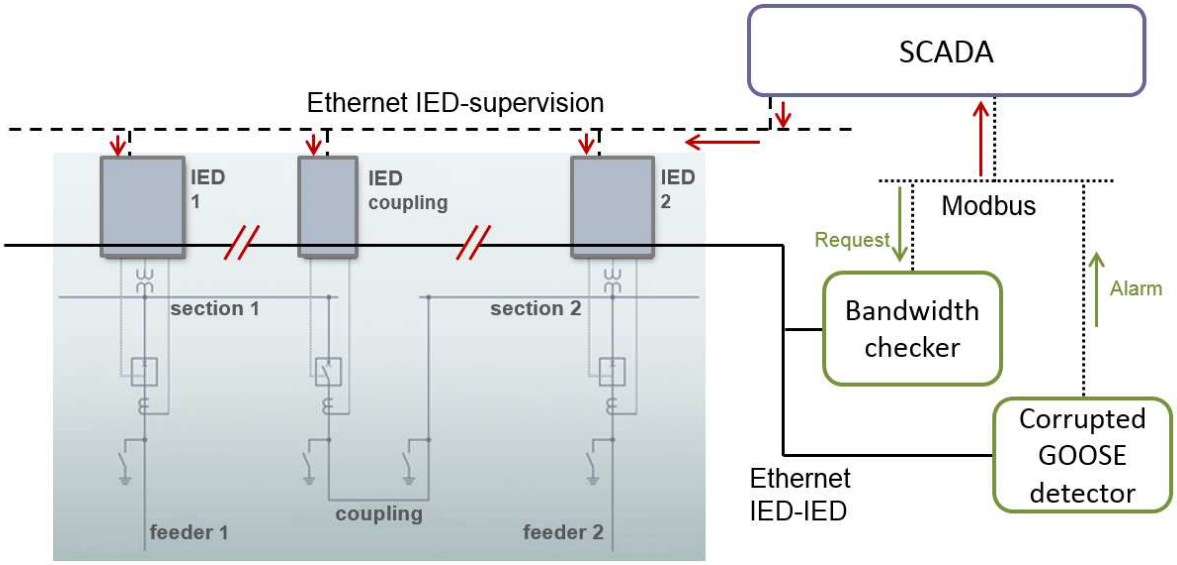

Figure 4.13: IEC 61850 automation architecture resilient to attacks on GOOSE communication

4.5.1 Example electrical system

The global architecture is shown in Figure 4.13. The electrical system considered in this study is a coupling of two busbars. Each of the two sections feeds several transmission lines (which do not appear on Figure 4.13) with its own generator. In normal mode the coupling switch is open. In case of a fault on the line powering section 1, this one is no longer supplied. Automatic switching shall allow generator 2 to take over the power feeding of section 1, by first opening generator 1 circuit breaker to isolate the fault and then closing coupling circuit breaker. In this architecture, the circuit breakers all have their own IED providing over current protection. In case of logical selectivity (see section 4.1.3.1 introducing the basics of electrical protection), the closing of the coupling circuit breaker is triggered by a signal sent by IED 1, which has detected the fault, to the coupling IED through GOOSE communication.

4.5.2 Communication system architecture

The whole concept relies on two cyber security modules reporting to the SCADA. One is responsible for measuring the bandwidth to detect attacks such as Ethernet storm. The second one integrates the detection rules for GOOSE protocol we developed (presented in section 4.3.

Using *ifstat*, a bandwidth monitor available in Linux, we measure the bandwidth, both instantaneous and average over a defined window time configurable by the user. The output is fed to the SCADA via a Modbus / TCP server. This echoes the security measure consisting of monitoring devices and resources availability, identified in IEC 62351

(see section 1.2.2).

The GOOSE frame verifier is based on tcpdump, an open-source packet analyzer. Our code for GOOSE parsing and verifying of the GOOSE transfer mechanism (as presented in section 4.3.3) was added into the core of a tcpdump instance.

The two cyber detectors send their analysis results to SCADA through two different mechanisms: SCADA periodically gets analysis results from the bandwidth checker through polling, while GOOSE frame verifier sends its alarms using a Modbus/TCP client-server mechanism. The idea is to not overwhelm the network with too many messages. Supervision forwards alarms to the IEDs to make them switch from normal mode (high confidence level in GOOSE communication) to a safe mode, based on an alternative program and information received from SCADA.

The proposed communication architecture is based on the following assumptions: (i) every communication links has its own independent network to prevent contamination of "clean" connections by a compromised network (SCADA – IEDs (MMS), IED – IED (GOOSE), network analyzer modules – SCADA (Modbus)), and (ii) vertical communication (SCADA – IEDs) is supposed reliable and secure.

4.5.3 Normal program vs safe program

In normal mode, upon receiving a GOOSE message, the IED triggers a timer. This delay allows the cyber security modules to check the communication integrity. If there was no alarm from the SCADA after detection time has expired then the IED takes into account the received information and process its normal program. This is conceivable because total operating time of electrical protection functions are of the order of 100ms or even 1s [34] whereas the total transfer time of a GOOSE message shall be 4ms maximum. If a cyber anomaly is detected and reported to the SCADA in the meantime, it sends a signal to the IEDs (red arrows in Figure 4.13) to make them enter safe mode: an alternative strategy takes over the real-time communication. This alternative strategy relies on specific programs and communication with the SCADA. The IED thus stops taking into account GOOSE communication (as symbolized by red double slashes in Figure 4.13) and executes its safe program as long as the cyber alert holds.

Regarding implementation, switching from one mode to the other is simple. The difficulty lies in the design of the safe function. It comes out from a thorough fault mode analysis (FMEA or other risk assessment methods as presented in section 2) of the considered system. For Figure 4.14 scenario, the safe mode function is rather simple: the coupling circuit breaker being in open position, if an anomaly is detected for the GOOSE connection carrying closing signal from IED 1 to coupling IED, the safe program consists in not closing the coupling circuit breaker. The system goes back to normal mode as soon as the SCADA disables the alarm.

Further research on this topic of a resilient automation architecture would require

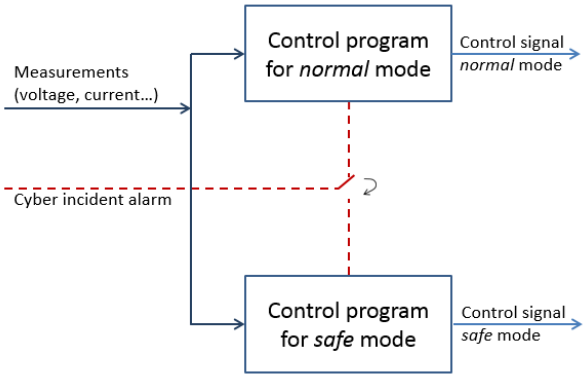

Figure 4.14: Double programming of IEDs in an IEC 61850 automation architecture resilient to GOOSE attacks

addressing the impact of the global system performance to verify whether the safety objectives related to electrical protection are not compromised.

Conclusion

This chapter focused on the concrete contributions of this work. We first demonstrated that off-the-shelf IEDs are vulnerable, first because of the lack of security of IEC 61850 protocols, and specifically the GOOSE protocol, and second because implementation of the GICS IEDs do not fully comply with standards requirements. We then exposed how valuable configuration files are when talking of behavior-based anomaly detection. Standard specifications (as long as they are followed by real-world implementations) help design general specifications of the IDS, while configuration information from SCL files helps automatically tune the specifications to tailor the IDS to the ICS to monitor. We presented the rules built from this knowledge. For the implementation, we have integrated a GOOSE parser to an existing widely used network traffic analyzer Bro. Performance evaluation showed limited results: our solution seems doable and useful for environments not too large but not scalable to complex systems. Although improvements are necessary, we conclude this chapter by proposing an IEC 61850 ICS architecture resilient to GOOSE attacks, thus putting the contributions of intrusion detection into a realistic context.

Conclusion and further work

This dissertation has presented a PhD project work tackling intrusion detection of GOOSE protocol in communication networks of IEC 61850 automation and control systems, through an deterministic anomaly-based approach.

This work being intrinsically tied to the IEC 61850 standard, we have tried to provide the reader with the necessary knowledge. A rough introduction to the standard is done in section 1.2.1, while more details are given in Chapter 3 with information about the communication architecture, specificities of the GOOSE protocol and the abstract IEC 61850 data object model.

The question of OT cyber security is rather recent and at the junction of two domains traditionally unrelated: cyber security that concerned IT almost exclusively until now, and IACS. It has been identified as a very serious topic by governments and international experts institutions, especially for CI such as the power grid. It thus was necessary to review the documentation issued by these bodies in order to catch the context of this study and the concepts at stakes. Section 1.2 gives an overview of the relevant international standards while prescriptions from standards and governmental guides regarding cyber risk management of ICS are reviewed in section 2.3.

The novelty of the field also explains the extensive state of the art one can find in this dissertation, which devotes two chapters out of four to it. Chapter 1 is about intrusion detection in IACS and more specifically in the context of the power grid. It defines the context of the study, main concepts of smart grid automation and intrusion detection and reviews papers. Chapter 2 focuses on cyber security risk assessment of IACS, reviewing methods from the two fields Dependability and IT, and research papers. In both chapters, we tried to catch the IACS specificities compared to IT. Thus, one can find remarks resulted from our analysis and/or quoted from literature, especially in sections 1.3, 2.2 and 2.4.

Based on this literature review, we addressed anomaly-based intrusion detection of IEC 61850 GOOSE communications from both abstract and concrete angles: a data model tallying with the IEC 61850 standard framework and an implementation proposition, presented in Chapters 3 and 4 respectively. The proposed approach was built based on the understanding of the cyber risk encountered by an IEC 61850 substation that we got from a risk assessment, including a demonstration of GOOSE intrusion feasibility. Risk assessment procedure and outcomes are detailed in the first section of Chapter 3 (section 3.2) while the study of GOOSE communication vulnerabilities is presented in Chapter 4 (section 4.1), whose orientation is more experimental compared to the previous chapter.

The proposed behavior-based anomaly detection is deterministic as it makes use of both standard specifications and SCL configuration files. The resulting proposed rules for GOOSE protocol are explained in section 4.3. A GOOSE parser was integrated into

Bro, a popular network traffic analyzer, and an example of detection script was presented. Results from the performance testing are limited and our implementation may need improvement in order to be usable in a real IEC 61850 ICS (section 4.4).

