Cockpit CI

FP7-SEC-2011-1 Project 285647

Cyber-security on SCADA: risk prediction, analysis and reaction tools for Critical Infrastructures

D2.3 – Modelling and prediction of QoS by heterogeneous modelling paradigms-Final

Organisation name of lead contractor for this deliverable
ENEA

General information	
Submission date	5 November 2014
Dissemination level	Confidential
State	Final Version
Work package	WP2000 - Modelling and prediction of QoS of interdependent SCADA and Telco Networks facing cyber attacks
Tasks	Task 2003 - 2004
Delivery date	31 August 2014

Editors

Name	Organisation
E. Ciancamerla, M. Minichino, T. Patriarca	ENEA

Authors (limited to the ones which provided active contribute to this version)

Name	Organisation
E. Ciancamerla, B. Fresilli, M. Minichino, S. Palmieri, T. Patriarca,	ENEA
L. Lev	IEC
S. Iassinovski	Multitel
T.Cruz, J. Proença, L. Rosa	FCTUC

Reviewers

Name	Organisation	Date
A. Graziano	SELEX	21/10/2014
L. Lev	IEC	24/10/2014

Executive Summary

This deliverable makes leverage on the following WP2000 results:

- An overview of modelling techniques and tools able to represent Industrial Control Systems (ICS) under cyber attacks. ICS include Supervisory Control and Data Acquisition (SCADA) systems (D.2.1).

- The Reference Scenario that is composed by a SCADA system, its Medium Voltage electrical power grid and a corporate network. The electrical grid, its SCADA system and the corporate network are interdependent System of Systems and they act as a whole. Topologies, main functionalities, main devices, main communications among devices of such a System of Systems, including communication protocols, with special attention on TCP/IP based protocols, interdependencies, cyber security issues such as cyber threats, vulnerabilities, pre-existent cyber security policies and technical solutions, attack cases (Malware spreading, Denial of Service and Man in the Middle (MITM), are described. The attack cases are described in terms of attack characteristics, attack initiation sources, attack targets and, when possible, expected consequences, coming from engineering judgement.

The deliverable describes the status of the Modelling and prediction of Quality of Service indicators of the interdependent Systems of Systems (SCADA, its electrical grid and the corporate network) under cyber attacks as described in the Reference Scenario.

The contents of the modelling effort within the deliverable reflect the assumption that, at the state of the art, no single modelling technique has the credible modelling power and the analytical tractability to adequately deal with the Quality of Service (QoS) of the Systems of Systems, such as the Medium voltage electrical grid and its SCADA system, under cyber attacks as described in the reference scenario.

To reach such an objective, a Modelling framework instantiated on the reference scenario is proposed. The framework allows one to describe the Interdependent Systems of Systems of the scenario, their elements, messages and message routes, vulnerabilities, states, attack and consequences scenarios, as well as influence of incorrect functioning on QoS indicators. The framework tends to model not only different cyber attack types, cyber attack spreading and electric infrastructure functioning, but namely the cyber attack influence on the functioning of the electric grid controlled by a vulnerable SCADA Control Centre, over a vulnerable communication infrastructure, which includes a portion of corporate network.

In such a framework different methodologies, models and tools have been investigated.

From the overview of modelling techniques and tools (D2.1) we extracted and investigated the most relevant ones:

1. The SIR model of epidemics: it may be used in cyber security to study how a malware infection spreads among different machines. SIR model represents a disease spread where individuals are susceptible to a disease, potentially contract the disease, recover and become immune to future infections after recovery. In order to compute the injection and spreading of malware within corporate network and SCADA, SIR models are implemented via the open source tool Netlogo.

2. The Attack Tree, which is basically a Fault Tree with the attack goal in place of a fault and where basic event probabilities are not failure rates. Most recently, attack trees have been applied to a SCADA communication system. The apparent limitation of Attack Trees is their limited modelling power, largely not sufficient to represent sophisticated attacks and the ADvanced Persistent Threats. On the contrary they are very immediate to address attack consequences.

3. The ADversary VIew Security Evaluation (ADVISE) method. The approach is to create an executable state-based security model of a system and an adversary that represents how the adversary is likely to attack the system and the results of such an attack. The method, which relies on the formalisms of the Stochastic Activity Networks, is ideally able to provide insight on weak points in the system defense, considering characteristics of both the system and its adversaries. It has been developed by Performability Engineering Research Group (PERFORM), Center for Reliable and High-Performance Computing, University of Illinois.

From the Cockpit CI partners expertise the following methodologies, models and tools have been investigated:

4. Agent based simulation, with the support of the Intelligent RAO simulator intends to address high-level, inter-system behaviour simulation. The idea is to develop a composite simulation model including specific simulation tools for high granularity subsystems (communication infrastructure, software to a certain extent, etc.) while using the RAO simulator for intersystem behaviour representation at the higher level (cyber attack scenarios level).

5. Attack Graph for security risk. The idea is to analyse how QoS parameters at service level would be affected as a result of fluctuation in the security risk level.

6. Risk prediction by holistic reductionist approach - This approach is made by two layers. The first layer, named holistic situation assessment, considers each infrastructure as a whole and evaluates the impact of faults or services using domain simulators. The second one can be considered as a reductionist impact assessment layer that is built out of experts reviews and tries to assess interdependencies and how faults and their consequences are reflected on other facilities.

7. Temporal network reliability analysis - the usual assumptions in probabilistic network analysis are that nodes and links are binary entities (up or down) with a probability assigned to the two states, nodes are non characterized and undifferentiated and the network elements (nodes and links) are statistically independent. The two previous standard assumptions do not cope with the complexity of CIs and should be extended in two directions. In real networks, (like electrical grids, aqueducts or telecommunication networks), a producer (denoted as source) may feed many consumers (denoted as sinks) and a consumer may be fed by different producers. We refer to this problem as the multi-source multi-sink reliability problem. The edges (and nodes) should be enriched with an attribute or weight characterizing their main function (e.g. capacity, bandwidth, resistance, cost, length), so that performance indices and QoS can be quantitatively computed. In CIs, time is often a crucial aspect of QoS delivery and the network description is augmented with time specifications. Here we explicitly examine the two new aspects documented before and describe the efforts to arrive to an analytical model that can provide timely and accurate information about the reliability status of the interacting SCADA and Power Grid

system, and that can rapidly be adapted to the changing configurations of the interacting networks. The effort is to arrive to an analytical model that can provide timely and accurate information about the reliability status of the interacting SCADA and Power Grid system, and that can rapidly be adapted to the changing configurations of the interacting networks. Models are under investigation by means of WNRA academic tool.

8. Composing epidemic and performance models - along the different phases of a cyber attack the Fault Isolation and System Restoration (FISR) service, performed by SCADA, has degraded time responses which affect the quality of power to grid customers. It is discussed a composite model implemented by means of NETLOGO to represent the injection and the spreading of the malware SCADA and corporate network, and by NS2 to represent DoS and MITM attacks and their consequences on indicators of Quality of FISR service and in turn the quality of power to grid customers.

9. Attack and defence tree - we start with a preliminary example, to better explain the theory under the attack and defence tree, the goal of the attack, represented by the root of the attack tree, that is the acquisition from an unauthorized user (hacker) of the root password of a Unix server with consequent possible attack to the system. Then the attack and defence tree is applied to SCADA system, assuming a more general architecture of SCADA, with attacks which may penetrate along three main lines: i) the Remote Terminal Unit (RTUs), the Master Terminal Unit MTU, which stores and processes the information from RTUs, and the network, composed by a proprietary private WAN with a redundant, that connects the RTUs to the MTU; ii) through the primary control centre (composed by two main blocks a SCC and a HMI) and its backup (composed by a switch and the backup SCC and HMI); iii) the central LAN and the equipment and facilities connected to the LAN, like the historian Data Base, and the Web service to the customers.

Some of these approaches have been already discussed on D2.3 preliminary deliverable. Here we report on separate sections just the ones not yet discussed.

Different numerical indicators of QoS of SCADA and in turn of the electrical grid, to be evaluated along the different phases of a cyber attack are proposed. Models should be ideally able to predict, possibly in real time, the degradation of such QoS indicators as expected consequences of a successful cyber attack. It should be considered that any attack can act on/influence the Observability and the Controllability of Remote Terminal Units from SCADA Control Centre. In such cases, SCADA operator, partially or completely, looses the supervision and control of the physical infrastructure, in our case the MV electrical grid.

Finally, the limits of modelling approach in cyber security are presented and then a test bed to conduct actual cyber attacks on SCADA and analyze their consequences on SCADA and on electrical grid are investigated. Within such a scope, ENEA remote Test Bed and its link with IEC HTB is presented and the cyber attacks implemented within ENEA and IEC HTB are discussed. Specifically, in the last subsection of this deliverable, we present the ENEA remote Test Bed functionality, its architecture and the cyber attack that we have realized locally.

Table of contents

List of figures

List of tables

1 Introduction

1.1 Context

CockpitCI is in line with the MICIE project [35] of which it resumes the main concept, i.e. that by increasing the cooperation among infrastructures it is possible to provide the operator with a better situation awareness in the presence of adverse events and therefore increase the CI level of service (business continuity). CockpitCI proposes this concept again in a wider operational range which includes cyber attacks among the adverse events. To implement such concepts, CockpitCI project is investigating a tool that could detect and react to anomalies and cyber anomalies on SCADA and corporate network of a utility, to be demonstrated on a Medium Voltage electrical grid and its SCADA. The tool development process ideally makes leverage on the:

- Reference scenario

and is fed by an adequate subset of

- Heterogeneous models of QoS delivered to CI customers in nominal conditions and under cyber attacks of SCADA and corporate network.

Reference scenario intends to ideally identify the whole set of knowledge, information and data needed to develop and demonstrate CockpitCI tool. Reference Scenario is composed by a SCADA system of a Medium Voltage power distribution grid, interconnected with a corporate network. Power distribution grid, its SCADA system and the corporate network are interdependent System of Systems and they act as a whole. Topologies, main functionalities, main devices, main communications among devices of such System of Systems, including communication protocols, with special attention on TCP/IP based protocols, interdependencies, cyber security issues such as cyber threats, vulnerabilities, pre-existent cyber security policies and technical solutions, attack cases, are described. Main functionalities described in reference scenario [67], and here retrieved, consist in:

- The procedure of Fault Isolation and System Restoration (FISR) of the power distribution grid. Such a procedure is executed by SCADA operator on a permanent failure of the Power distribution grid;

- The Fault identification and handling procedure of the corporate network.

The main aim of CockpitCI heterogeneous modelling is to understand if and how cyber attacks may degrade the functionalities (in terms of availability, continuity and performances) of Reference scenario according to adequate Quality of Service (QoS) indicators. Models should be aware of a credible pre-existent cyber security policies and technical solutions in use by the utility and in particular by an electrical grid utility. Modelling techniques able to represent cyber attacks, their exploitation throughout cyber vulnerabilities of Critical Infrastructures (CIs), up to penetration within Industrial Control Systems (ICSs) and SCADA have been investigated. A special attention has been paid to the ability of such techniques, tools and models to predict the impact of successful attacks on the Quality of Service (QoS) of Industrial Control Systems of which SCADA systems is a subset, and in turn on the QoS delivered by the target CI as described in the reference scenario.

1.2 Aim

The deliverable describes the status of the Modelling and prediction of Quality of Service indicators of the interdependent Systems of Systems (SCADA, its electrical grid and the corporate network) under cyber attacks as described in the reference scenario [67].

The contents of the deliverable reflect the assumption that, at the state of the art, no single modelling technique has the credible modelling power and the analytical tractability to adequately deal with the QoS the Systems of System under cyber attacks as described in the reference scenario [67]

A Modelling framework instantiated on the reference scenario is proposed. The framework allows one to describe the Interdependent Systems of Systems of the Scenario, their elements, messages and message routes, vulnerabilities, states, attack and consequences scenarios, as well as influence of incorrect functioning on Quality of Service indicators. The framework tends to model not only cyber attack spreading or electric infrastructure functioning, but namely the cyber attack influence on the functioning of electric grid controlled by vulnerable SCADA Control Centre (SCC) over vulnerable Communication Infrastructure (CI).

Different numerical indicators of QoS of SCADA and in turn of the electrical grid to be evaluated along the different phases of a cyber attack are proposed. Models should be ideally able to predict possibly in real time the degradation of such QoS indicators as expected consequences of a successful cyber attack. It should be considered that any attack can act on the false/lack of Observability and Controllability of Remote Terminal Units from SCADA Control Centre. In such cases, the operator partially or completely has a false/looses the supervision and control of the physical infrastructure, in our case the MV electrical grid.

1.3 Document Structure

This section describes the contents of the chapters of the deliverable, that are preceded by the **Executive Summary**.

Chapter 1 deals with context, aim and structure of the document.

Chapter 2 introduces a view and a relationship among Critical Infrastructure interdependency and operation, cyber attacks, Quality of Service and risk analysis and cyber security of SCADA.

Chapter 3 focuses on methodologies and tools for SCADA security modelling, at the state of the art, considered in building CockpitCI models.

Chapter 4 proposes the Modelling framework instantiated on the Reference Scenario. The framework tends to model not only cyber attack spreading or electric infrastructure functioning, but namely the cyber attack influence on the functioning of electric infrastructure controlled by vulnerable SCADA Control Centre over vulnerable communication infrastructure. This chapter gives a general view of modelling framework.

Specific cyber attacks, their characteristics and when possible their consequences are specified in **chapter 5** of the deliverable. Three kinds of cyber attacks are considered:

Malware spreading, Denial of Service (DoS) and Man in the Middle (MITM). Each attack is specified in terms of peculiar characteristics, attack initiation sources, attack targets and expected consequences. Furthermore, different numerical indicators of QoS of SCADA and of electrical grid are proposed. QoS indicators reflect the fact that any attack can act on the Observability of Remote Terminal Units from SCADA Control Centre and on the Controllability of Remote Terminal Units from SCADA Control Centre. In such cases, the operator partially or completely looses the supervision and control of the physical infrastructure, in our case the MV electrical grid.

Chapter 6 discusses a composite model implemented by means of i) NETLOGO to represent the injection and the spreading of a malware throughout SCADA and corporate network elements and ii) by NS2 to represent DoS and MITM attacks and their consequences on quality of FISR service indicators and, in turn, on quality of power to grid customers.

Chapter 7 describes the simulation model of the Electrical Critical Infrastructure under cyber attack, developed by RAO Simulator. Test simulations have been run according to three cases: i) FISR execution in normal CCI and SCADA operation; ii) Cyber attack before the beginning of FISR procedure; iii) cyber attack which is still spreading during the FISR process. In all the three cases, simulation results are described.

Chapter 8 deals with attack and defence tree. The theory, underlining the attack and defence tree, is explained by means of a preliminary example, then the attack and defence tree is applied to a SCADA system, assuming a general SCADA architecture, under attacks which may penetrate along three main lines: i) the Remote Terminal Unit (RTUs), the Master Terminal Unit (MTU), which stores and processes the information from RTUs, and the communication network, composed by a proprietary redundant WAN, that connects the RTUs to the MTU; ii) through the primary SCADA Control Center (SCC), (composed by two main blocks a SCC and a HMI) and its backup (composed by a switch and the backup SCC and HMI); iii) the central Local Area Network (LAN) and the equipment and facilities connected to the LAN, like the Hystorian Data Base, and the Web service to the customers.

Chapter 9 deals with "Modelling versus Test Beds". Models hardly relies on the assumptions which characterize the actual world, including SCADA and ICT technology world. More over cyber security is a very complex and dynamic argument, far to be well understood and completely captured by modelling. Laboratory activities may help in better representing the actual world, understanding the value of model parameters, validating models and improve their adherence to the actual world. A test bed laboratory, with the heart at the Israel Electric Corporation and terminals distributed among the other partners of the project, has been realized. Particularly, at ENEA, a test bed laboratory is going to be realized where cyber attacks can be reproduced on a simple mock up of SCADA, and parameters analyzed.

Chapter 10 reports a short discussion and some conclusions.

References, Appendix 1, Glossary and Acronyms and symbols are respectively reported in **Chapters 11, 12, 13 and 14.**

2 Critical Infrastructure interdependencies versus cyber security modelling

Critical Infrastructures (CIs) are complex physical and cyber-based systems that form the lifeline of a modern society as such infrastructures include power and drinking water production, telecommunication network and the management of the transport systems [78]. Consequently, the security of such infrastructure is not an option as the consequences could have far reaching impacts on such sensitive areas as the economy and national security. The consequence of the compromise of a CI can even reach calamitous proportion when considering how tangled today's services are. For instance, transport infrastructures such as high speed trains and undergrounds rely on electric or nuclear power for operating. Similarly energy generated by those power plants is used to power up such service of vital importance as hospitals and banks. With this interdependency between key services in mind, one can quickly appreciate how far reaching a security incident may have down the line.

A different type of dependency that is not to be under-estimated is one that concerns the interconnection between modern SCADA system and the ICT network. Such an interconnection was designed to achieve a better management and a close monitoring of the service provisioning process. Although the benefit of such an exercise is unequivocal, it has also led to the rise of a number of concerns emanating from the cyber-systems, most of which are security related. Indeed, the recent years have witnessed a sharp rise in the number of attacks directed at the CI through exploits of vulnerabilities in the midst of the cyber-system. According to the US Homeland Security department, 198 incidents have been reported in 2011 compared to only 9 in 2009. Worms and virus have been the weapons of choice for such attacks. For illustration, in 2003, the SQL Slammer worm effectively paralysed the SCADA system of two US utilities and nuclear power plant by saturating the bandwidth of the carrier used for the communication. In the same year, the Sodig virus shut down the train signalling system in Florida resulting in delays in train schedules. The STUXNET worm infection of the Iranian nuclear plant at Natanz in June 2010 perfectly represents the frailty of the regulatory systems devoted to control the CIs. STUXNET is a computer virus specifically designed for attacking Windows based industrial computers and taking control of PLCs (Programmable Logic Controllers), influencing the behaviour of remote actuators and leading to instability phenomena or even worse. In this particular incident, STUXNET resulted in the centrifuges spinning uncontrollably, causing damages. Another variation of the STUXNET virus DUQU, and the Flame virus dissected in 2011 are known to target CI with the aim to modify information within or about the infrastructure.

Overall, the consequences of compromising CIs could range from disruption of the underlining activities to destruction or disclosure of sensitive information. Still, very often and especially in the case of utility services, such incidents lead to a decline in the level of QoS provided through the CI. As a result it becomes difficult to model security of CIs without accounting for the propagation of the impact within inter-related services. Such modelling would help to reach an understanding on how a state of cyber insecurity may affect the expected level of vital interdependent services and, take some corrective measures towards keeping the QoS level acceptable, as the evidence of the security compromise is becoming evident.

CockpitCI project intends to investigate a tool that could detect and react to cyber anomalies on SCADA and corporate network of a utility, to be demonstrated on a medium voltage electrical grid.

Within the project there is the challenge of combining Critical Infrastructures interdependency versus cyber security aspects.

The modelling and analysis of interdependencies between CIs is a very important field of study and intensive research efforts are documented in many countries around the world [1, 2]. The main challenge in dealing with CIs is that the interactions often create complex relationships, dependencies, and interdependencies that cross infrastructure boundaries. Modelling is complicated by the quality and availability of data, intricacy of systems, complexity of interactions between infrastructure sectors, presence of natural or malicious faults and implications and sensitivity of results.

Research projects are directed to develop models that accurately simulate critical infrastructures behaviour and identify interdependencies and vulnerabilities. The results of these simulations are directed to provide knowledge to locate critical situations, enhance redundancy, reduce costs, and to prepare for and respond to emergencies.

An excellent survey of the state of the art about the ongoing research in the field of critical infrastructure interdependency modelling, mainly in the US but with a look to the rest of the world, is given in [3]. To give an idea of the difficulties to be faced in developing simulation tools for CIs, [3] claims that “Critical Infrastructure interdependency modelling has many of the same challenges that one can expect with any modelling and simulation domain: data accessibility, model development, and model validation. Interdependency modelling is further complicated by the extremely large and disparate cross sector analysis required”. In the appendix of [3], the survey presents a number of leading research efforts in US and in the world. Going through the appendix it can be recognized that this activity can be successfully carried on only inside very large public or private research organizations.

A specific and more restricted area regarding CI operation is the one of cyber attacks [4]. Threats to Critical Infrastructures are increasing across the globe. Denial of Service attacks, network intrusions, financially motivated attacks and cyber war are orchestrated by nation states (foreign nations), terrorists and organized cyber criminals. Nations across the globe have low levels of preparedness to meet cyber attacks that have devastating impact. Countermeasures must include appropriate legislation and enforcement through security governance frameworks. A recent and exhaustive survey of the threats and vulnerabilities that systems incur with recommendations on how to protect the system is given in [5]. The risk associated to the pervasive use of information technology and how to manage the security related problems in complex private or public organizations is addressed in [6]. Information systems are subject to serious threats that can have adverse effects on organizational operations, organizational assets, individuals, other organizations by exploiting both known and unknown vulnerabilities to compromise the confidentiality, integrity, or availability of the information being processed, stored, or transmitted by those systems. Guidance for an integrated, organization-wide program for managing information security risk is provided in [6], where it is stated that security risk related to the operation and use of information systems is just one of many components of organizational risk. Other sources of risk can include: program management risk, investment risk, budgetary risk, legal liability risk, safety risk, inventory risk, supply chain risk, and security risk.

Mainly in the US, but also in Europe and in the rest of the world, a number of nations’ Critical Infrastructure and government systems are administered by the private sector, but many companies are reluctant to share information about attacks for fear of regulatory sanctions and negatively affecting stockholder confidence. Effective partnerships with governments is recommended [5]. The biggest obstacles to ensuring security are cost, lack of security awareness and the inability of security folks to persuade decision makers to deal with real threats. Security is not a market differentiator for critical industries. The current focus of Critical Infrastructure has been reliability and availability and not security. Making security a top priority in Critical Infrastructures is perhaps the number one counter measure.

2.1 Critical Infrastructure Quality of Service

Although there is no universally accepted definition of QoS, the concept commonly refers to a defined and expected measured of performance or dependability of a system or network. Several taxonomies for QoS parameters have been introduced in the literature, notably in the publications of Truong et al. [79] and [81]. In general the classification of these parameters is made within two categories. The first of such categories relates to QoS measures for assessing the performance of a particular service and includes such parameters as latency, jitter, bandwidth availability, etc. The second category is about the dependability aspect of a service and is assessed through such indicators as availability, accessibility, accuracy, reliability, capacity, ease of management and security. A detailed taxonomy of QoS is provided in Figure 1.