Intrusion detection being just one tool, it is not sufficient per se to secure IACS and it must be part of a global security framework. We tried to adopt a wider point of view and proposed an IEC 61850 control and automation architecture resilient to GOOSE intrusions. This proposition concludes Chapter 4 and is a first step to a global security solution. Further research is needed to answer the need for secure IACS in IEC 61850 environments and we suggest a few research tracks below.

Further work

Extending anomaly detection to other IEC 61850 stack protocols

The proposed anomaly detection function focuses on the GOOSE protocol, regarding both its IEC 61850 information model-compliant specification and its implementation. We estimate that it can rather easily be adapted to SV protocol given the similarities of both protocols: SV messages are periodically broadcasted on a publisher-subscriber basis and counters help follow sequences of frames. As stressed by the risk assessment of a generic substation proposed in Chapter 3, SV communications may be very critical to the system integrity depending on the application implementation. We deliberately discarded SV protocol from our work, though, because we do not have material means to study them: G-ICS platform has no device running SV protocol. Complementary work is required to integrate anomaly detection for MMS protocol. It is a connection-based protocol, built over TCP/IP with real-time constraints much less stringent than for the two others (see section 3.1). Thus, MMS communications may hold a bit more complexity than with GOOSE and SV protocols. MMS-based reporting mechanism is well specified and all the required tuning information can be extracted from configuration files. But control commands issued by human operators introduce some flexibility if not randomness and thus they cannot be as precisely characterized using only system configuration. Further study is required here to find other methods to specify MMS communications behavior.

Correlation and leveraging process knowledge

An interesting research topic would be correlation of outputs of several analyzers. First, it would be necessary to compare semantic content of the GOOSE, SV and MMS messages to check whether they are relevant to each other and regarding the sequence of previous messages. This would enable to leverage knowledge of the physical process such as electrical protection functions and monitoring of the system. For instance MMS command signals may be compared to measurements carried by SV frames to make sure

they would not cause an undesired system state.

A further level of correlation would make use of other information sources, such as logs available in the ICS and outputs of existing tools (e.g. for dependability purposes).

Towards a global architecture

Talking of correlation brings us to another research track: correlation requires to think about a global deployment architecture. How to deploy detection sensors to optimize resources (computational power of involved devices and amount of data flows)? What information shall be sent up to a SIEM? Are prevention actions possible at some points of the system?

Improvement of the implementation

As shown by the performance evaluation in section 4.4.10, the proposed implementation with Bro would be usable for limited systems with few GOOSE applications and short GOOSE messages. But it is not scalable to extreme conditions: large systems with a lot of GOOSE applications and/or with boundary configurations as defined in the IEC 61850 standard. It would be necessary to seek for optimization of the existing implementation into Bro or explore other implementation options to meet real-time requirements of IEC 61850 power utility ICS.

Confronting real-world

In order to be truly pertinent to real-world IEC 61850 environments, it would be necessary to confront the proposed solutions using data from real electrical substations. It would first help capture real systems specificities and eventually validate solutions.

Appendix A

ANSSI Classification Method and Key Measures

ANSSI classification method flowchart is shown in Figure A.1.

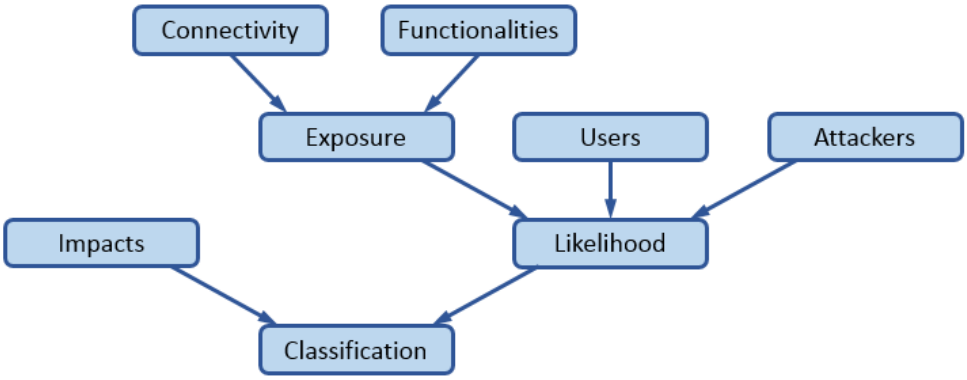

Figure A.1: Diagram of ANSSI classification method [8]

- After system perimeter definition, first step is to determine its degree of *Functionalities* according to the complexity of its components. They can be "minimal systems" (e.g. sensors, actuators, I/O, PLCs...), "complex systems" (e.g. SCADA, local databases...) or "very complex systems" (engineering stations, centralised databases...).

- In parallel, ICS's *Connectivity* has to be evaluated. The system is either "isolated", "connected to a Management Information System", "using wireless technology" or "distributed with private infrastructure of permitting operations from outside".

- The combination of the two aforementioned criteria yields the *Exposure* ranging between level 1 (the least exposed) and level 5 (the most exposed).

- Accessibility of the ICS is determined depending on whether *Users* are trained and aware of cybersecurity stakes and on whether authentication and logging are implemented. The four ensuing categories are "authorized, certified and controlled user", "authorized and certified user", "authorized user" and "unauthorized user".

- Five levels are proposed to render the *Attacker*'s level of skills: "non-targeted" (such as viruses), "hobbyist", "isolated attacker", "private organization" and "state organization". This classification is quite common [30].
- From the three previous criteria, *Likelihood* is calculated.
- Estimating *Impact* on integrity and availability of risks concerned with human impacts, environmental impacts and impacts from interruption of the service. A five-level scale is used from "insignificant" to "catastrophic".
- Finally, association of "Likelihood" and "Impact" gives the security class the system as "low", "significant" or "critical".

Appendix B

IEC 61850 data objects related to security

B.1 LN: Generic security application (GSAL)

Logical Node "Generic security application" is the system and device security LN, whose description is given in IEC 61850:5 as "Containing logs about security violations." [74]. Its specification is given in IEC 61850:7.4 as recalled in Table B.1 along with the textual description "This node shall be used to monitor security violations regarding authorisation, access control, service privileges and inactive associations." [71].

Table B.1: GSAL Class Table

GSAL class			
Data Attr Name	Attribute Type	Explanation	M/O
Common Logical Node Information			
		LN shall inherit all Mandatory Data from Common Logical Node Class	M
OpCntRs	INC	Resetable Security Violation Counter	M
Controls			
NumCntRs	INC	Number of counter resets	M
Status Information			
AuthFail	SEC	Authorization failures	M
AcsCtlFail	SEC	Access control failures detected	M
SvcViol	SEC	Service privilege violations	M
Ina	SEC	Inactive associations	M

B.2 Security Violation Counting (SEC) Common Data Class specification

Since many Data in IEC 61850 catalogs use the same attributes, they have been collected for re-use in Common Data Classes (CDC), which are defined in IEC 61850-7-3 [70]. They give templates for LNs' data types, listing Data Attributes with their basic

types and the Services associated to the CDC. A LN's data objects are grouped according to their purpose: status information, measurand information, controls, status settings, analogue information, description information. CDCs from one of these data categories usually have common Services, which are specified once in the standard. Tables detail Data Attributes for each CDC.

The "Security Violation Counting (SEC)" common data class is related to status information. Its specification is replicated in Table B.2. Each attribute has a name, a type, a functional constraint, a trigger option, a value/value range, and an indication of whether the attribute is mandatory, optional or subject to conditions.

Services related to status information common data classes are given in the basic status information template [70]. Among them "GetDataValues", "GetDataDefinition", "GetDataDirectory", "GetDataSetValues", "Report" apply to all SEC Data Attributes and "SendGOOSEMessage" apply to SEC Data Attributes with functional constraint ST.

Functional constraint ST stands for Status information. The value of a data attribute with this functional constraint may be read but shall not be written. Functional constraint DC "Description" is for data attributes whose value may be read and written. Data attributes with FC = EX, "Extended information", shall represent an extension information providing a reference to a name space and shall not be writeable. More information are available in IEC 61850-7-2 [69].

Table B.2: Security Violation Counting specification as given in IEC 61850 standard

SEC class					
Attribute Name	Attribute Type	FC	TrgOp	Value/Value Range	M/O/C
DataAttribute					
status					
cnt – Counter value of security violations.	INT32U	ST	dchg – data-change		M[1]
sev – Severity of the last violation detected.	ENUMERATED	ST		- unknown: Severity cannot be determined. - critical: Severity is critical in terms of safe operation or data considered critical and privileged access was attempted. - major: Severity is major in terms of safe operation or data considered of major importance and privileged access was attempted. - minor: Severity is minor in the sense that access control was denied to data considered privileged. - warning: Is less severe than minor.	M
t – Time	TimeStamp	ST			M
addr – Address of the remote source that last caused the count to be incremented.	OCTET STRING64	ST			O[2]
addInfo – Additional information that may give further clarification as to the last detected violation.	VISIBLE STRING64	ST			O
configuration, description and extension					
d – Textual description of the data.	VISIBLE STRING255	DC		Text	O
dU – Textual description of the data using unicode characters.	UNICODE STRING255	DC			O
cdcNS – Common data class name space.	VISIBLE STRING255	EX			AC_DLNDA_M[3]
cdcName – Common data class name.	VISIBLE STRING255	EX			AC_DLNDA_M
dataNS – Data name space.	VISIBLE STRING255	EX			AC_DLN_M[4]
Services					
As defined in basic status information template [70]					

1. Attribute is mandatory.
2. Attribute is optional.
3. The attribute shall be present, if the name space of the CDC deviates from either the name space defined in ldNs/lnNs or the name space defined in dataNs, or both.
4. Applies to dataNs in all CDCs, dataNs shall be present if the name space of the DATA deviates from the name space defined in ldNs/lnNs.

Appendix C

Specification of anomaly detection LNs and PICOMs

This appendix presents the specifications of all logical nodes that compose the function "Anomaly detection" described in Chapter 3, along with their associated PICOMs. For the sake of clarity, the same convention as in Chapter 2 is used: the data objects that we have created for the purpose of our cybersecurity function are in italic type.

C.1 Textual description of LNs involved in the "Anomaly detection" function

Every logical node specification must start with its textual description. Table C.1 recalls textual description of all LNS involved into anomaly detection function. The names of newly created LNs dedicated to anomaly detection are given *in italics*.