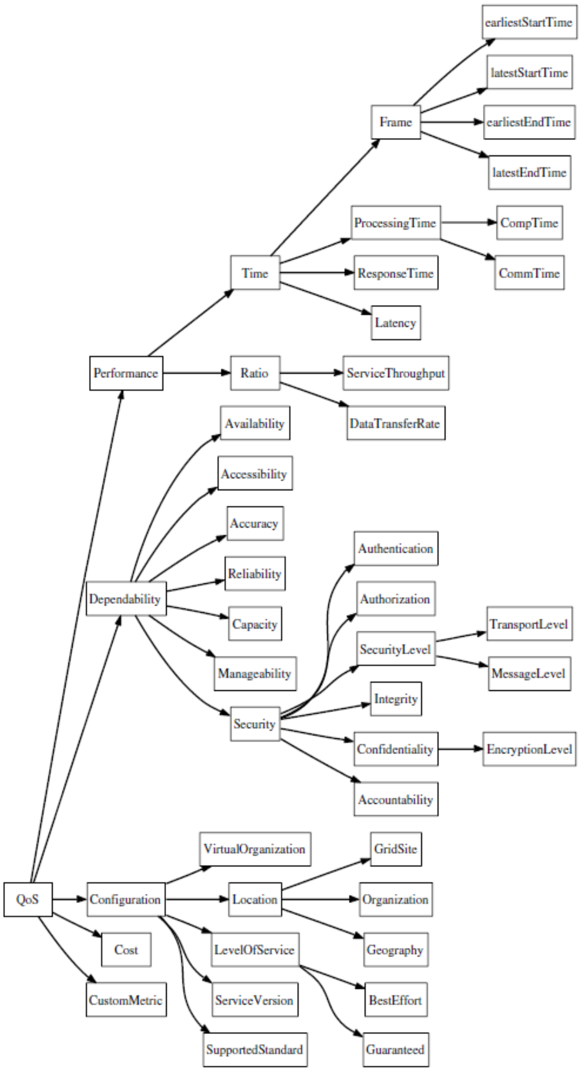


Figure 1: QoS Taxonomy [79]

When considering the assessment of the impact on the QoS due to a security incident that occurred in a system, two assessment approaches can be considered.

The first of such alternative involves an assessment of the QoS through methods based on simulation or analytical methods with the actual impact on the QoS estimated with quantitative or qualitative indicators. The use of simulation and analytical methods involve the replication of system components or to first represent the system through a mathematical model such as Markov chain or Petri Net. This is then followed by a step by step monitoring of the performance or dependability parameters that are of interest, while gradually injecting some attack within the system.

Alternatively, one can rely on expert knowledge for reasoning on the possible impact on the QoS, once the level of security risk is determined. We resorted to this latest approach for qualitatively working out the impact of a security compromise on QoS parameters. This has the merit to provide a generic framework that can be effectively tailored to adapt to each system without the need to have an in-depth knowledge of the system of interest, as it is the case for the first option. For the systematic evaluation of the impact of a security incident on QoS, we have tailored a four step methodology (Figure 2), which consists of: (i) model the interdependency between the CI services; (ii) model the attack tree associated to each service's critical components; (iii) estimate the level of risk in the services of a CI; (iv) a qualitative assessment of the impact of a cyber-attack on some parameters of quality of service (QoS) such as reliability, availability and response time of infrastructure services.

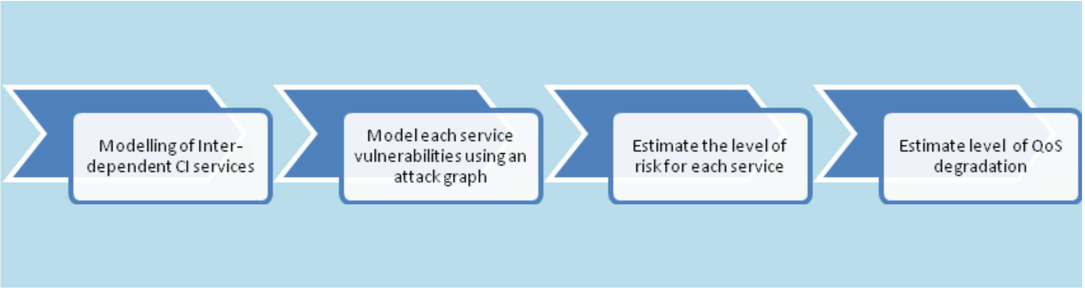

Modelling of Inter-dependent CI services
Model each service vulnerabilities using an attack graph
Estimate the level of risk for each service
Estimate level of QoS degradation

Figure 2: Steps of the methodology

2.2 Interdependent Critical Infrastructure services

Our modelling of a CI is based on a service-oriented approach for representing the CI through its essential services. Each service is then represented by its hardware and software components that are critical for its well-functioning. By critical component of a service, we mean any component that when compromised, will greatly hamper the overall operation of the system, therefore affecting the QoS of the resulting service.

This modelling allows capturing both the logical and physical links between services belonging to the same infrastructure. Being able to identify such links is paramount for a subsequent identification of the set of the services that may be affected as a result of a security attack targeting a certain component or service. The diagram of Figure 3 shown below illustrates the modelling of the CI.

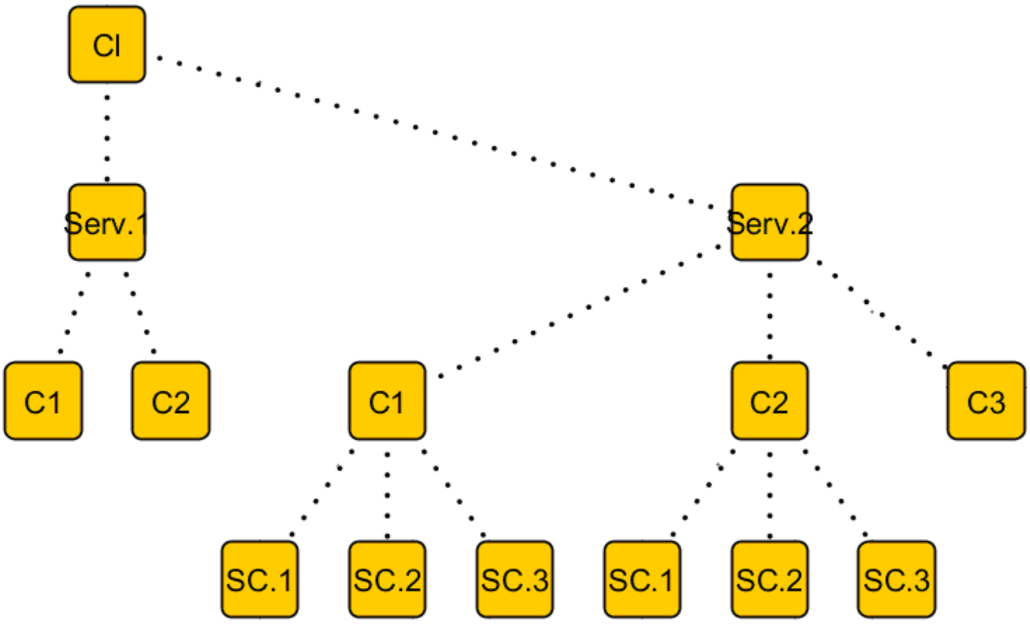


Figure 3: Modelling of interdependent services for a CI

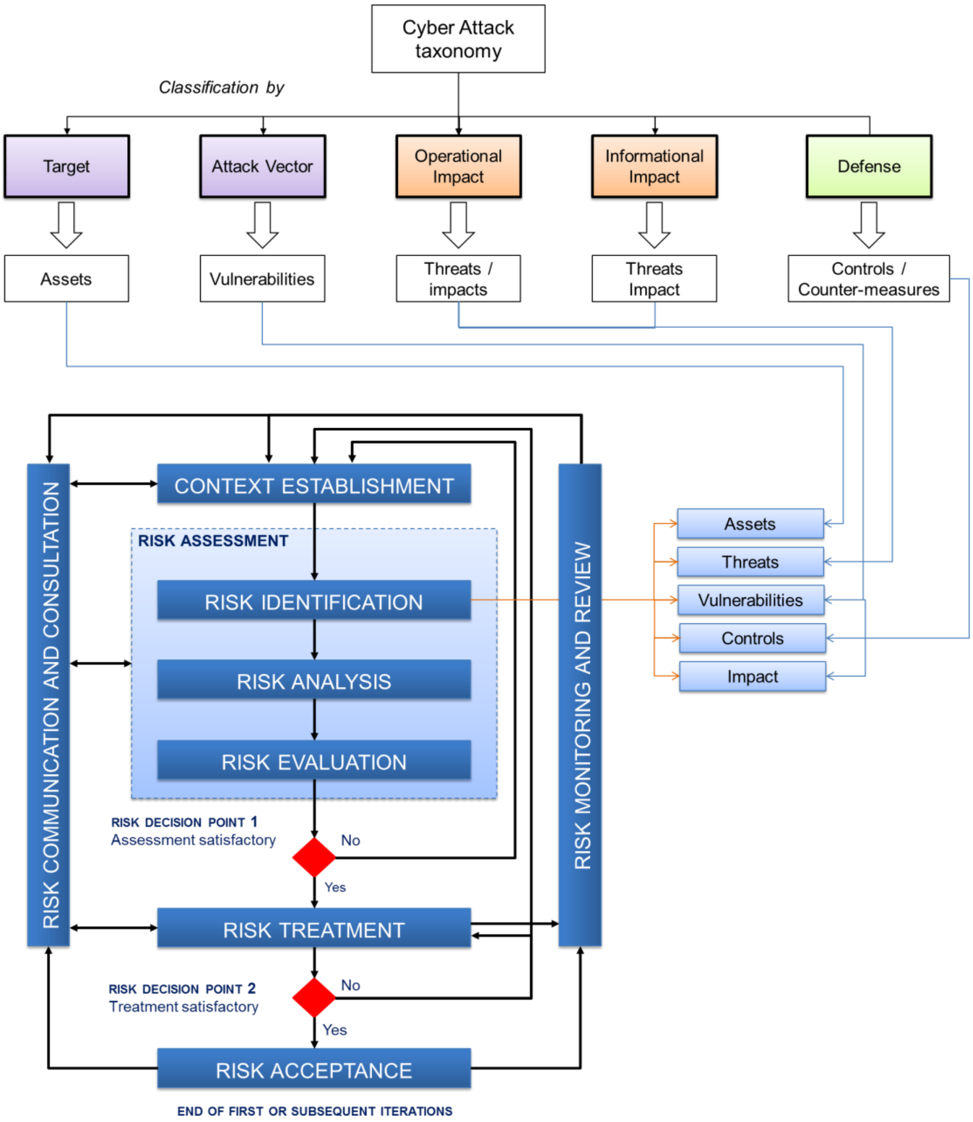
2.3 Cyber attacks and risk analysis

The cyber attack taxonomy (see Appendix 1) allows us to perform a risk analysis compatible with the ISO 27005 standard, as the identification of attack is in line with the risk identification of the standard.

The risk analysis approach is a systematic approach to predict or estimate the impact of cyber attacks on the Quality of Services delivered by a CI.

However, considering the complexity of the Critical Infrastructure, the Quality of Services delivered to customers which depend upon the hardware/software/internal services, and the plethora of cyber attacks and CI vulnerabilities to be exploited, a rigorous risk analysis approach could result unplayable.

According to the standard, the risk analysis aims at assessing the probability of a threat to successfully exploit vulnerability and to assess the impact of a successful attack according to a fixed scale. The risk evaluation has to assess the level of risk according to a fixed scale of risk level. In CockpitCI these steps of the risk analysis are performed by the cyber-simulation and by the prediction tool. The Incident Management Team (IMT) will perform the risk treatment.



2.4 Cyber security of SCADA

Ever since the integration of SCADA systems with ICT based network, the security concerns of ICS based system have grown dramatically. Reasons as to why SCADA systems get attack vary from individual setting themselves as challenge to get through a certain defence mechanism, to industrial espionage or sabotage. In either case, it is now commonly accepted that the highest threat to SCADA are those that try to disable the system or to be able it control it remotely with the intent to damage equipment or processes.

According to [87], the move from proprietary to standardized protocols for SCADA meant that documentation is available to such an extent that anyone could use similar protocols on a Personal Computer (PC) and the same radio and could actually sit nearby and start sending instructions to the RTUs. Similarly, the connectivity to the Internet implies that most anyone with a link to the Internet could bump into the SCADA system and potential bypass its security if such security is not stringent enough.