The "Starting criteria" column identifies the starting criteria launching the LN function and other inputs of the LN from a communication point of view if relevant.

Table C.1: Description of LNs involved in the Intrusion Detection function

LN name	LN acronym	Grouping	Task description and context of execution	Starting criteria
Sniffer	*CYSN*	System and device security	*Sniff all GOOSE frames OR Sniff messages and identify protocol.*	Start of automation system.
Analyzer	*CYAN*	System and device security	*Extract relevant features from received packets. Detect if there is an anomaly in packet structure that makes impossible to compute required features.*	Reception of a message.
Model Checker - Single Frame	*CYMdlSgl*	System and device security	*Check application-related criteria such as control operation, value range...*	Reception of a message.
Model Checker - Many Frames	*CYMdlMany*	System and device security	*Check whether the extracted features match the model of the system normal behavior, such as correlation of successive commands and state information...*	Reception of a message.
Communication Checker - Single Frame	*CYComSgl*	System and device security	*Check conformance of a single frame with protocol specification: packet structure, source address, destination address, message identifiers (GOOSE: APPID, GoCBRef, GoID), number of data entries...*	Reception of a message.
Communication Checker - Many Frames	*CYComMany*	System and device security	*Check whether the communication features of the received messages match the system communication configuration: message succession counters (GOOSE: Sequence and State Numbers SqNum, StNum), transmission time, reception time, frequency of messages...*	Reception of a message.

Alarm application	*CYAL*	System and device security	*Deal with alarm generation / reporting from the checking LNs to management LNs (Control + Interface). May be used for local correlation if relevant.*	PICOM reception from sources LN.
Alarm handling (creation of group alarms and group events)	CALH	Control	For the communication, there is no difference between alarms and events if a time tag is added to any data transmitted. If several events or alarms have to be combined to group alarms, a separate, configurable function is needed. The related LN may be used to calculate new data out of individual data from different logical nodes. Remote acknowledgement with different priority and authority shall be possible. The definition and handling of alarms is an engineering issue.	PICOM reception from sources LN.
Operator interface (control local at bay level - control at station level)	IHMI	Interface, logging, and archiving	1) Front-panel operator interface at bay level to be used for configuration, etc. and local control. 2) Local operator interface at station level to be used as workplace for the station operator. The role of the different HMI is not fixed for most of the functions and is defined in the engineering phase.	Start of automation system. PICOM reception from source LN.
Remote control interface or telecontrol interface	ITCI	Interface, logging, and archiving	Telecontrol interface to be used for remote control from higher control level. Basically, the TCI will communicate the same data as the station level HMI or a subset of these data. The role of the different interfaces is not fixed for most of the functions and is defined in the engineering phase.	Start of automation system. PICOM reception from source LN.

Remote monitoring interface or telemonitoring interface	ITMI	Interface, logging, and archiving	Telemonitoring interface to be used for remote monitoring and maintenance using a subset of all information available in the substation and allows no control. The role of the different interfaces is not fixed for most of the functions and is defined in the engineering phase.	Start of automation system. PICOM reception from source LN.
Archiving	IARC	Interface, logging, and archiving	Archiving to be used as sink and source for long-term historical data, normally used globally for the complete substation on station level.	PICOM reception from source LN.

C.2 LN: Communication Checker - Many Frames (CYComChkMany)

Logical node "Communication Checker - Many Frames" shall monitor violations of communication features that involve many frames, either regarding the protocol definition or communication configuration. For GOOSE protocol, CYComChkMany shall check messages succession (sequence and state counters), transmission time, transfer time scheme, frequency of messages, etc. All these criteria are given as status information data elements in CYComChkMany specification. To set the window length of cross-packet examination, a data element is defined under "Settings" section as WinLgth of type "ING (Integer status setting)" (see IEC 61850-7-3 [70] for complete definition). This common data class makes available an attribute "setVal", under which one can configure an integer value. Unit shall be chosen at configuration time to be either a number of messages or a time unit (ms for instance).

Table C.2: CyComChkMany Class Table

CYComChkMany (Communication Checker - Many Frames)			
Data Attr Name	Attribute Type	Explanation	M/O
Common Logical Node Information			
Mod	INC (Controllable integer status)	Mandatory Data from Common LN class	M
Beh	INS (Integer Status)		M
Health	INS		M
NamPlt	LPL (LN name plate)		M
OpCntRs	INC	Resetable Security Violation Counter	M
Controls			
NumCntRs	INC	Number of counter resets	M
Status Information			
MinViol	SEC	*Reception of too few messages in total from a specific source (calculated over the time window length WinLgth)*	O
MaxViol	SEC	*Reception of too many messages in total from a specific source (calculated over the time window length WinLgth)*	O
CountAlm	SEC	*Inconsistency in state and sequence numbers incrementation*	O
TxAlm	SEC	*Inconsistency of transmission scheme: retransmission times (in stable conditions T0, after an event T1 and next till achieving stable conditions T2, T3) diverge from the ones configured*	O
Settings			
WinLgth	ING (Integer status setting)	*Window length for GOOSE frames examination*	M
Services			
GetLogicalNodeDirectory			
GetAllDataValues			

C.3 LN: Model Checker - Single Frame (CYMdlChkSgl)

LN "Model Checker - Single Frame" shall check system application-related criteria such as a variable's (control command or value setting) origin, its value/value range, etc.

Table C.3: CyMdlChkSgl Class Table

CYMdlChkSgl (Model Checker - Single Frame)			
Data Attr Name	Attribute Type	Explanation	M/O
Common Logical Node Information			
Mod	INC (Controllable integer status)	Mandatory Data from Common LN class	M
Beh	INS (Integer Status)		M
Health	INS		M
NamPlt	LPL (LN name plate)		M
OpCntRs	INC	Resetable Security Violation Counter	M
Controls			
NumCntRs	INC	Number of counter resets	M
Status Information			
AddrAlm	SEC (Security violation counting)	*Variable or its value is not supposed to be sent from this source address.*	M
ValAlm	SEC (Security violation counting)	*Value of a variable of interest (control variable or any variable that led to the considered message aplication definition) carried by the message is out of acceptable range.*	M
Services			
GetLogicalNodeDirectory			
GetAllDataValues			

C.4 LN: Model Checker - Many Frames (CYMdlChkSgl)

LN "Model Checker - Many Frames" shall check whether the messages' features match the model of the system normal behavior, such as correlation of successive commands and pieces of state information, etc. Criteria are given as status information data elements in CYMdlChkMany specification. To set the window length of cross-packet examination, a data element is defined under "Settings" section as WinLgth of type "ING (Integer status setting)" (see IEC 61850-7-3 [70] for complete definition). This common data class makes available an attribute "setVal", under which one can configure an integer value. Unit shall be chosen at configuration time to be either a number of messages or a time unit (ms for instance).

Table C.4: CyMdlChkMany Class Table

CYMdlChkMany (Model Checker - Many Frames)			
Data Attr Name	Attribute Type	Explanation	M/O
Common Logical Node Information			
Mod	INC (Controllable integer status)	Mandatory Data from Common LN class	M
Beh	INS (Integer Status)		M
Health	INS		M
NamPlt	LPL (LN name plate)		M
OpCntRs	INC	Resetable Security Violation Counter	M
Controls			
NumCntRs	INC	Number of counter resets	M
Status Information			
ValAlm	SEC (Security violation counting)	*Value of the considered variable is inconsistent considering previous messages received and/or the present state of the system. (E.g. would put the system in a bad state ?)*	M
CtlAlm	SEC	*Control command not relevant at this time regarding previous commands and previous states*	M
Settings			
WinLgth	ING (Integer status setting)	*Window length for GOOSE frames examination*	M
Services			
GetLogicalNodeDirectory			
GetAllDataValues			

They may be one LN CYmdlChkMany per protocol and one for many or all of the IEC 61850 protocols, according to the security objectives that have been determined.

C.5 LN: Alarm application (CYAL)

LN "Alarm application" shall deal with alarm generation / reporting from the checking LNs to management LNs (Control and Interface). More precisely, this LN shall be used to aggregate analysis result from checking LNs, thus doing some correlation locally. According to processing capabilities of hosting device, this LN may or may not be implemented.

Table C.5: Alm Class Table

CYAlm (Alarm application)			
Data Attr Name	Attribute Type	Explanation	M/O
Common Logical Node Information			
Mod	INC (Controllable integer status)	Mandatory Data from Common LN class	M
Beh	INS (Integer Status)		M
Health	INS		M
NamPlt	LPL (LN name plate)		M
OpCntRs	INC	Resetable Security Violation Counter	M
Controls			
NumCntRs	INC	Number of counter resets	M
Status Information			
Alm	SEC (Security violation counting)	*Compiling and processing analysis results from checking LNs for a first step of correlation.*	O
Services			
GetLogicalNodeDirectory			
GetAllDataValues			

C.6 Logical nodes and their related PICOMs

All PICOMs involved in the cybersecurity function dedicated to anomaly detection are either alarms or analysis reports, as defined in Tables 3.3 and 3.4 respectively. Sources and sinks are given in Table C.6.

Table C.6: PICOMs of source cyber security related LNs

LN	PICOM Name	Source	Sink 1	Sink 2	Sink 3	Sink 4	Sink 5
Communication Checker - Single Frame		***CyComChkSgl***					
	Message structure alarm	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Protocol value alarm	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Protocol co-values alarm	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Address alarm	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	ID alarm	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Configuration value alarm	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Analysis result	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
Communication Checker - Many Frames		***CyComChkMany***					
	Minimum number of messages alarm	*CyComChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Maximum number of messages alarm	*CyComChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Message counters alarm	*CyComChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Transmission scheme alarm	*CyComChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Analysis result	*CyComChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
Model Checker - Single Frame		***CYMdlChkSgl***					
	Source address alarm	*CyMdlChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Model value alarm	*CyMdlChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Analysis result	*CyMdlChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
Model Checker - Many Frames		***CyMdlChkMany***					

	Model co-values alarm	*CyMdlChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Control command alarm	*CyMdlChkMany*	*CYAL*	CALH	IHMI	ITCI	ITMI
	Analysis result	*CyMdlChkSgl*	*CYAL*	CALH	IHMI	ITCI	ITMI
Alarm application		***CYAL***					
	Anomaly alarm	*CYAL*	CALH	IHMI	ITCI	ITMI	

Appendix D

Protection functions of the example transmission substation, small size, first topology by IEC 61850 standard

Functions implemented in the example transmission substation, small size, first topology T1-1 (see Figure 3.6) proposed by the IEC 61850 standard and taken as a use case for the risk analysis of section 3.2 are explained in this appendix. The ANSI/IEEE standardized code corresponding to each function is recalled, along with the associated IEC 61850 LNs [60]. LN first letter indicates the functions group: P stands for protection, R for protection-related and C for control.