Among the various security problem encountered in a number of critical systems, networks and operational control systems (SCADA) have been reported to be under repeated cyber attack from high level adversaries, and the security of SCADA system has become of primary concern. The main problem related to the security of SCADA systems can be defined as quoted in [7]: “most SCADA systems and protocols were designed long before network security was perceived to be a problem. The traditional SCADA system was a closed serial network that contained only trusted devices with little or no connection to the outside world. As control networks evolved, the use of TCP/IP and Ethernet became common place and interfacing to business systems became the norm. The result was that the closed trust model no longer applied and vulnerabilities in these systems began to appear. In particular, network security problems from the business network and the world at large could be passed onto process and SCADA networks, putting industrial production, environment integrity and human safety at risk”.

An excellent guide on how to recognize and tackle the problem to improve security in Industrial Control Systems is provided in the NIST Special Publication SP-800-82 [8].

Initially the report [8] emphasizes the peculiarities of ICS systems with respect to traditional Information Technology (IT) systems. In the past, ICS had little resemblance to traditional Information Technology systems in that ICS were isolated systems running proprietary control protocols using specialized hardware and software. Widely available, low-cost Internet Protocol (IP) devices are now replacing proprietary solutions, which increases the possibility of cyber security vulnerabilities and incidents. As ICS are adopting IT solutions to promote corporate business systems connectivity and remote access capabilities, and are being designed and implemented using industry standard computers, Operating Systems (OS) and network protocols, they are starting to resemble IT systems.

The increasing use of wireless networking places ICS implementations at greater risk from adversaries who are in relatively close physical proximity but do not have direct physical access to the equipment.

Originally, ICS implementations were susceptible primarily to local threats because many of their components were in physically secured areas and the components were not connected to IT networks or systems. However, the trend toward integrating ICS systems with IT networks provides significantly less isolation for ICS from the outside world than predecessor systems, creating a greater need to secure these systems from remote, external threats. Although some characteristics are similar, ICS also have characteristics that differ from traditional information processing IT systems. Peculiar differences are based on the following characteristics:

- ICS are generally time-critical, are usually continuous in nature and thus require very high availability, while throughput is typically not essential. On the contrary IT are designed to get high throughput while some level of delay or error (non satisfied requests) are tolerated.

- ICS have unique performance and reliability requirements and often use operating systems and applications that may be considered unconventional to typical IT personnel. Furthermore, the goals of safety and efficiency sometimes conflict with security in the design and operation of control systems.

- While IT are designed to preserve data confidentiality and integrity, safety other than (or more than) security is of primary concern in ICS. Improper function of ICS may endanger human life, public health, loss of equipment or products. ICS designers and operators must understand the link between safety and security.

- ICS have a strong interaction with physical processes and with the environment, so that logic executing in ICS has a direct affect on the physical world. The security function in ICS must not compromise or interfere with the correct relation with the external physical world.

- ICS components and operating systems may not have typical desirable functionalities like, data encryption, error logging, password protection.

- IT components and systems have typical useful life of the order of 3-5 years due to the technological obsolescence. ICS have lifetimes of the order of 15-20 years and longer.

- IT component and system are usually accessible and designed to be easily and quickly repairable. ICS component can be isolated, difficult to reach and repair.

Possible incidents an ICS may include the following [8]:

- Blocked or delayed flow of information that prevent real-time operation;

- Unauthorized change of instructions;

- Inaccurate information that cause inappropriate actions;

- Software infections of various kinds;

- Interference of the safety with the security functions.

More specifically in [7], eleven goals have been identified that an intruder might attempt to achieve against a SCADA system. These eleven goals range from gain access to various parts of the system, disable components, disrupt communications and compromise data. Furthermore, four intermediate objectives have been identified that may not constitute one of the final goals but may be used to achieve an objective. These may be: Denial of Service (DoS) attacks, interception and modification of data, TCP sequence number attack and sniff traffic. A useful table summarizes the attacker goals with some related features: technical difficulty, severity of impact, probability of detection, underlying critical vulnerabilities. The use of the data reported in the table may be useful in a more detailed analysis of intrusion detection using ad hoc simulation tools.

The report [8] not only indicates the possible risks associated with SCADA operation but identifies also the possible source of the threats. Threats to control systems can come from numerous sources, including adversarial sources such as hostile governments, terrorist groups, industrial spies, disgruntled employees, malicious intruders, and natural sources such as from system complexities, human errors and accidents, equipment failures and natural disasters. To protect against adversarial threats (as well as known natural threats), it is necessary to create a defense-in-depth strategy for the ICS. The categorization of the possible adversary is also useful in the implementation of an intrusion detection simulation tool.

With the aim of helping in building protected ICS, the report [8] suggests some useful architectural principles. When designing a network architecture for an ICS deployment, it is usually recommended to separate the ICS network from the corporate network. The nature of network traffic on these two networks is different: Internet access, File Transfer Protocol (FTP), e-mail, and remote access will typically be permitted on the corporate network but should not be allowed on the ICS network.

A basic rule to achieve a reasonable level of security is to introduce a simple two-port firewall between the corporate and control networks, as shown in Figure 4. Properly configured, a firewall significantly reduces the chance of a successful external attack on the control network.

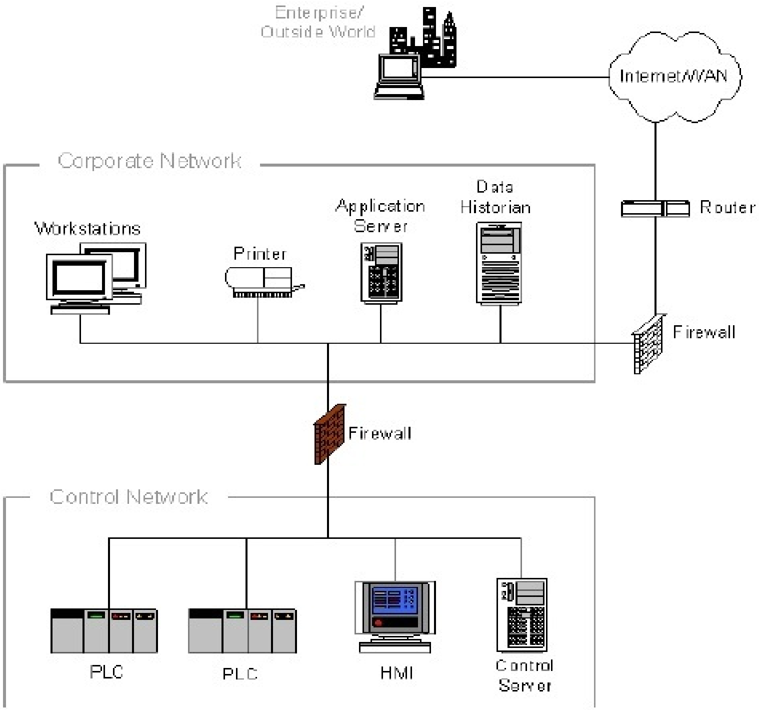


Figure 4: Firewall between Corporate Network and Control Network

To improve the security the Report [8] indicates how to transfer some services in a Demilitarized Zone (DMZ). A DMZ is a physical or logical subnetwork that contains and exposes an organization’s external services to a larger untrusted network, usually the Internet. The purpose of a DMZ is to add an additional layer of security to an organization’s Local Area Network (LAN); an external attacker only has access to equipment in the DMZ, rather than any other part of the network. Each DMZ holds one or more critical components, such as the data historian, the wireless access point, or remote and third party access systems. In effect, the use of a DMZ-capable firewall allows the creation of an intermediate network. Creating a DMZ requires that the firewall offers three or more interfaces, rather than the typical public and private interfaces. One of the interfaces is connected to the corporate network, the second to the control network, and the remaining interfaces to the shared or insecure devices such as the data historian server or wireless access points on the DMZ network. Figure 5 provides an example of this architecture.

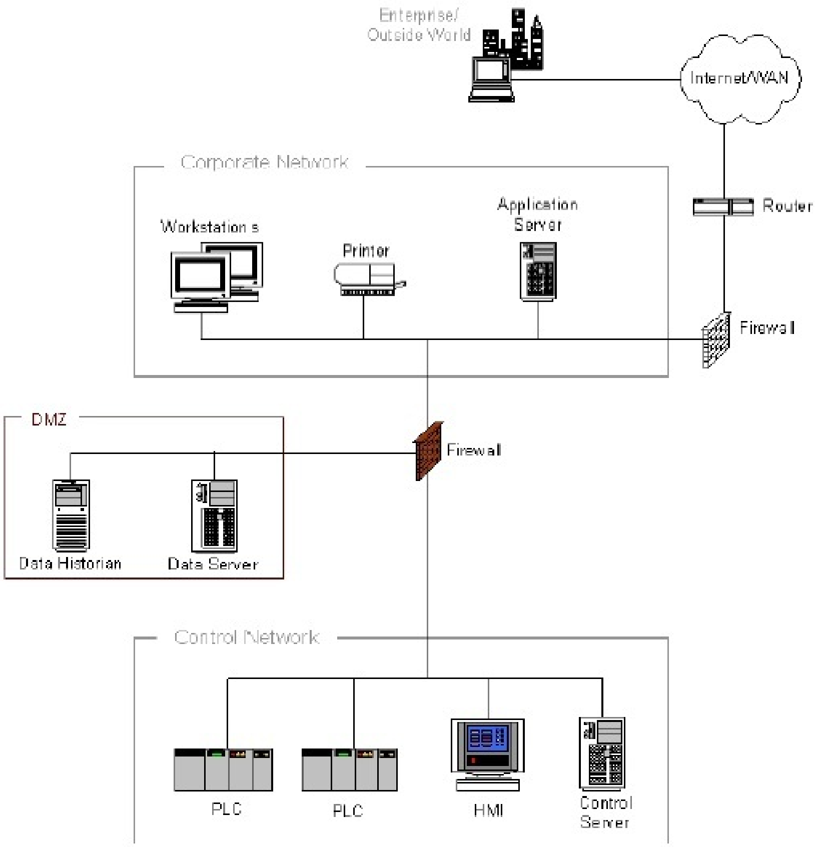


Figure 5: Paired Firewalls between Corporate Network and Control Network

3 Methodologies and tools for SCADA security models

Cyber security methodologies, tools and models are fundamentally based on identification of attacker profiles, attack objectives, attack steps characterization, spreading throughout ICS network and consequences on CI customers.

In [9] different cyber security methodologies, models and tools have been discussed, used as a single package to address specific aspects of the attack scenario, and/or integrated together to afford the whole attack scenario:

- Attacks/attacker/vulnerability models
  a. Attack/vulnerability trees
  b. Petri nets
  c. Game theory
- ICS & corporate network models
  a. Network simulators
  b. Emulators
- CI models
  a. Power flow simulators
- Composite models
  a. To represent more than one aspect of the attack scenario (at least two different kinds of the previous models) till the whole attack scenario (i.e. attacks model plus ICS & corporate network model plus CI model)
  b. Require more than one (Hybrid versus homogeneous) method and tool .

Several tools which cover partially or as a whole the above methods and models, have been identified. Some of them are PENET, ADVISE, I2SIM,CISIA, NETLOGO, RAO,NS2.

According to the underlined formalisms, many of them rely on the stochastic approach as Petri nets, Game theory, Markov chains, Bayesian networks, Monte Carlo methods. Others rely on different approaches such as Agent based simulation, discrete event simulation, etc.

3.1 Stochastic approach

In the stochastic approach, models involves probabilities, or randomness, associated with time and events. A state transition diagram can be used to describe all relevant operational system states and the possible transitions between these states. To describe time aspects between events, a rate matrix has to be specified. One usually assumes that the event that will occur next, as well as the time until the next event, is random. Hence, the behaviour of the system is a stochastic process. The main advantage of this modelling approach is that it captures the dynamic system behaviour, i.e., the sequence and time aspects of events, such as failures and repairs. The stochastic process can then be used as a basis for quantitative analysis of the modelled system. By using mathematical analysis techniques, closed-form solutions may be obtained, which describe how the failure and repair rates affects the expected system dependability in terms of its reliability, availability and so forth. In many cases, the stochastic modelling approach is the most appropriate system evaluation method when quantitative dependability measures are needed.

According to the definition of dependability provided in [10], dependability comprises several system properties, amongst them also the Confidentiality, Integrity, Availability (CIA), typically, security attributes. One would therefore expect that security can be modelled and analyzed by the same methodologies as the other dependability properties. However, it turns out that this is not the case. The main reason is that malicious behaviour is rarely considered as a possible fault source when evaluating system dependability.

This means that the stochastic modelling approach that is so useful when analyzing systems to obtain quantitative measures cannot be applied as it is to evaluate security properties. At the state of the art different approaches try to overcome this problem by proposing methodologies that makes it possible to incorporate attacker behaviour into the transition rates of a stochastic model, so that a comprehensive system evaluation can be performed.

Modelling Malicious Behaviour

Given that a system is represented by a stochastic model, the execution of a transition caused by malicious behaviour will henceforth be referred to as an attack action. It is assumed that a large number of adversaries, i.e., attackers, can target the system simultaneously. This is a realistic assumption for most of the networked ICT systems of today, which are on line round the clock. By studying log files one can see that these systems are constantly subject to more or less suspicious activity, such as probing, worm activity or other kinds of vulnerability exploitation. The rate value of a transition in the stochastic model, which represents an attack action, will then model the accumulated failure intensity, given that all attackers will always try to attack the system. Unfortunately, this rate value is in itself not enough to accurately describe the expected time before the transition actually will occur. One of the main reasons is that attacks are not truly random processes. Because attackers act with intent, they are not always well characterized by models of a random nature [11].

For example, assume that the system that is to be evaluated is a small corporate LAN consisting of a private fileserver, a publicly accessible web server and a router connecting the LAN to the Internet. Now assume that the expected time a remote attacker would need to break into and read access restricted files on the fileserver is about the same as the expected time needed to break into and deface the web server. The latter can be characterized as an integrity failure and the former as a confidentiality failure. However, in practice it may be much more common that web servers get defaced than that fileservers get compromised. In fact, the network administrator of this particular LAN assess the frequency of the former to be five times as high as the latter. When using a stochastic model to evaluate this system, the rate values of these two security failures must represent the actual occurrence rates of the events, rather than the success rates of the individual attack actions.

Attacks that are caused by human beings, and that lead to security failures, are very often highly intentional with the specific aim of causing maximum benefit to the adversary or damage to the system. The basic idea is that the probability of an attack will depend on not only the expected time (or effort) required to perform the attack but also on how motivated the particular attacker is. As already seen, there are a number of factors that drive humans to attack computing system, such as financial gain, curiosity, pure entertainment, a rise of ego, etc. On the other hand, a number of factors may reduce the attacker's motivation and make him refrain from certain attack actions. For example, an employee, with a user account on the corporate LAN, may put his future career at risk if he tries to abuse his insider privileges to attack the local computer network. The gain from a successful break-in into the fileserver may therefore be smaller than the possible consequences he will experience if the intrusion is detected by the system administrator. As another example, the illegal aspect of actions (criminal offense) may prevent even a remote attacker to use available tools to exploit vulnerabilities in such networks. Even though the expected time or effort to perform an attack action may be randomly distributed, the decision to perform the attack will therefore be a trade-off between the gain from a successful attack and the possible consequences of detection.

Attacker behaviour is represented as a probability distribution over all the possible attack actions available in a particular system state. These probabilities are then reflected in the transition rates of the stochastic model by weighting the corresponding (accumulated) attack intensities. For example, if an attacker will choose a particular attack action with probability 0.5, then we can expect 50% of all attackers to take this action, given that they all share the same motivation. Hence, by introducing attack probabilities as parts of the transition rates, the result from a successful attack can be modelled as one or more intentional state changes of the underlying stochastic process, which represents the dynamic behaviour of the system. This is illustrated in Figure 6 where 1 is a good system state, 2 is a (security) failed system state, *"a"* is an attack action, λ_{12}(a) is the accumulated attack intensity (given that all attackers always take action "*a"* and π_1(a) is the probability of action "*a"* in state 1.

Some stochastic modelling approaches can be considered high-level approaches in that they focus on the impact of the intrusions on the system rather than on the specific attack procedures themselves. This facilitates the modelling of unknown attacks in terms of generic state transitions. For example, in the stochastic model depicted in Figure 6 the attack *"a"* can simply be explained as "the action that seeks to transfer the system from the good state 1 to the failed state 2".

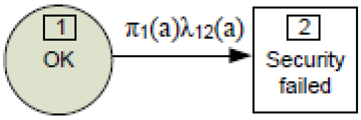


Figure 6: A stochastic model with assigned failure rate [12].

Attacks modelled as a series of state changes

Attacks on an operating computer system can often be modelled as a series of state changes of the system that lead from an initial secure state to one or more target compromised states, i.e., security breach states. A successful attack against the system may therefore consist of many subsequent elementary attack actions. At each intermediate stage of the attack, the attacker will therefore have the choice of either i) *Attack* by performing the next elementary step in the attack (the system will be transferred from state i to state i + 1, If the attacker succeeds; the system will remain (temporary) in state i, If the attacker fails) or ii) *Resign* and interrupt the ongoing attack (the system will be remain (temporary) in state i). On the other hand, at each intermediate stage, the system administrator may *detect* the attack and bring the system back to a secure state (the system will be transferred from state i to state 0, hence, the attacker will not have the possibility of continuing the attack).

Figure 7 shows the attack stages. In the model it is assumed that once an attack is initiated, the attacker will never voluntarily try to revert the system to any of the previous states.

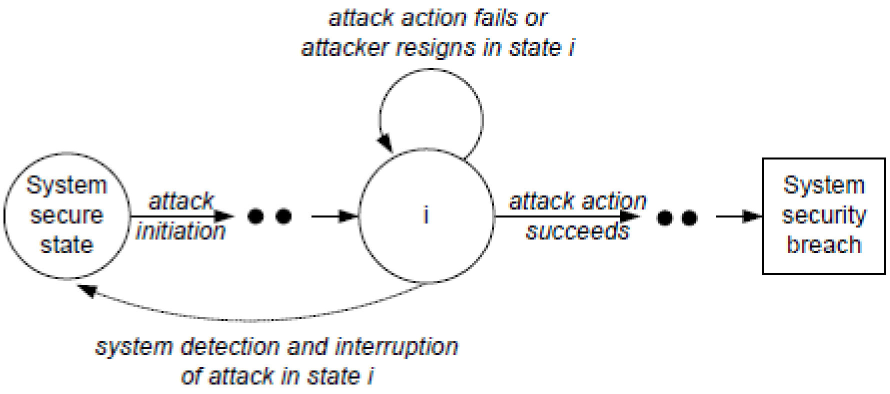


Figure 7: Penetration of a computer system modelled as a series of state changes [12]

The model also assumes there is only one single path to the security breach state; a somewhat simplified view of reality. Since the state transition model presented in Figure 7 is stochastic by nature, the time spent in each state of the system model will be a random variable. The time or effort taken for an attacker to cause a transition will depend on several factors, such as the attacker's knowledge and background, robustness of the system etc.

3.2 Game theory

Game theory has been perceived as natural way of modelling cyber security. Indeed, a game is a description of the strategic interaction between opposing, or co-operating, interests where the constraints and payoff for actions are taken into consideration [13]. Depending on the nature and amount of information held by each player locked in a play, a game can be perfect or imperfect, complete or incomplete, static or dynamic. A game is labelled perfect when all players involved in the game are aware of the set of actions that an adversary player has already taken. Conversely, an imperfect game is one where at least one player does not know the next moves of an opponent. A complete game depicts one where all the players are well accounted to the strategy of their adversary and their objectives. However, the set of actions that may be taken towards meeting such objectives may not be necessarily known. The distinction between a complete game and a perfect game resides in the fact that it does not take into account the actions each player has already taken [13]. By analogy, a game is said to be incomplete when at least one player is not aware of some of the strategy and objective of a certain player. A game is said to be static if no players can change his/her strategy during the course of the game. Generally speaking, a static game is considered as a one off game as each player plays up his/her strategy in one go without subsequent move left. A static game is an imperfect game by nature as no further information as what the next move of an adversary player will be. As opposed to a static game, players in the context of a dynamic game choose their strategies as the game is being unravelled.

Game theoretic based approach to cyber security

Reference [14] investigates the usefulness of game theory to capture information warfare. In the paper, the authors reviewed four different games before discussing how a dominant position can be achieved and maintained through the orchestration of an appropriate strategy.
The first of such games involves two armies engaged in a military warfare, with one set to use its technological capability to disable the enemy's Command, Control, Communication, and Intelligence (C^3I) before the actual military offensives take place. The second example used by the authors concerns a cyber attack on such critical infrastructure as nuclear and electricity power plants, telecommunication, water and gas, using DoS tools, virus and others worms. The ultimate aim of the attackers in this case is to wreak havoc and nurture fear, in the midst of the society. The third example discussed by the authors has great similarities to the previous one as it involves a terrorist attack on a number of business and companies that may be key to the economy of a country. The successful launch of such an attack depends on attacker being able to gather information on the targets and also in determining the optimal timing for such an attack. The fourth and final example involves a dormant warfare which aims at collecting strategic information related to the economy and technology in view of hindering progress.

Having applied a game theory approach to these examples, the authors concluded that:

- A bold strategy is required to force an enemy to believe that a player will not accept any threats.

- Mixed strategies can mitigate the dominative position of the attacker, especially when any defense strategy is effective only against a specific attack strategy. Changing the defense strategy somehow randomly will increase the probability of mitigating attacks.

- An attacker should overload a network only part of the time, so that the defender will not stop using the network completely.

- Maintaining a dominating position requires the stronger player to limit the long term costs to the weaker party since this may otherwise lead to a rebellion leading to damages on both sides.

The authors of the reference [15] have argued that a comprehensive grasp of an Attacker's Intent, Objectives and Strategies (AIOS) is key to a successful risk assessment and harm prediction. Subsequently, the author proposed a game theoretic approach to inferring AIOS. A brute force DDoS attacks is used as a case study in the experiment conducted by the authors to demonstrate how attack strategies can be inferred in real world attack defense scenarios. Some of the key findings of the authors are that the security and assurance of the system greatly depends on the appropriate selection of the game model. Furthermore, the effectiveness of the IDS and the correlation of the attack actions play a role in the determination of the best AIOS game models.

Reference [13] adopts a game theoretical approach to the modelling of a DoS and DDoS in network systems. The precept of such initiative lies on the potential of game theory concepts to capture the realm of cyber security: that of two entities competing for contradictory payoffs. Indeed, a Distributed Denial of Service is modelled as a two-player game in which the attacker attempts to find the most effective packet sending rate or botnet size, while the defender or network administrator is concerned with putting in place the best firewall setting to block unwanted traffic while allowing the legitimate traffic through. A DoS is represented with a single attacking node while multiple nodes are used in the context of a DDoS. In both cases, the authors assumed that the malicious nodes are operated by one attacker and that, two possible cases can be considered. The first of such cases considers the game as being static, i.e. neither the administrator nor the attacker change their strategy during the course of the game. In this set up, the strategy of the attacker is confined to a couple of actions including: the selection of the malicious nodes, the size of the botnet (m) to launch the (D)DoS and the set-up of the rate of malicious traffic (r_A). Conversely, the defender or network administrator can only change the mid-point (M) of the firewall which represents the rate of packets being dropped by the firewall.
The Nash equilibrium of this game is defined to be a pair of strategies (r_A m, M), which represent the best strategy for both players. The author remarked that the peculiarity of a dynamic game makes it hard to actually compute the Nash equilibrium since the change in strategy by both players may result in a continuous shift of the latter. For instance, the authors highlighted that, an attacker A can think that if he/she sets r_A low and m high during the first few time steps, the defender D will set M to a low value, and then A can exploit it by setting r_A high and m low in the next few time steps assuming that D does not change M. A similar reasoning can be adopted by the defender based on assumption made about the attacker' behaviour.