- **Breaker failure**: Also known as "backup protection". An instantaneous overcurrent relay functions instantaneously on an excessive value of current or on an excessive rate of current. In case of a breaker failure, the fault is not cleared. Therefore, neighboring breakers have to be tripped.

  ANSI/IEEE code: 50BF, IEC 61850-7.4 LN: RBRF

- **Buchholz relay overload protection**: Oil-filled transformers may be subject to an overload or an electric arc that both cause oil to vaporize. A Buchholz relay mechanically detects such a gaz accumulation first activating an alarm and secondly, if gaz keeps rising beyond the limit, triggering a circuit breaker to isolate the faulty transformer.

  ANSI/IEEE code: 49, IEC 61850-7.4 LN: PTTR (Thermal overload)

- **Differential protection**: If output current of the protected zone is different from input current (amplitude or phase), there is an internal fault. This protection trips circuit breakers at both ends of the protected zone.

  ANSI/IEEE code: 87, IEC 61850-7.4 LN: PDIF

- **Distance protection**: A distance relay operates depending on the impedance between the fault location and the relay location. Each instance of a distance protection covers a bounded zone of the line.

  ANSI/IEEE code: 21 (Distance relay), IEC 61850-7.4 LNs: PDIS (Distance) + PSCH (Protection scheme)

- **Distance protection with automatic recloser**: Reclosing function specifically aims at eliminating transient or semipermanent faults while limiting service cut-off. Orders for reclosing the circuit breaker(s) are automatically generated after a configured time delay has passed to let isolation to regenerate.

  ANSI/IEEE code: 21 + 79, IEC 61850-7.4 LNs: PDIS (Distance) + RREC (Autoreclosing)

- **Interlocking**: Any kind of logic allowing and preventing opening disconectors and/or circuit breakers according to the states of one or many other isolating devices.

  ANSI/IEEE code: 3 (Checking or interlocking device), IEC 61850-7.4 LN:CILO (Interlocking)

- **Overload protection**: The name "overload protection" does not appear in the reference documents ([60] and [71]) so we interpret it as an overcurrent protection, which is one the most common electrical protection functions. **Overcurrent-time-protection**: An overcurrent protection trips the associated circuit breaker when current becomes higher than a definite threshold, either instantaneously (Instantaneous overcurrent) or after a delay. This delay can be set to a fixed value (definite time) or inversely proportional to the current value, let's say to the severity of the overcurrent.

  ANSI/IEEE code: 50 (Definite time) + 51 (Inverse time), IEC 61850-7.4 LN: PTOC (Time overcurrent)

- **Protection signaling**: We did not find any reference to "protection signaling" function in the reference documents ([60] and [71]). We assume it is about alarm handling and reporting.

  Mentioned as ALARM in [60], IEC 61850-7.4 LNs: CALH (Alarm handling) and some or all interface LNS, e.g. IARC (Archiving), IHMI (Human machine interface), ITCI (Telecontrol interface), ITMI (Telemonitoring interface)

- **Synchronism-check relay**: Relays that check power differences (voltage magnitude, phase angle, frequency) between two circuits of the grid are within acceptable limits before allowing closure of the circuit breaker between them.

  ANSI/IEEE code: 25 (Synchronizing or synchronism-check relay), IEC 61850-7.4 LN: RSYN (Synchronism-check)

- **Transformer differential protection**: It protects against short-circuits between turns of a winding and between windings that correspond to phase-to-phase or three-phase type short-circuits.

  ANSI/IEEE code: 87T (Differential transformer), IEC 61850-7.4 LNs: PDIF (Differential) + PHAR (Harmonic restraint)

- **Voltage regulator**: This device purpose is to output a voltage as stable as possible, that is at a certain value or between certain (generally close) limits.

ANSI/IEEE code: 90V (Regulating device for voltage), no corresponding LN was found in the references documents [60] or [71]

Appendix E

Test cases for GOOSE protocol testing

These appendix gives detail about the different cases evaluated in the malformed GOOSE frame test presented in section 4.1.4.2.

Table E.1: Testing of off-the-shelf Siemens SIPROTEC 5 IEDs handling of malformed GOOSE frames

Field name	**Normal values**	**Test cases**					**Remarks**	**Results**
		1	**2**	**3**	**4**	**5**		
Destination address	Following Part 8.1 (annex B) requirements, shall be comprised between 01-0C-CD-01-00-00 and 01-0C-CD-01-01-FF: **01-0C-CD-01-00-01**	Not the one allocated to the considered GOOSE application but one of another GOOSE application for the same publisher IED: 01:0c:cd:01:00:04	Not one allocated to any GOOSE application of the considered project but still in the recommended range: **01:0c:cd:01:00:06**	Out of the recommended range and not included in address range for GSSE or Sampled Values: **01-0C-CD-01-02-00**	Address from recommended range allocated to GSSE: **01-0C-CD-02-00-00**	Address from recommended range allocated to Multicast Sampled Values: **01-0C-CD-04-00-00**		RxCounter of 7SJ82 Subscriber 2 is incremented only for valid destination address. For the 5 tests, test frames were discarded by the IED.
Source address	As given in GOOSE application configuration **00:09:8e:fa:b7:1a**	Another of the source addresses configured in the project but not the one of the considered GOOSE application –> Source address value mismatches its associated APPID, GoCBRef, DatSet and GoID **00:09:8e:fa:b7:1c**	A source address not involved in the project: **00:09:8e:fa:b7:1b**, **00:09:8e:fd:b7:1b**, **00:79:8e:fa:b7:1b**					RxCounter is incremented by 5 for test case 1 but by 6 for test case 2, which would mean that if source address corresponds to another IED of the DIGSI project then the GOOSE frame is discarded but if source address is a random one then it is read. Confirmation with a series of frames making first Boolean blink –> corresponding LED does blink. That means that the GOOSE frame is read and used.
VLAN tag	**TPID**: As specified in Part 8.1 (annex C), TPID indicates the Ethertype assigned for 802.1Q Ethernet encoded frames: 0x8100. **VLAN tag does not exist in considered frames**	0x8100	Another value: 0x800 (IPV4)					RxCounter of 7SJ82 Subscriber2 is incremented only for valid frame. For the 5 tests, test frames were discarded by the IED.

Table E.1: Testing of off-the-shelf Siemens SIPROTEC 5 IEDs handling of malformed GOOSE frames

Field name	**Normal values**	**Test cases**					**Remarks**	**Results**
		1	**2**	**3**	**4**	**5**		
	TCI: As specified in Part 8.1 (annex C), TCI = User priority (byte 1 / bits 8-7-6) + CFI (bit 5, shall be 0) + VID (byte 1 / bits 4 to 0 and byte 2). As configured in GOOSE application. Default values if not configured: Priority = 4 and VID = 0 (0x8000). **VLAN tag does not exist in considered frames**	Default values: **user priority = 4, CFI = 0, VID = 0**		User priority different from the default one but still in range (0-low to 4-high). **user priority = 2, CFI = 0, VID = 0**	User priority > 4: **user priority = 5, CFI = 0, VID = 0**	VID different than the default one: **user priority = 4, CFI = 0, VID = 2**		
Ethertype	As specified in Part 8.1 (annex C): **0x88b8**	**0x88b9** (GSE management)	**0x88ba** (Sampled Values)	Another value (e.g. **0x88bb**)				RxCounter of 7SJ82 Subscriber2 is incremented only for valid frame. For the 3 tests, test frames were discarded by the IED.
APPID	As specified in Part 8.1 (annex C), the value of APPID is the combination of the APPID Type, defined as the two most significant bits of the value (00 for GOOSE), and the actual ID. Reserved range = 0x0000 to 0x3FFF. **0x0002**	APPID of another GOOSE application for same publisher IED–> APPID value mismatches its associated Destination address, GoCBRef, DatSet and GoID **0x0005**	APPID of another GOOSE application for another publisher IED–> APPID value mismatches its associated Destination address, Source address, GoCBRef, DatSet and GoID **0x0001**	APPID Type 00 + ID not available in project configuration but still in authorized range –> APPID value mismatches its associated Destination address, Source address, GoCBRef, DatSet and GoID **0x0003**	APPID Type 01 (Sampled Values) + actual ID **0x4002**			RxCounter of 7SJ82 Subscriber2 is incremented only for valid frame. For the 4 tests, test frames were discarded by the IED.
Length	As specified in Part 8.1 (annex C), 8 + m, where m is the length of the APDU and m is less than 1492. Frames with inconsistent or invalid length field shall be discarded. 147	True length - 1: **146**	True length + 1: **148**	**0**	> authorized value: **1493**		Check authorized length both in most recent Ed.1 and Ed.2	RxCounter of 7SJ82 Subscriber2 is incremented only for valid frame. For the 4 tests, test frames were discarded by the IED.

Table E.1: Testing of off-the-shelf Siemens SIPROTEC 5 IEDs handling of malformed GOOSE frames

Field name	**Normal values**	**Test cases**					**Remarks**	**Results**
		1	**2**	**3**	**4**	**5**		
Reserved 1	As specified in Part 8.1 (annex C), the Reserved1 and Reserved2 are reserved for future standardized applications and shall be set to 0 as default. **0x0000** and **0x0000**	Non null value: **0x0101**		Non null value: **0x0101**	**0xFFFF**	Null value		For the 4 first test cases, frames were accepted. Further testing with blinking Boolean ok. We can conclude that IED does not use (read?) these 2 fields. If one or both the Reserved fields are missing, RxCounter is not incremented.
Reserved 2			Non null value: **0x0101**	Non null value: **0x0101**	**0xFFFF**	empty (2 bytes were deleted)		
GoCBRef	As configured in GOOSE application **SIP2CB1/LLN0$ GO$ Control_ Dataset**	Another from project configuration –> GoCBRef value mismatches its associated APPID, DatSet and GoID **SIP2CB1/LLN0$ GO$ Control_ Dataset_ 1**	An invalid GoCBRef value, not available in project configuration. **SIP2CB1/LLN0$ GO$ Control_ Dataset_ 2**					RxCounter is not incremented for any of the test frames of the 2 considered test cases.
TimeAllowed-ToLive	As configured in GOOSE application: **1500**	Another value among available transfer profiles from DIGSI 5: **3000**	Less than the configured value and less than the max time between 2 GOOSE messages of the same application: <1000ms: **800**	Less than the configured value and less than half the max time between 2 GOOSE messages of the same application: <500ms. SIPROTEC 5 devices are designed to allow at most one lost GOOSE message before considering the connection as lost. For this test, a series of GOOSE frames must be sent with time interval = max time, here 1000ms. SqNum is incremented appropriately. **300**	Negative value: **-1000**		May require further testing	For the 3 first test cases, RxCounter is incremented for the test frame. Frame discarded for test case 4, which means that IED interpret field TATL as a signed integer.