A Markov game approach to the assessment of risks is proposed by [16]. The authors argued that a comprehensive assessment of risk in network information systems should account for, not only the current, but also the future risks. The work [16] is based on the extension of the relationship between threat, vulnerability and asset commonly used in the determination of a risk level. They noted that a vulnerability that remains unpatched can help in the spread of risk, while a risk can be considerably reduced if a prompt and decisive action is taken by the administrator. Subsequently, [16] proposed a game where the threat and the vulnerability agents are represented as the players. Thus the threat agent increases the risk through the action "threat spreading" and the vulnerability agent decreases the risk through the action "system administrator's repairing the vulnerability". The ultimate aim of the game is to a get more comprehensive value of risk as well as enabling the system administrator to select the best system repair scheme.

3.3 Attack Trees

Attack trees were introduced by Schneier [17] as a way of formally analyzing the security of systems and subsystems based on varying attacks. This is basically FTA with the attack goal in place of a fault and basic event probabilities are not failure rates. Schneier's work is notable because it was the first to apply this approach to the area of information security. The attack goal is the root of the tree and the different ways of accomplishing the attack are the leaves, with connections via AND and OR nodes.

Moore et al, [18] describe and illustrate an approach for documenting attacks on software systems using attack tree information in a structured and reusable form. Analysts can then use the approach to document and identify commonly occurring attack patterns and then modify attack trees to enhance security development.

Most recently, attack trees have been applied to a SCADA communication system [38]. The authors identified eleven attacker goals and associated security vulnerabilities in the specifications and development of typical SCADA systems. The team defined eleven such goals:

1. Gain SCADA System Access
2. Identify MODBUS Device
3. Disrupt Master-Slave Communications
4. Disable Slave
5. Read Data from Slave
6. Write Data to Slave
7. Program Slave
8. Compromise Slave
9. Disable Master
10. Write Data to Master
11. Compromise Master

Each goal was ranked roughly in terms of the potential severity of impact (e.g. reading data from a slave device is likely less serious as compared to writing data to the slave). Figure 8 shows these basic relationships and ranking. In addition, the study team defined four Supporting Goals that would likely not be an end goal on their own, but would be often required by an attacker to achieve his or her objectives. Each is used in more than one attacker goal. These include:

- Denial of Service Against Networked Device
- Intercept or Modify Data Through Man-In-The-Middle (MITM) Attack
- TCP Sequence Number Attack
- Sniff Traffic

On such basis they suggested best practices for SCADA operators and improvements to the MODBUS standard. Their application was qualitative in that attack tree analysis was used only to identify paths and qualify the severity of impact, probability of detection, and level of difficulty. They did not calculate the probability of an actual attack being successful.

A related approach that arose in the computer and information security literature is vulnerability tree analysis. Vulnerability trees are hierarchy trees constructed as a result of the relationship between one vulnerability and another vulnerability and or steps that a threat agent has to carry out to reach the top of the tree [39]. Vulnerability trees help security analysts understand and analyze different attack scenarios that a threat agent might follow to exploit a vulnerability. With this understanding, countermeasures can be taken.

The top of the tree is known as the top vulnerability or the parent vulnerability. There are a large number of ways that such a top vulnerability can be exploited. Each of these ways will constitute a branch of the tree. The branches will be constructed by child vulnerabilities. Consequently the child vulnerabilities can be exploited by steps that the threat agent will have to perform in order to get to the parent. Each vulnerability will have to be broken down in a similar way. Normally this will end up in more than one level of decomposition. When the point is reached where the branches contain only steps, and no child vulnerabilities, then we know that we have reach the lowest level of decomposition (the “step-only” level).

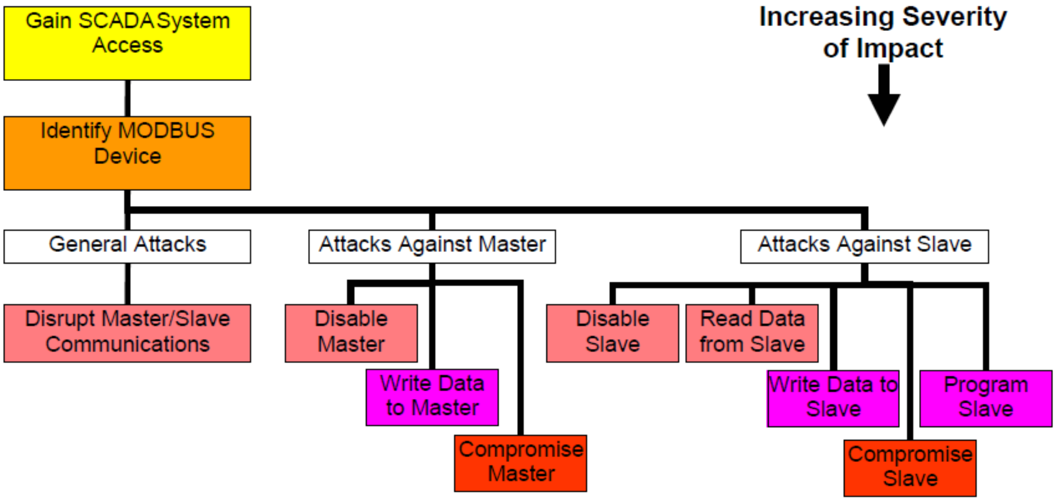


Figure 8: Interrelations and approximate severity of attacker goals [18]

Gain SCADA System Access Attack Tree

Attack tree can be seen as a multi-level hierarchical structure based on logical AND and OR operators.

The top node is the ultimate goal with the grouping of different sub goals. The grouping can be composed with a number of attack leaves that are attributed with logic operators “AND” or “OR”. To build an attack tree also vulnerabilities of the system under attack have to be exploited. In [1], three vulnerability indices are introduced: system, scenario, and leaf vulnerabilities, accounting the power system control framework based on existing cyber security conditions.

To evaluate the vulnerability indices in a systematic manner, the following steps are followed:

- Identify adversary attack objectives.
- Identify possible security vulnerability and construct the attack tree.

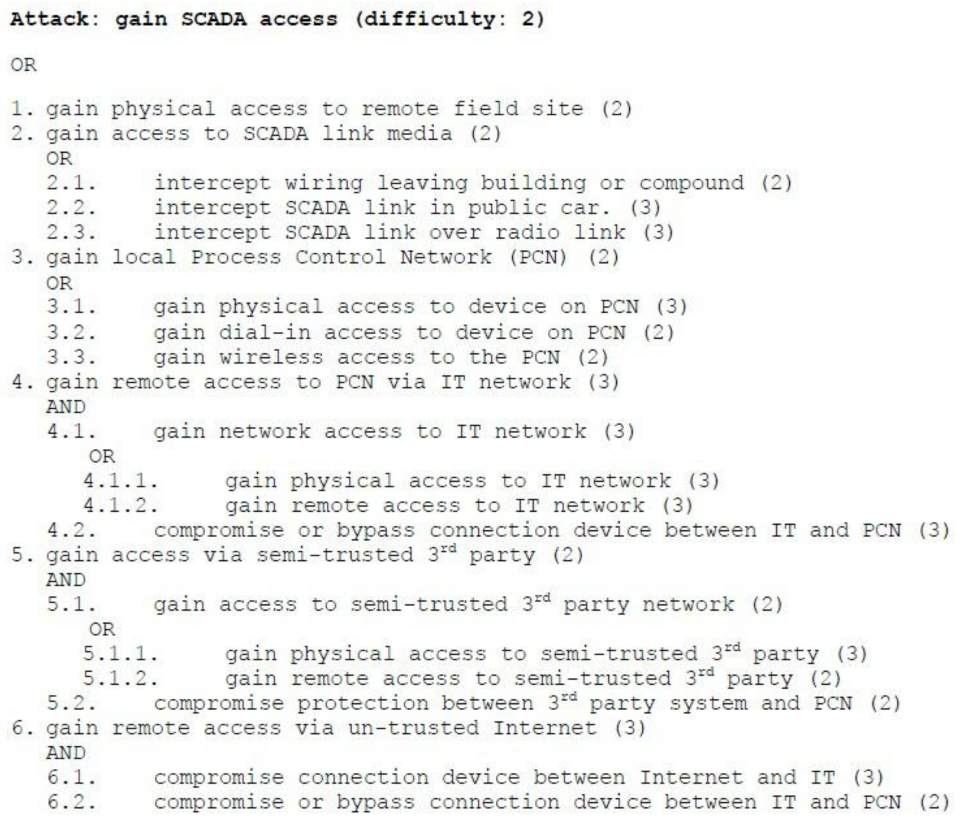
- Determine the combination of intrusion scenarios with each cyber security condition on each attack leaf.

- Compute leaf vulnerability with respect to the password enforcement and existing technological implementations, given that the cyber security conditions are determined.

- Scenario vulnerability can be computed according to the combination of corresponding leaf vulnerability indices.

- Finally, determine the pivotal attack, i.e., system vulnerability based on scenarios' vulnerabilities, and improve system security.

Figure 9 illustrates a possible attack tree for a SCADA system [20], where the difficulty to reach each point can be estimated.



```
Attack: gain SCADA access (difficulty: 2)

OR

1. gain physical access to remote field site (2)
2. gain access to SCADA link media (2)
   OR
   2.1.    intercept wiring leaving building or compound (2)
   2.2.    intercept SCADA link in public car. (3)
   2.3.    intercept SCADA link over radio link (3)
3. gain local Process Control Network (PCN) (2)
   OR
   3.1.    gain physical access to device on PCN (3)
   3.2.    gain dial-in access to device on PCN (2)
   3.3.    gain wireless access to the PCN (2)
4. gain remote access to PCN via IT network (3)
   AND
   4.1.    gain network access to IT network (3)
      OR
      4.1.1.    gain physical access to IT network (3)
      4.1.2.    gain remote access to IT network (3)
   4.2.    compromise or bypass connection device between IT and PCN (3)
5. gain access via semi-trusted 3rd party (2)
   AND
   5.1.    gain access to semi-trusted 3rd party network (2)
      OR
      5.1.1.    gain physical access to semi-trusted 3rd party (3)
      5.1.2.    gain remote access to semi-trusted 3rd party (2)
   5.2.    compromise protection between 3rd party system and PCN (2)
6. gain remote access via un-trusted Internet (3)
   AND
   6.1.    compromise connection device between Internet and IT (3)
   6.2.    compromise or bypass connection device between IT and PCN (2)
```

Figure 9: Attack goal: gain SCADA access

To reach the main goal (the one in bold) it is enough to reach one of the sub goals (one of the possible choices from 1 to 6). Such sub-goals have different ways to be reached, and all work in recursive way. For each leaf, a label can be set to indicate the difficulty of reaching the related sub goal. For setting the difficulty grade of their father, if the sons are in AND, the difficulty grade of the father is chosen as the max of the difficulty grade of his sons, while in case the sons are in OR, the difficulty grade of the father is chosen as the minimum of the difficulty grade of his sons.

The attack trees are very scalable because different trees can be easily joined to make a bigger tree. The limit of the attack tree approach is twofold, one is that enough knowledge is needed to go in deep details as possible, the second one is that the attack tree remains on the paper, but can be supported by automatic tools. The natural evolution is towards the use of Petri nets attack models. Petri nets introduce the concept of attack restoration but reduce the scalability of the related models.

Attack Tree tool

Several tools are available even on the commercial site to implement attack trees. A short description of the attack tree provided by *Isograph*, named AttackTree+, follows [92].

AttackTree+, through the use of attack tree models, allows the user to model the probability that different attacks will succeed. AttackTree+ also allows users to define indicators that quantify the cost of an attack, the operational difficulty in mounting the attack and any other relevant quantifiable measure that may be of interest. Questions such as ‘which attacks have the highest probability of success at a low cost to the attacker?’ or ‘which attacks have the highest probability of success with no special equipment required?’ can be answered using AttackTree+. In AttackTree+, different categories and levels of consequence may also be assigned to nodes in the attack tree. A successful attack may have financial, political, operational and safety consequences. A partially successful attack may have a different level of consequence to a totally successful attack. All these types of consequence measure may be modelled in AttackTree+.

3.4 Petri Nets

Petri Nets (PN) [21,22], in their various shapes and sizes, have been used for the study of the qualitative properties of systems exhibiting concurrency and synchronization characteristics.

The use of PN-based techniques for the quantitative analysis of systems requires the introduction of temporal specifications in the basic, untimed models.

This fact has been recognized since a fairly long time, and several different proposals for the introduction of temporal specifications in PN have appeared in the literature. The main alternatives that characterize the different proposals concern:

- The PN elements (either places or transitions) with which timing is associated,
- The semantics of the firing in the case of timed transitions (either atomic firing or firing in three phases),
- The nature of the temporal specification (either deterministic or probabilistic).

In [23] the idea of using PN for attack analysis introduced by McDermott in [12 of 23] and extended by others such as Zhou et al. [13 of 23] to add some advantages (Colored PN (CPNs), mapping an attack tree to a CPN) or Dahl [14 of 23] (concurrency and attack model) is followed too.

Particularly, they add a new algorithm for the automatic generation of Petri Nets from the description of a SCADA network and its vulnerabilities and propose an approach for risk measures that account for any reachable attack state, given initial conditions on the attacker's access to network resources and host configuration on the network.

Using these measures, one can explicitly account for all high-consequence attack states, irrespective of likelihood, and support a more flexible notion of risk that can be resolved as one of several computable measures on the discrete attack space. The techniques for evaluating these metrics are based on a Petri Net's minimal cover ability set.

in Figure 10, a PN model for a network attack scenario is displayed, where operations against target networks are attack steps involving one or more of the following [23]:

- Improved knowledge of the target network through reconnaissance,
- Access to one or more hosts on the network through exploitation of a software vulnerability or the deception of a legitimate user,
- Increased privilege on one or more hosts on the network through exploitation of a software vulnerability or the deception of a legitimate user,
- The establishment of sustainable access to one or more hosts on the network by, for example installing a back door, or
- Viewing, stealing, manipulating, or preventing legitimate access to protected information

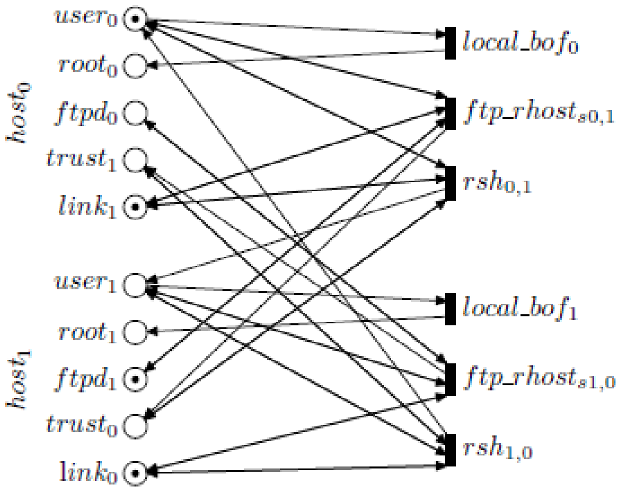


Figure 10: Example attack net [23]

In the model, each attack step is represented by a transition, arrows that point in from places represent preconditions, and arrows that point out to places represent post conditions. The places in the PN of Figure 10 represent host attributes in the network being modelled. The attributes and associated places in Figure 10 include privilege levels (user$_i$, root$_i$), services (ftpd$_i$), trust relationships (trust$_i$), and connectivity (link$_i$).

Ph is the set of places corresponding to host *h*. In order to represent the fact that *h* is characterized by a particular attribute, the corresponding place is marked by a token. Thus *Ph* represents the attributes that host *h* can have; the places in *Ph* that are marked represent the attributes that *h* actually does have. For example, the place $ftpd_1 \in Ph_1$ is marked by a token, indicating that $host_1$ is running an ftp server, while the place $ftpd_0 \in Ph_0$ is not marked, indicating that $host_0$ is not running an ftp server.

For the purposes of attack analysis, transitions represent exploits of vulnerabilities such as buffer overflow (local bof_i), ftp (ftp $rhost_{i,j}$), and rsh ($rsh_{i,j}$). An exploit is intended as any action an attacker takes, including what ordinarily would count as legitimate use of resources, such as the use of *rsh*. For every exploit *e* there is a set of preconditions, represented by a set of places *pre(e)*; and a set of post conditions, represented by set of places *post(e)*. In the example, a precondition for performing a local buffer overflow exploit is that the attacker has user access on the target host, and a post condition is that the attacker has root access on the target host. Therefore, for each *host* h_i, $user_i \in pre(local\ bof_i)$, and $root_i \in post(local\ bof_i)$. The actual occurrence of an exploit is represented by the firing of the corresponding transition. An algorithm has been used to auto-generate the attack Petri net, that executes in three phases: an initialization phase and two processing phases. The initial marking m_0 of the net indicates the conditions that have been met before any transitions in T have fired.

SCADA network on which the attack Petri Net model has been built is comprised of a data historian, a Human Machine Interface (HMI), an engineering workstation, a Master Terminal Unit (MTU), three Remote Terminal units (RTU), and two Programmable Logic Controllers (PLC), as shown in Figure 11. The MTU communicates with the RTUs and IEDs via a Radio Serial Link (RSL), the maintenance server is accessible via dial-up modem from the Public Switched Telephone Network (PSTN), and all other communication is conducted over TCP/IP on Ethernet. In one modelled configuration, a firewall (FW) is used to control traffic between the SCADA network, corporate network (LAN), and the maintenance network. In alternate configurations the historian and workstations are also isolated by the firewall. That is, they reside in separate so-called “DeMilitarized Zones” (DMZs).

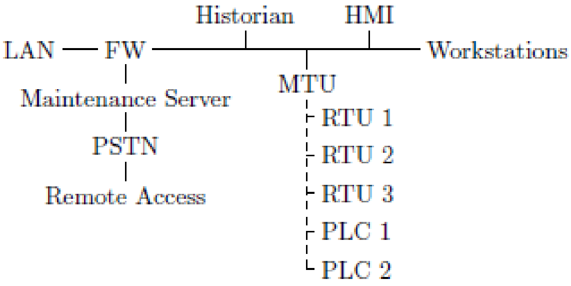


Figure 11: Sample SCADA network [23]

Figure 12 illustrates the PN model of remote manual operation of a valve.

To open the valve, an operator must issue an open command at the Human Machine Interface (HMI), and the valve’s state at the HMI must closed. If these preconditions are met, the HMI relays the command to the Master Terminal Unit (MTU) via the Ethernet connection, the MTU communicates the command to the appropriate RTU via the RSL, the RTU driver delivers power to actuate the valve, and the open state is then registered at the RTU and relayed back to the HMI through the MTU.

Of the large number of possible process failures, [23] discusses six in detail by describing the corresponding component failure, the state of the process at the time of failure, and the resulting impact.

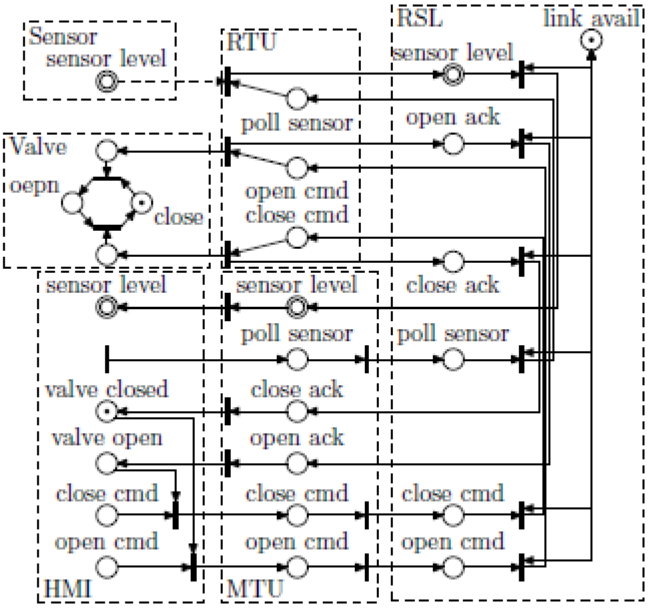


Figure 12: Remote Manual Operation [23]

Each process failure is related to a set of SCADA attacks, where each SCADA attack has the same result as the induced process failure, but is caused by an attack on the SCADA computing infrastructure. Moreover, for each process failure, the authors assign a measure of its severity in terms of expected number of personnel injuries due to inhalation or skin irritation, by ammonia. They relate this process failure and associated consequence to a set of attacks on the SCADA system as shown in Figure 13.

In failure mode "*FM 1.1*" the attacker gains user privileges on the HMI and issues a command to open the valve *v11* before the execution of Task 4, and ammonia will discharge into the dilution drum.

A similar, but possibly more devastating attack can occur in *FM 1.2* when and attacker gains root privileges on the HMI, opens valve *v11* before Task4, and spoofs a closed state for *v11*. This attack gives the legitimate HMI operator the impression that the process state is correct for the task at hand and can increase the amount of ammonia discharged. As a result, the expectation of injuries doubles. A third attack *FM 1.3* targets the MTU. This attack has the same effects as the HMI super-user attack.

Using coverability analysis, they can determine all of the resources an attacker can acquire in the SCADA network. The SCADA attack set will map those sets of resources to SCADA failure modes that can be induced by the attacker, and the system model will analyze the impact of that failure mode.

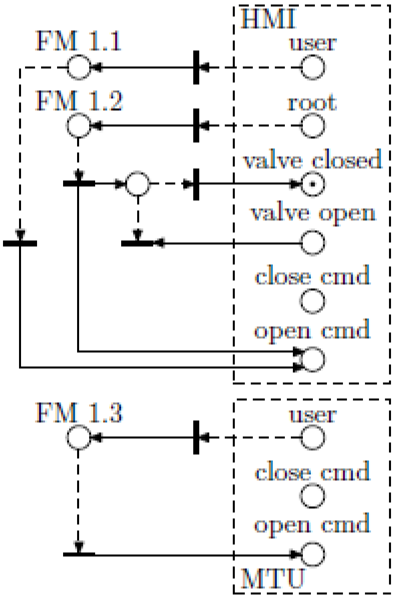


Figure 13: Attack-induced Process Failures [23]

PENET tool

Among tools based on Petri Nets, PENET tool introduces concepts such as the dynamic nature of attacks [24], the reparability of a system, and the existence of reoccurring attacks.

It attempts to find a balance between ease of use and representation power by providing a set of constructs, parameters, performance metrics, and a time domain analysis of attacks. Particularly, users can draw model diagrams of a given system throughout an intuitive user interface, perform time-domain simulations and carry out security evaluations.

Time-domain analysis produces outputs such as “time to reach the main goal” and the “path taken” by the attacker.

PENET tool was completely written in C# .NET using Visual Studio 2005 as a development environment. It requires .NET 2.0 framework to run. Because of these requirements, it is not suitable for operating systems other than Microsoft Windows.

The main contribution of the tool is to extend modelling capabilities of attack trees by using Petri net constructs in order to significantly improve the analytical capabilities of attack trees, specifically by:

- Addressing existing issues in attack trees such as limited representation power, imprecision, and lack of defined defence modelling.
- Introducing concepts of recurring attacks, defence modelling, and dynamic constructs.
- Introducing an analysis approach that follows attack execution in time domain.
- Providing means to evaluate system survivability and defence strategies.