Table E.1: Testing of off-the-shelf Siemens SIPROTEC 5 IEDs handling of malformed GOOSE frames

Field name	**Normal values**	**Test cases**					**Remarks**	**Results**
		1	**2**	**3**	**4**	**5**		
DatSet	As configured in GOOSE application **SIP2CB1/LLN0$ Dataset**	DatSet of another GOOSE application for same publisher IED. –> GoCBRef value mismatches its associated Destination address, APPID, GoCBRef and GoID: **SIP2CB1/LLN0$ Dataset_1**	DatSet of a GOOSE application for another publisher IED. –> GoCBRef value mismatches its associated Destination address, Source address, APPID, GoCBRef and GoID: **SIP1CB1/LLN0$ Dataset**	An invalid DatSet value, not available in project configuration: **SIP2CB2/LLN0$ Dataset_1**				Test frame is accepted for the 3 test cases
GoID	As configured in GOOSE application: **SIP2/CB1/LLN0 /Control_Dataset**	GoID from another GOOSE application, same publisher IED –> GoCBRef value mismatches its associated Destination address, APPID, DatSet and GoCBRef: **IED2/CB1/LLN0 /Control_ Dataset_1**	GoID from GOOSE application for the other publisher IED –> GoCBRef value mismatches its associated Destination address, Source address, APPID, DatSet and GoCBRef: **SIP1/CB1/LLN0 /Control_Dataset**	An invalid GoID value, not available in project configuration: **SIP3/CB1/LLN0 /Control_Dataset**	GoID is an optional field (IEC 61850 Part 8-1): **No GoID field**			Test frame is accepted for the 4 test cases
T	UTC time as specified in Part 8.1 (8.1.3.6 and Annex G)	Future date compared to reception time						
StNum	As specified in Part 8.1: This INTEGER value shall have a range of 1 to 4 294 967 295.	Series of 5 frames with StNum = $2^{32} - 1 = 4294967295$ and then 6^{th} frame has a **0** StNum	Series of 5 frames with StNum = $2^{32} - 1 = 4294967295$ and then 6^{th} frame has a StNum > 1: **9**					
SqNum	As specified in Part 8.1: This INTEGER value shall have a range of 0 to $2^{32} - 1 = 4294967295$. The value of 0 is reserved for the first transmission of a StNum change. SqNum will increment for each transmission, but will rollover to a value of 1.	Series of 5 frames with SqNum = 4 294 967 291 to 4 294 967 295 and then 6th frame has SqNum = **0** It may be necessary to send it twice	Series of 5 frames with SqNum = 4 294 967 295 and then 6th frame has SqNum >1: **9** It may be necessary to send it twice					

Table E.1: Testing of off-the-shelf Siemens SIPROTEC 5 IEDs handling of malformed GOOSE frames

Field name	**Normal values**	**Test cases**					**Remarks**	**Results**
		1	**2**	**3**	**4**	**5**		
Test	Boolean, default **FALSE**	**TRUE**	Test has a default FALSE value as defined in Annex A of IEC 61850 Part 8-1, which means it is optional. **No Test field**				Frame is unsurprisingly accepted. The difference should occur in frame content processing. How is not clear, though (more info in Siprotec 5 documentation?).	Test frames are accepted (RxCounter incremented)
ConfRev	**1**	Another value **2**						Test frame is accepted (RxCounter incremented)
NdsCom	Boolean, default **FALSE**	**TRUE**	NdsCom has a default FALSE value as defined in Annex A of IEC 61850 Part 8-1, which means it is optional. **No NdsCom field**				Can we configure ndsCom value criticity for a data received by GOOSE? Maybe it is not important for non critical variables but for tripping signals or other critical variables?	Test frames are accepted (RxCounter incremented). Checked with blinking Boolean frames. That means that a wrong value of ndsCom does not prevent IED from using the frame content.
NumDatSet Entries	As configured in GOOSE application: **4**	Less than its true value: **3**	More than its true value: **6**					RxCounter is not incremented for any of the test frames of the 2 considered test cases.
Security	Not included in GOOSE messages published by available SIPROTEC 5 devices	Adding this field at the end of the frame with appropriate value of Length field. Value **00**						RxCounter is not incremented for the test frame.
Field order shift	Fields order is as given in the standard	Shift TATL and DatSet	Shift StNum and SqNum					Frames are not accepted

Appendix F

Boundary-value analysis of the GOOSE parser

As part of our pull request to Bro for integrating the GOOSE parser, we provided testing material to check whether it works properly and while analyzing GOOSE frames with values at the boundary or beyond the protocol specifications. That is what is called "boundary-value analysis". Such an analysis is chosen when an exhaustive testing is not possible. It consists of feeding a function with input arguments taking in turn a value at the limit of the function specifications. While boundaries of a parameter are tested, the other ones take their typical values, far from their own boundaries. To test our GOOSE parser, we feed a Bro instance running it with the GOOSE traces corresponding to all the test cases defined in what follows.

F.1 Data types under test

We checked the data types bit-string, integer, unsigned integer and array.

F.1.1 Bit-string

This type allows sending a series of bits. As the smaller size unit is the byte, padding bits are used to fill a byte. Thus encoding of a bit-string following ASN.1 rules results in an identifier tag byte, followed by a byte for the size (in bytes) of the bit-string including a byte giving the number of padding bits and the bit-string value itself completed with the padding bits. The number of padding bits is necessarily less than 8. Typical value chosen for the test cases is 3. The test cases cover 0, 1 and 2-byte long bit-strings. Bit-strings may theoretically be of infinite size, though.

F.1.2 Integer

An integer field carried by a GOOSE frame is encoded following the ASN.1 rules. In terms of bytes, it gives a tag byte identifying the data field, a byte for the size (in bytes) of the integer and the integer value itself. If the very first bit received is a 1, the integer is

interpreted as a negative number. The parser we implemented only decodes integer coded on eight bytes at the most. Greater integers are decoded as 0.

The tests make the integer size and value vary, keeping consistency between them. The typical size is 4 bytes and the typical value is 42. The values to be tested are -1, 0, 1 and the boundary values for 4 bytes, i.e. $2^{31} - 1$ and -2^{31}. The test cases are therefore as given in Table F.1.

Table F.1: Test cases for boundary-value analysis of integer data type

Test case	Size (bytes)	Value	Expected output
1	0	-	0
2	1	42	42
3	8	42	42
4	9	42	0
5	4	-1	-1
6	4	0	0
7	4	1	1
8	4	-2147483648	-2147483648
9	4	2147483647	2147483647

F.1.3 Unsigned integer

GOOSE protocol extends ASN.1 for encoding of unsigned integers. The boundary values for 4 bytes are 0 and $2^{32} - 1$.The test cases are given by Table F.2.

Table F.2: Test cases for boundary-value analysis of unsigned integer data type

Test case	Size (bytes)	Value	Expected output
1	0	-	0
2	1	42	42
3	8	42	42
4	9	42	0
5	4	0	-1
6	4	1	0
7	4	4294967295	4294967295

F.1.4 Array

The tests for "array" data type have two objectives: verify whether an item of an array can be of any type, including "array", and check the parser robustness to malformed arrays. A malformed array is encoded on a number of bytes different than declared by the length bytes.

F.2 Test cases integration into Btest

The Bro project includes a framework for unit testing, BTest, that developers can use and complete with their own test cases. The testing material for the GOOSE parser has been integrated as a script to display the GOOSE parser output, pcap files with the traces corresponding to all the test cases and the expected output logs of a Bro instance running the GOOSE parser. It is available at the GOOSE parser pull request web page[1].

[1] https://github.com/bro/bro/pull/76

Bibliography

[1] ABB. 615 Series Cyber Security Deployment Guideline, October 2015.

[2] ABB. 615 Series IEC 61850 Engineering Guide, October 2015.

[3] Abdulmohsen Almalawi, Xinghuo Yu, Zahir Tari, Adil Fahad, and Ibrahim Khalil. An unsupervised anomaly-based detection approach for integrity attacks on SCADA systems. *Computers & Security*, 46:94–110, 2014.

[4] Alstom. Network Protection and Automation Guide, 2011.

[5] American National Standards Institute (ANSI) / International Society of Automation (ISA). ISA 62443-2-1: Security for Industrial Automation and Control Systems - Establishing an Industrial Automation and Control Systems Security Program, January 2009.

[6] G. Andersson, P. Donalek, R. Farmer, N. Hatziargyriou, I. Kamwa, P. Kundur, N. Martins, J. Paserba, P. Pourbeik, J. Sanchez-Gasca, R. Schulz, A. Stankovic, C. Taylor, and V. Vittal. Causes of the 2003 major grid blackouts in North America and Europe, and recommended means to improve system dynamic performance. *IEEE Transactions on Power Systems*, 20(4):1922–1928, Nov 2005.

[7] ANSSI – Working Group on Industrial Control System cybersecurity. Cybersecurity for Industrial Control Systems - Managing Cybersecurity for Industrial Control Systems. Technical report, The French Network and Security Agency (ANSSI - Agence Nationale de la Sécurité des Systèmes d'Information), June 2012. https://www.ssi.gouv.fr/uploads/2014/01/Managing_Cybe_for_ICS_EN.pdf.

[8] ANSSI – Working Group on Industrial Control System cybersecurity. Cybersecurity for Industrial Control Systems - Classification Method and Key Measures. Technical report, The French Network and Security Agency (ANSSI - Agence Nationale de la Sécurité des Systèmes d'Information), January 2014. http://www.ssi.gouv.fr/uploads/2014/01/industrial_security_WG_Classification_Method.pdf.

[9] ANSSI (Agence Nationale de la Sécurité des Systèmes d'Information). EBIOS: Expression des Besoins et Identification des Objectifs de Sécurité - bases de données, January 2010. urlhttp://www.ssi.gouv.fr/uploads/2011/10/EBIOS-1-GuideMethodologique-2010-01-25.pdf.

[10] ANSSI (Agence Nationale de la Sécurité des Systèmes d'Information). EBIOS: Expression des Besoins et Identification des Objectifs de Sécurité - méthode de gestion des risques, January 2010. http://www.ssi.gouv.fr/uploads/2011/10/EBIOS-1-GuideMethodologique-2010-01-25.pdf.

[11] A. Ashok, S. Krishnaswamy, and M. Govindarasu. PowerCyber: A remotely accessible testbed for Cyber Physical security of the Smart Grid. In *2016 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, September 2016.

[12] A. Avizienis, J. C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 1(1):11–33, January 2004.

[13] Rafael Ramos Regis Barbosa, Ramin Sadre, and Aiko Pras. A first look into scada network traffic. In *2012 IEEE Network Operations and Management Symposium*, pages 518–521, April 2012.

[14] Rafael Ramos Regis Barbosa, Ramin Sadre, and Aiko Pras. Towards periodicity based anomaly detection in scada networks. In *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies Factory Automation (ETFA 2012)*, September 2012.

[15] Rafael Ramos Regis Barbosa, Ramin Sadre, and Aiko Pras. Exploiting traffic periodicity in industrial control networks. *International Journal of Critical Infrastructure Protection*, 13:52–62, 2016.

[16] Robin Berthier and William H. Sanders. Specification-Based Intrusion Detection for Advanced Metering Infrastructures. In *Proceedings of the 2011 IEEE 17th Pacific Rim International Symposium on Dependable Computing*, PRDC '11, pages 184–193. IEEE Computer Society, 2011.

[17] R. Caralli, J. Stevens, L. Young, and W. Wilson. Introducing OCTAVE Allegro: Improving the Information Security Risk Assessment Process. Technical Report CMU/SEI-2007-TR-012, Software Engineering Institute, Carnegie Mellon University, May 2007.

[18] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. Nai Fovino, and A. Trombetta. A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems. *IEEE Transactions on Industrial Informatics*, 7(2):179–186, May 2011.

[19] Marco Caselli, Emmanuele Zambon, and Frank Kargl. Sequence-aware Intrusion Detection in Industrial Control Systems. In *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security*, CPSS '15, pages 13–24, New York, NY, USA, 2015. ACM.

[20] CEN-CENELEC. European Standardization - CEN-CENELEC sectors Energy management & energy efficiency - Smart grids.

[21] CEN-CENELEC-ETSI Smart Grid Coordination Group. Smart Grid Information Security. Technical report, The European Committee for Standardization (CEN), the European Committee for Electrotechnical Standardization (CENELEC) and the European Telecommunications Standards Institute (ETSI), November 2012.

[22] CEN-CENELEC-ETSI Smart Grid Coordination Group. Smart Grid Reference Architecture. Technical report, The European Committee for Standardization (CEN), the European Committee for Electrotechnical Standardization (CENELEC) and the European Telecommunications Standards Institute (ETSI), November 2012.

[23] CEN-CENELEC-ETSI Smart Grid Coordination Group. Sustainable Processes. Technical report, The European Committee for Standardization (CEN), the European Committee for Electrotechnical Standardization (CENELEC) and the European Telecommunications Standards Institute (ETSI), November 2012.

[24] CEN-CENELEC-ETSI Smart Grid Coordination Group. Smart Grid Information Security. Technical report, The European Committee for Standardization (CEN), the European Committee for Electrotechnical Standardization (CENELEC) and the European Telecommunications Standards Institute (ETSI), December 2014.

[25] CESG, the UK National Technical Authority for Information Assurance. HMG IA Standard No. 1 Technical Risk Assessment. Technical report, October 2009.

[26] Y. Cherdantseva, P. Burnap, A. Blyth, P. Eden, K. Jones, H. Soulsby, and K. Stoddart. A review of cyber security risk assessment methods for SCADA systems. *Elsevier Computers & Security*, 56, February 2016.

[27] Steven Cheung, Bruno Dutertre, Martin Fong, Ulf Lindqvist, Keith Skinner, and Alfonso Valdes. Using Model-based Intrusion Detection for SCADA Networks. In *Proceedings of the SCADA Security Scientific Symposium*, January 2007.

[28] Pierre Chifflier and Arnaud Fontaine. Architecture système sécurisée de sonde IDS réseau. In *Proceedings of C&ESAR 2014, Computer & Electronics Security Applications Rendez-vous*, November 2014.

[29] CIGRÉ - Conseil International des Grands Réseaux Électriques. Communication requirements of data flow within substation, Cigré REF.180 SC 34 WG 34.03, 2001.

[30] D. D'Elia. The Economics of Cybersecurity: From the Public Good to the Revenge of the Industry. In *Security of Industrial Control Systems and Cyber PhysicalSystems*. Springer, 2016.

[31] D. E. Denning. An Intrusion-Detection Model. In *Security and Privacy, 1986 IEEE Symposium on*, April 1986.

[32] David Diallo and Mathieu Feuillet. Détection d'intrusion dans les systèmes industriels : Suricata et le cas Modbus. In *Proceedings of C&ESAR 2014, Computer & Electronics Security Applications Rendez-vous*, November 2014.

[33] Patrick Düssel, Christian Gehl, Pavel Laskov, Jens-Uwe Bußer, Christof Störmann, and Jan Kästner. Cyber-critical infrastructure protection using real-time payload-based anomaly detection. In *International Workshop on Critical Information Infrastructures Security*, pages 85–97, 2009.

[34] Merlin Gerin / Schneider Electric. Protection des réseaux électriques – Guide de la protection, 2003.

[35] European Union Agency for Network and Information Security (ENISA). Protecting Industrial Control Systems - Annex V: Key Findings. Technical report, December 2011.

[36] European Union Agency for Network and Information Security (ENISA). Smart Grid Security - Annex II: Security aspects of the smart grid. Technical report, April 2012.

[37] European Union Agency for Network and Information Security (ENISA). Proposed security measures for smart grids. Technical report, December 2013.

[38] M. Fabro and V. Maio. Using Operational Security (OPSEC) to Support a Cyber Security Culture in Control Systems Environments, version 1.0 draft. Technical report, INL Critical Infrastructure Protection Center, February 2007. https://ics-cert.us-cert.gov/sites/default/files/recommended_practices/RP_Using%20OpSec_v1_Draft.pdf, Last consulted 08/01/2016.

[39] Tom Fawcett. ROC graphs: Notes and practical considerations for researchers. *Machine learning*, 31(1):1–38, 2004.

[40] Y. Fourastier and J.-C. Jabot. Analyse des risques. In Y. Fourastier and L. Pietre-Cambacedes, editors, *Cybersécurité des installations industrielles, Défendre ses systèmes numériques*, chapter VI, pages 211–234. Cépaduès, December 2015.

[41] Yannick Fourastier. Typologie des installations industrielles. In Yannick Fourastier and Ludovic Pietre-Cambacedes, editors, *Cybersécurité des installations industrielles, Défendre ses systèmes numériques*, chapter II, pages 39–70. Cépaduès, December 2015.

[42] I. N. Fovino, A. Carcano, T. D. L. Murel, A. Trombetta, and M. Masera. Modbus/DNP3 State-Based Intrusion Detection System. In *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, pages 729–736, April 2010.

[43] I. Nai Fovino, A. Coletta, A. Carcano, and M. Masera. Critical State-Based Filtering System for Securing SCADA Network Protocols. *IEEE Transactions on Industrial Electronics*, 59(10):3943–3950, October 2012.

[44] Shailendra Fuloria, Ross Anderson, Kevin McGrath, Kai Hansen, and Fernando Alvarez. The protection of substation communications. In *Proc. of SCADA Security Scientific Symposium*, 2010.

[45] H. Hadeli, R. Schierholz, M. Braendle, and C. Tuduce. Generating configuration for missing traffic detector and security measures in industrial control systems based on the system description files. In *2009 IEEE Conference on Technologies for Homeland Security*, pages 503–510, May 2009.

[46] H. Hadeli, R. Schierholz, M. Braendle, and C. Tuduce. Leveraging determinism in industrial control systems for advanced anomaly detection and reliable security configuration. In *2009 IEEE Conference on Emerging Technologies Factory Automation*, September 2009.

[47] A. Hahn, A. Ashok, S. Sridhar, and M. Govindarasu. Cyber-Physical Security Testbeds: Architecture, Application, and Evaluation for Smart Grid. *IEEE Transactions on Smart Grid*, 4(2):847–855, June 2013.

[48] T. Hecht, L. Langer, and P. Smith. Cybersecurity Risk Assessment in Smart Grids. In *5th Symposium on Communications for Energy Systems (ComForEn 2014)*, 2014.

[49] R. C. Borges Hink, J. M. Beaver, M. A. Buckner, T. Morris, U. Adhikari, and S. Pan. Machine learning for power system disturbance and cyber-attack discrimination. In *Resilient Control Systems (ISRCS), 2014 7th International Symposium on*, August 2014.

[50] Frank Hohlbaum, Markus Braendle, and Fernando Alvarez. Cyber security practical considerations for implementing IEC 62351. In *Protection Automation and Control World (PACWorld 2010)*, 2010.

[51] Junho Hong, Chen-Ching Liu, and Manimaran Govindarasu. Detection of cyber intrusions using network-based multicast messages for substation automation. In *Innovative Smart Grid Technologies Conference (ISGT), 2014 IEEE PES*, February 2014.

[52] Junho Hong, Chen-Ching Liu, and Manimaran Govindarasu. Integrated anomaly detection for cyber security of the substations. In *2014 IEEE PES General Meeting | Conference Exposition*, July 2014.

[53] S. Hong, M. Lee, and D. Y. Shin. Experiments for Embedded Protection Device for Secure SCADA Communication. In *2010 Asia-Pacific Power and Energy Engineering Conference*, March 2010.

[54] S. Hong, D. Y. Shin, and M. Lee. Evaluating Security Algorithms in the Substation Communication Architecture. In *2009 International Conference on Scalable Computing and Communications; Eighth International Conference on Embedded Computing*, pages 314–318, September 2009.

[55] J. Hoyos, M. Dehus, and T. X. Brown. Exploiting the GOOSE protocol: A practical attack on cyber-infrastructure. In *2012 IEEE Globecom Workshops*, pages 1508–1513, December 2012.

[56] Martin Hutle, Gerhard Hansch, William Fitzgerald, Thomas Hecht, Ewa Piatkowska, and Paul Smith. Smart Grid Protection Against Cyber Attacks (SPARKS): D2.2 Threat and Risk Assessment Methodology. Technical report, September 2015. https://project-sparks.eu/wp-content/uploads/2014/04/D2_2_Threat_and_Risk_Assessment_Methodology.pdf, Last consulted on 07/22/2016.

[57] Idaho National Laboratory (INL). Common Cyber Security Vulnerabilities Observed in Control System Assessments by the INL NSTB Program, November 2008.

[58] Idaho National Laboratory (INL). Electric grid reliability, 2014. https://factsheets.inl.gov/FactSheets/electric-grid-reliability.pdf.

[59] IEEE Power and Energy Society. 1686-2004: IEEE Standard Communication Delivery Time Performance Require- ments for Electric Power Substation Automation, February 2005.

[60] IEEE Power and Energy Society. C37.2: IEEE Standard Electrical Power System Device Function Numbers, Acronyms, and Contact Designations, 2008.

[61] IEEE Power and Energy Society. 1686: IEEE Standard for Intelligent Electronic Devices Cyber Security Capabilities, 2014.

[62] IEEE Power and Energy Society. C37.240: IEEE Standard Cybersecurity Requirements for Substation Automation, Protection, and Control Systems, 2014.

[63] Idaho National Laboratory INL. INL Cyber Security Research, 2014. http://www4vip.inl.gov/research/inl-cyber-security-research/d/inl-cyber-security-research.pdf.

[64] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems in substations - Part 3: General requirements, 2002.

[65] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 2: Glossary, 2003.

[66] International Electrotechnical Commission (IEC). IEC 62351: Power systems management and associated information exchange - Data and communications security - Part 1: Communication network and system security - Introduction to security issues, 2007.

[67] International Electrotechnical Commission (IEC). IEC 62351: Power systems management and associated information exchange - Data and communications security - Part 6: Security for IEC 61850, 2007.

[68] International Electrotechnical Commission (IEC). IEC 62443: Industrial communication networks - Network and system security - Part 1-1: Terminology, concepts and models, 2009.

[69] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 7.2: Basic information and communication structure - Abstract communication service interface (ACSI), 2010.

[70] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 7.3: Basic communication structure - Common Data Classes, 2010.

[71] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 7.4: Basic communication structure - Compatible logical node classes and data object classes, 2010.

[72] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 8.1: Specific Communication Service Mapping (SCSM) - Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3, 2011.

[73] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 1: Introductory and overview, 2013.

[74] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 5: Communication requirements for functions and device models, 2013.

[75] International Electrotechnical Commission (IEC). IEC 61850: Communication networks and systems for power utility automation - Part 7.1: Basic communication structure - Principles and models, 2013.

[76] International Electrotechnical Commission (IEC). IEC 61812: Analysis techniques for system reliability – Procedure for failure mode and effects analysis (FMEA), 2016.

[77] International Organization for Standardization (ISO). ISO 13335-2: Guidelines for the management of IT Security - Part 2: Managing and planning IT Security, 1997.

[78] International Organization for Standardization (ISO). ISO 9506: Industrial automation systems - Manufacturing Message Specification, 2003.

[79] International Organization for Standardization (ISO). ISO 31000: Risk management - Principles and guidelines, 2009.

[80] International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC). ISO/IEC 27005: Information technology - Security techniques - Information Security Management Systems - Information Security Risk Management, June 2011.

[81] International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC). ISO/IEC 27001: Information technology - Security techniques - Information Security Management Systems - Requirements, October 2013.

[82] International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC). ISO/IEC 27002: Information technology - Security techniques - Information Security Management Systems - Code of Practice for Information Security Controls, October 2013.

[83] International Society of Automation (ISA) / International Electrotechnical Commission (IEC). ISA/IEC 62443-1-1: Security for Industrial Automation and Control Systems - Models and Concepts - Draft 5, Edit 5, October 2015.

[84] International Society of Automation (ISA) / International Electrotechnical Commission (IEC). ISA/IEC 62443-1-3: Security for Industrial Automation and Control Systems - System Security Compliance Metrics - Draft 1, Edit 19, October 2015.

[85] International Society of Automation (ISA) / International Electrotechnical Commission (IEC). ISA/IEC 62443-2-1: Security for Industrial Automation and Control Systems - Industrial Automation and Control System Security Management System - Draft 7, Edit 5, November 2015.

[86] International Society of Automation (ISA) / International Electrotechnical Commission (IEC). ISA/IEC 62443-3-2: Security for Industrial Automation and Control Systems - System Risk Assessment for System Design - Draft 6, Edit 3, August 2015.

[87] Maëlle Kabir-Querrec, Stéphane Mocanu, Pascal Bellemain, Jean-Marc Thiriet, and Eric Savary. Architecture des systèmes d'automatisation des postes résiliente aux attaques des trames GOOSE. In *Journées C&ESAR 2015*, November 2015.

[88] Maëlle Kabir-Querrec, Stéphane Mocanu, Pascal Bellemain, Jean-Marc Thiriet, and Eric Savary. Corrupted GOOSE Detectors: Anomaly Detection in Power Utility Real-Time Ethernet Communications. In *GreHack 2015*, Grenoble, France, November 2015. Verimag.

[89] Maëlle Kabir-Querrec, Stéphane Mocanu, Jean-Marc Thiriet, and Eric Savary. Power Utility Automation Cybersecurity: IEC 61850 Specification of an Intrusion Detection Function. In *25th European Safety and Reliability Conference (ESREL 2015)*, Zürich, Switzerland, September 2015.

[90] Maëlle Kabir-Querrec, Stéphane Mocanu, Jean-Marc Thiriet, and Eric Savary. A Test bed dedicated to the Study of Vulnerabilities in IEC 61850 Power Utility Automation Networks. In *21st IEEE Emerging Technologies and Factory Automation*, Berlin, Germany, September 2016.

[91] B. Kang, P. Maynard, K. McLaughlin, S. Sezer, F. Andrén, C. Seitl, F. Kupzog, and T. Strasser. Investigating cyber-physical attacks against IEC 61850 photovoltaic inverter installations. In *2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA)*, September 2015.

[92] William Knowles, Daniel Prince, David Hutchison, Jules Ferdinand Pagna Disso, and Kevin Jones. A survey of cyber security management in industrial control systems. *International Journal of Critical Infrastructure Protection*, 9:52 – 80, 2015.

[93] P. Kobes. Zomm sur la norme internationale IEC 62443 pour la cybersécurité des systèmes industriels. In Y. Fourastier and L. Pietre-Cambacedes, editors, *Cybersécurité des installations industrielles, Défendre ses systèmes numériques*, chapter XV, pages 447–481. Cépaduès, December 2015.

[94] Oualid Koucham. Détection d'intrusion pour les systèmes de contrôle industriels - rapport d'état d'avancement des travaux. March 2016.

[95] G. Koutsandria, V. Muthukumar, M. Parvania, S. Peisert, C. McParland, and A. Scaglione. A hybrid network IDS for protective digital relays in the power transmission grid. In *Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on*, pages 908–913, November 2014.

[96] Nishchal Kush, Ejaz Ahmed, Mark Branagan, and Ernest Foo. Poisoned GOOSE: exploiting the GOOSE protocol. In *Proceedings of the Twelfth Australasian Information Security Conference-Volume 149*, pages 17–22. Australian Computer Society, Inc., 2014.

[97] Y. Kwon, H. K. Kim, Y. H. Lim, and J. I. Lim. A behavior-based intrusion detection technique for smart grid infrastructure. In *PowerTech, 2015 IEEE Eindhoven*, June 2015.

[98] L. Langer, P. Smith, and M. Hutle. Smart grid cybersecurity risk assessment. In *Smart Electric Distribution Systems and Technologies (EDST), 2015 International Symposium on*, pages 475–482, September 2015.

[99] Lucie Langer, Florian Skopik, Paul Smith, and Markus Kammerstetter. From old to new: Assessing cybersecurity risks for an evolving smart grid. *Elsevier Computers & Security*, 62, 2016.

[100] Robert M. Lee, Michael J. Assante, and Tim Conway. Analysis of the Cyber Attack on the Ukrainian Power Grid - Defense Use Case. Technical report, SANS ICS (SysAdmin, Audit, Network, Security Institute - Industrial Control Systems) and E-SIC (Electricity - Inforamtion Sharing and Analysis Center), Mach 2016.

[101] M. Levorato and U. Mitra. Fast anomaly detection in SmartGrids via sparse approximation theory. In *Sensor Array and Multichannel Signal Processing Workshop (SAM), 2012 IEEE 7th*, June 2012.

[102] Hui Lin, Adam Slagell, Catello Di Martino, Zbigniew Kalbarczyk, and Ravishankar K. Iyer. Adapting Bro into SCADA: Building a Specification-based Intrusion Detection System for the DNP3 Protocol. In *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop*, CSIIRW '13. ACM, 2013.

[103] O. Linda, T. Vollmer, and M. Manic. Neural Network based Intrusion Detection System for critical infrastructures. In *2009 International Joint Conference on Neural Networks*, pages 1827–1834, June 2009.

[104] Litos Strategic Communication. The Smart Grid: an introduction. Technical report, U.S. Department of Energy, November 2008.

[105] K. Manandhar, X. Cao, F. Hu, and Y. Liu. Detection of Faults and Attacks Including False Data Injection Attack in Smart Grid Using Kalman Filter. *IEEE Transactions on Control of Network Systems*, 1(4):370–379, December 2014.

[106] J. Mcdonald, N. Oualha, A. Puccetti, A. Hecker, and F. Planchon. Application of EBIOS for the risk assessment of ICT use in electrical distribution sub-stations. In *PowerTech (POWERTECH), 2013 IEEE Grenoble*, June 2013.

[107] T. Morris, R. Vaughn, and Y. Dandass. A Retrofit Network Intrusion Detection System for MODBUS RTU and ASCII Industrial Control Systems. In *2012 45th Hawaii International Conference on System Sciences*, pages 2338–2345, January 2012.

[108] Thomas Morris, Rayford Vaughn, and Yoginder S Dandass. A testbed for SCADA control system cybersecurity research and pedagogy. In *Proceedings of the Seventh Annual Workshop on Cyber Security and Information Intelligence Research*. ACM, 2011.

[109] National Institute of Standards and Technology (NIST). SP 800-30, Risk Management Guide for Information Technology Systems, July 2002.

[110] National Institute of Standards and Technology (NIST). Interagency Report 7628 - Guidelines for Smart Grid Cybersecurity, September 2014.

[111] National Offshore Petroleum Safety Authority (NOPSEMA). Guidance Note N-04300-GN0165 Revision 4, Risk Assessment, December 2012. https://www.nopsema.gov.au/assets/Guidance-notes/N-04300-GN0165-Risk-Assessment.pdf, Last consulted 08/03/2016.

[112] National Offshore Petroleum Safety Authority (NOPSEMA). Guidance Note N-04300-GN0166 Revision 6, ALARP, June 2015. https://www.nopsema.gov.au/assets/Guidance-notes/N-04300-GN0166-ALARP.pdf, Last consulted 07/22/2016.

[113] North American Electric Reliability Council (NERC). Security Guidelines for the Electricity Sector (version 1.0): Vulnerability and Risk Assessment, June 2002.

[114] North American Electric Reliability Council (NERC). Standard CIP–002–3 — Cyber Security — Critical Cyber Asset Identification, December 2009.

[115] S. Pan, T. Morris, and U. Adhikari. Developing a Hybrid Intrusion Detection System Using Data Mining for Power Systems. *IEEE Transactions on Smart Grid*, 6(6):3104–3113, November 2015.

[116] M. Parvania, G. Koutsandria, V. Muthukumary, S. Peisert, C. McParland, and A. Scaglione. Hybrid Control Network Intrusion Detection Systems for Automated Power Distribution Systems. In *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, pages 774–779, June 2014.

[117] Vern Paxson. Bro: a system for detecting network intruders in real-time. *Computer networks*, 31(23):2435–2463, 1999.

[118] Frédéric Planchon (FPC Ingénierie). Analyse Pour l'Evaluation des Risques Opérationnels - APERO : Guide utilisateur, 2014.

[119] U. Premaratne, J. Samarabandu, T. Sidhu, R. Beresh, and J. C. Tan. Security Analysis and Auditing of IEC61850-Based Automated Substations. *IEEE Transactions on Power Delivery*, 25(4):2346–2355, October 2010.

[120] U. K. Premaratne, J. Samarabandu, T. S. Sidhu, R. Beresh, and J. C. Tan. An Intrusion Detection System for IEC61850 Automated Substations. *IEEE Transactions on Power Delivery*, 25(4):2376–2383, October 2010.

[121] Dorin Adrian Rusu, Béla Genge, and Christos Siaterlis. SPEAR: A systematic approach for connection pattern-based anomaly detection in SCADA systems. *Procedia Technology*, 12:168–173, 2014.

[122] C. Schmittner, T. Gruber, P. Puschner, and E. Schoitsch. Security Application of Failure Mode and Effect Analysis (FMEA). In *SAFECOMP 2014*, June 2014.

[123] Schneider Electric. Electrical network protection: Protection guide, April 2006.

[124] Wenli Shang, Lin Li, Ming Wan, and Peng Zeng. Industrial communication intrusion detection algorithm based on improved one-class SVM. In *2015 World Congress on Industrial Control Systems Security (WCICSS)*, pages 21–25, December 2015.

[125] Siemens. Siprotec 5 Application Note: Communication Architecture Under Cyber Security Aspects, SIP5-APN-009 V6.00 and higher – Manual, May 2012.

[126] Siemens. Siprotec 5 Operation V6.00 and higher – Manual, October 2014.

[127] Siemens. Siprotec 5 PIXIT, PICS, TICS IEC 61850 V6.00 – Manual, November 2014.

[128] P. Singh, S. Garg, V. Kumar, and Z. Saquib. A testbed for SCADA cyber security and intrusion detection. In *Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC), 2015 International Conference on*, August 2015.

[129] F. Skopik, I. Friedberg, and R. Fiedler. Dealing with advanced persistent threats in smart grid ICT networks. In *Innovative Smart Grid Technologies Conference (ISGT), 2014 IEEE PES*, February 2014.

[130] Siddharth Sridhar, Adam Hahn, and Manimaran Govindarasu. Cyber-Physical System Security for the Electric Power Grid. *Proceedings of the IEEE*, 100(1):210–224, January 2012.

[131] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, and A. Hahn. SP 800-82, Revison 2, Guide to Industrial Control Systems (ICS) Security: Supervisory Control and Data Acquisition (SCADA) Systems, Distributed Control Systems (DCS), and Other Control System Configurations such as Programmable Logic Controllers (PLC). Technical report, National Institute of Standards and Technology – NIST, May 2015. http://dx.doi.org/10.6028/NIST.SP.800-82r2.

[132] Chee-Wooi Ten, Manimaran Govindarasu, and Chen-Ching Liu. Cybersecurity for Critical Infrastructures: Attack and Defense Modeling. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 40(4):853–865, July 2010.

[133] Chee-Wooi Ten, Junho Hong, and Chen-Ching Liu. Anomaly Detection for Cybersecurity of the Substations. *IEEE Transactions on Smart Grid*, 2(4):865–873, December 2011.

[134] USA Department of Defense. MIL-STD-1629A, Procedures for Performing a Failure Mode, Effect and Criticality Analysis, November 1980.

[135] V. Verendel. Quantified Security is a Weak Hypothesis: A Critical Survey of Results and Assumptions. In *Proceedings of the 2009 Workshop on New Security Paradigms Workshop*, pages 37–50. ACM, 2009.

[136] A. Villemeur. *Sûreté de fonctionnement des systèmes industriels: fiabilité, facteurs humains, informatisation*. Eyrolles, 1988.

[137] C. J. Wallnerström, P. Hilber, and J. G. Travi. Implementation and Evaluation of Commonly Used Risk Analysis Methods Applied to a Regional Power Distribution System. In *Electricity Distribution (CIRED 2013), 22nd International Conference and Exhibition on*, June 2013.

[138] Wenye Wang and Zhuo Lu. Survey Cyber Security in the Smart Grid: Survey and Challenges. *Computer Networks*, 57(5):1344–1371, April 2013.

[139] D. Wei, Y. Lu, M. Jafari, P. M. Skare, and K. Rohde. Protecting Smart Grid Automation Systems Against Cyberattacks. *IEEE Transactions on Smart Grid*, 2(4):782–795, December 2011.

[140] Joshua S White, Thomas Fitzsimmons, and Jeanna N Matthews. Quantitative analysis of intrusion detection systems Snort and Suricata. In *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics, 2013.

[141] S. S. Wu, C. C. Liu, and A. Stefanov. Distributed specification-based firewalls for power grid substations. In *IEEE PES Innovative Smart Grid Technologies, Europe*, October 2014.

[142] T. Xu, H. Hou, H. Yu, D. You, X. Yin, and Y. Wang. Analysis on IEC 61850 Interoperability Support. In *IEEE Power Engineering Society General Meeting*, June 2007.

[143] Dayu Yang, Alexander Usynin, and J Wesley Hines. Anomaly-based intrusion detection for SCADA systems. In *5th Intl. Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies (NPIC & HMIT 05)*, pages 12–16, 2005.

[144] Hyo-Sik Yang, Sang-Sig Kim, and Hyuk-Soo Jang. Optimized security algorithm for IEC 61850 based power utility system. *Journal of Electrical Engineering and Technology*, 7(3):443–450, 2012.

[145] Y. Yang, H. T. Jiang, K. McLaughlin, L. Gao, Y. B. Yuan, W. Huang, and S. Sezer. Cybersecurity test-bed for IEC 61850 based smart substations. In *2015 IEEE Power Energy Society General Meeting*, July 2015.

[146] Y. Yang, K. McLaughlin, S. Sezer, T. Littler, E. G. Im, B. Pranggono, and H. F. Wang. Multiattribute SCADA-Specific Intrusion Detection System for Power Networks. *IEEE Transactions on Power Delivery*, 29(3):1092–1102, June 2014.

[147] Y. Yang, H. Q. Xu, L. Gao, Y. B. Yuan, K. McLaughlin, and S. Sezer. Multidimensional intrusion detection system for iec 61850-based scada networks. *IEEE Transactions on Power Delivery*, 32(2):1068–1078, April 2017.

[148] Yi Yang, K. McLaughlin, Lei Gao, S. Sezer, Yubo Yuan, and Yanfeng Gong. Intrusion detection system for IEC 61850 based smart substations. In *2016 IEEE Power and Energy Society General Meeting (PESGM)*, July 2016.

[149] Yi Yang, Kieran McLaughlin, Timothy Littler, Sakir Sezer, Bernardi Pranggono, and HF Wang. Intrusion detection system for IEC 60870-5-104 based SCADA networks. In *Power and Energy Society General Meeting (PES), 2013 IEEE*, 2013.

[150] Man-Ki Yoon and Gabriela F Ciocarlie. Communication pattern monitoring: Improving the utility of anomaly detection for industrial control systems. In *NDSS Workshop on Security of Emerging Networking Technologies (SENT)*, 2014.

[151] Lichen Zhang. Multi-view approach to specify and model aerospace cyber-physical systems. In *Computational Science and Engineering (CSE), 2013 IEEE 16th International COnference on*, pages 595–602, December 2013.

[152] Chunjie Zhou, Shuang Huang, Naixue Xiong, Shuang-Hua Yang, Huiyun Li, Yuanqing Qin, and Xuan Li. Design and Analysis of Multimodel-Based Anomaly Intrusion Detection Systems in Industrial Process Automation. *IEEE Transactions on Systems, Man, and Cybernetics Systems*, 45(10):1345–1360, October 2015.

[153] Bonnie Zhu and Shankar Sastry. SCADA-specific intrusion detection/prevention systems: a survey and taxonomy. In *Proceedings of the 1st Workshop on Secure Control Systems (SCS)*, 2010.