Primary audience of this tool is individuals and organizations who want to use such a tool in vulnerability evaluation of cyber attacks and developing defence strategies for their systems. Secondary audience is research community desiring to learn more about attacker behaviour modelling and PENET approach [24].

Stochastic Petri Net Package

Since attacks occur randomly, a stochastic process can be used for the model. In some studies [26], the intrusion and cyber-net are modelled by a Generalized Stochastic Petri Net (GSPN) model [25]. The states of the stochastic process are the status of intrusions to a network that are inferred from the abnormal activities. These include malicious packets flowing through pre-defined firewall rules and failed logon password on the computer system. Transition probabilities are obtained from the abnormal activity data in the system.

A GSPN consists of two different transition classes: immediate and timed transitions. An arrow head denotes a transition of the system status. An immediate transition is shown as a solid bar. Immediate transitions are assigned probability values. Timed transitions denoted by empty bars have delay times associated with the response that an attacker receives from the system. Tokens (dots inside a circle) are used to model the number of intrusion attempts where an attack starts. Token passing describes the change of each transition, or marking.

SCADA systems typically have specially designed firewall rules and password policies to achieve a high level of computer security. A firewall is a technology of cyber security defence that regulates the packets flowing between two networks. As there may be different security trust levels between networks, a set of firewall rules is configured to filter out unnecessary traffic. These rules are written with the following criteria for acceptance or rejection:

- Type of protocols
- Incoming and outgoing traffic
- Specific port service or a port service range
- Specific IP address or an IP address range

These audit fields are recorded in a firewall and are used offline by a system administrator to analyze malicious behaviours. Due to the high volume of daily network traffic, it is not practical for a system administrator to monitor the network with the available datasets. Thus, an add-on commercial firewall analyzer is implemented to detect anomalies in these datasets.

The malicious packets flowing through a firewall must be identified. Together with the traffic denied by the firewall, such data can determine the probability of cyber attack occurrences either being granted access or being attempted. These datasets can be analyzed from the firewall logs in two ways:

1) The number of records rejected compared to the total number of firewall traffic records, and

2) The number of malicious records bypassing compared with total records for each rule.

The firewall model depicted in Figure 14 includes *n* paths corresponding to *n* rules in the firewall model. The attacker receives responses from the system through the feedback paths starting with the circles representing rules. The paths vertically passing the circles representing rules are successful attempts.

This model consists of two terminals that can be connected to other sub models. For instance, a network that consists of three zones, including a demilitarized zone (DMZ), can be modelled by connecting two firewall models in series. The construction of the model conforms to the number of rules that are implemented in the firewall. In case the number of firewall rules is large, only a subset of rules considered potentially malicious are included in the formulation.

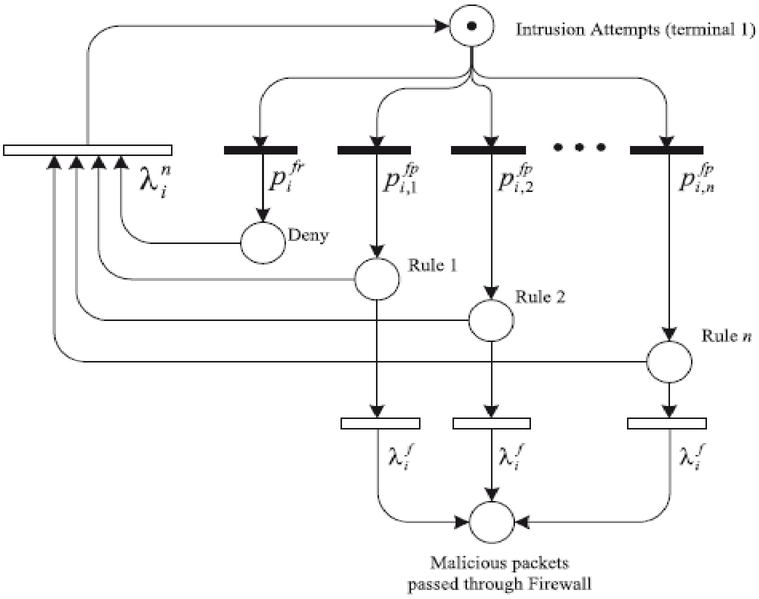


Figure 14: Firewall Model with Malicious n Rules [60]

The sub model consists of circles that are the states representing the denial or access of each rule. Each solid bar is assigned a firewall penetration probability that can be calculated from firewall logs.

3.5 SIR Model of Epidemics

SIR stands for Susceptible, Infected, Recovered and it is an epidemics based model [27] that may be used in cyber security to study how a malware infection spread among different machines. SIR model represents a disease spread where individuals are susceptible to a disease, potentially contract the disease, recover and become immune to future infections after recovery. There is also a variant of SIR called Susceptible, Infected, Removed, that allow infected individuals to die due to the disease and thus leave the considered population. An individual potentially moves from the susceptible to the infected group when s/he comes in contact with an infected individual.

Given specific assumptions on the average number of spread transmission possible from a given infected individual in each period and on the recovering rate of each individual, there are specific algorithms that shows the result of spread transmission; in [27] if individuals are going to die from an infectious disease it is better that they die fast for the purpose of ending the epidemic; the other result is that it is not needed to immunize everyone in the population in order to prevent an epidemic.

There are several analogies between the malware and the epidemics affecting the animal world. Cyber security domain considers each individual as a machine that may be infected by a malware and a recover capability as the action of antivirus software that are in place to remove the infections. Dying individuals represent the machines that have been fatally compromised. In [27], an individual can pass from S to I and from I to R. When R is reached, the subject is removed from the study (this can occur for death or because the subject become immune to this disease). The passage between each state is governed by several variables. In [28,29], the work of Tassier [27] has been tailored to deal with cyber disease spreading along an ICT network composed by a SCADA system interconnected to a corporate network. The network has been simply described by a graph. Each ICT device (an individual in [27]) is a node of the network, and there is an arc if two nodes can communicate each other (the arcs are symmetric).

The Susceptible, Infected and Resistant (SIR) model was originally developed to study the evolution of a disease over a population, where each individual could be susceptible to the infection, having contracted the infection, or be immune/resistant. The similarity with malwares is very high but ICT models as the one in [62,63] got the problem that model variables have different values depending on the cyber security solution adopted for each kind of node.

Let N be the number of the nodes of the net, it is constant. Let j=1,...,N each node, and for every node, d_j is the number of the neighbours of node j. α is the malware spread value. There are several kinds of malwares, and we can differentiate them on the basis of the spreading velocity. A malware that spreads itself too fast can be easily detected due to the high traffic on the net. So, $\beta_j = \alpha \cdot d_j$ indicates on how many neighbours the malware spreads itself. Malware spreads itself just on β_j nodes.

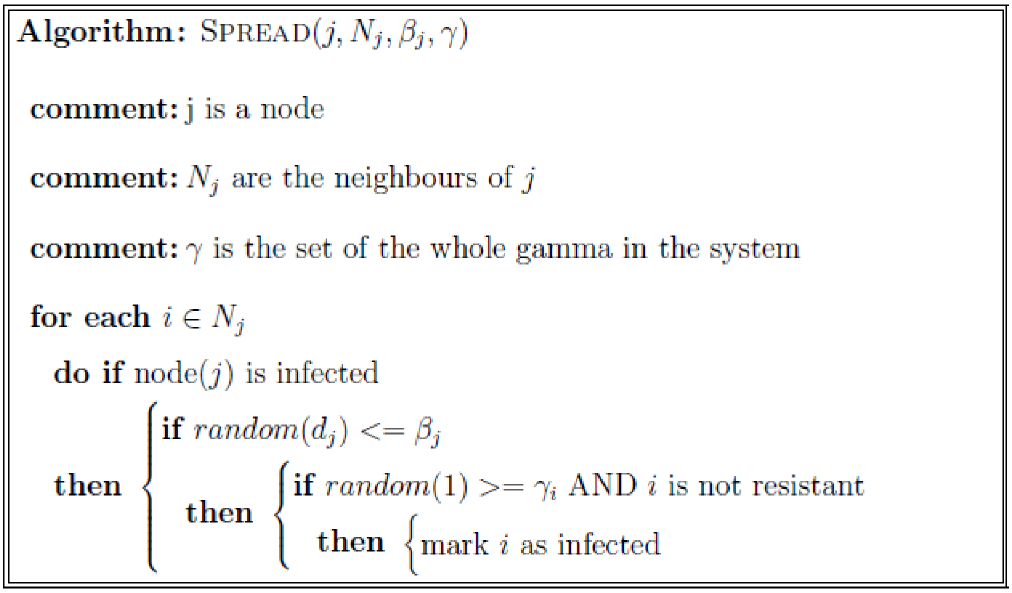
Each host device gets its own security policies (e.g. system patched), or simply relies on an operative system non compatible with the specific malware (e.g. a malware written for Windows cannot infect a Linux machine). A system full patched isn't secure 100% because there are always the zero-days vulnerabilities. Let γ_j that probability (probability to contract the malware).

In [27], once a node becomes infected, a variable (k) keeps into account that after a certain time the node automatically will become resistant. In [28,29] to remove the malware it is necessary to do some actions, such as an antivirus scan or maintenance.

The antivirus is able just to detect malware with a known signature, with a certain probability, or based on a heuristic. ϕ_j is the probability that the antivirus can detect (and then remove) the malware and k_j is the rate at which the scan is performed. In Tassier k, β are constants and ϕ and γ are not defined.

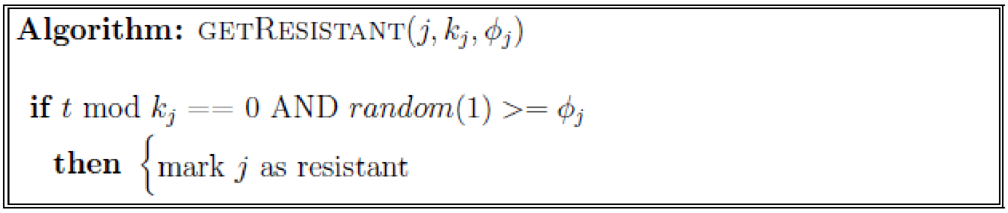
In [28,29] the spreading algorithm is:

At time t:



```
Algorithm: SPREAD($j, N_j, \beta_j, \gamma$)

 comment: j is a node

 comment: $N_j$ are the neighbours of $j$

 comment: $\gamma$ is the set of the whole gamma in the system

 for each $i \in N_j$
   do if node($j$) is infected
     then { if $random(d_j) <= \beta_j$
            then { if $random(1) >= \gamma_i$ AND $i$ is not resistant
                   then { mark $i$ as infected
```

Instead for the resistant:



```
Algorithm: GETRESISTANT($j, k_j, \phi_j$)

 if $t$ mod $k_j$ == 0 AND $random(1) >= \phi_j$
   then { mark $j$ as resistant
```

NetLogo tool

[28] implements SIR model which represents cyber disease spreading along an ICT network composed by a SCADA system interconnected to a corporate network by means of Netlogo. NetLogo [30] is a multi-agent programmable modelling environment. It provides an user interface with three tabs:

Interface tab - This tab is used both by the end-user and by the programmer. The programmer uses this tab to create buttons (e.g. for the setup and the start of the simulation) and the screen for the visualization. The end user indeed uses this tab just for see the simulation process.

Information tab - This is a standard and not modifiable tab that is common for all the NetLogo's program. It can be used by an end user to gain some extra general information about NetLogo.

Procedure tab - In this tab the programmer writes its code. It is composed by several procedures and some variable are sets by the interface tab with some sliders. These kinds of variable are global. In the procedure you can't pass a variable, so if you have to use a variable over several procedure, you have to declare them global.

NetLogo use three types of agents: turtles, links and patches. Mobile agents (turtles) move over a grid of stationary agents (patches). Link agents connect turtles to make networks, graphs, and aggregates. NetLogo allows the creation of sub-kind of turtles and links (called breed). A breed is a collection of agents with the same proprieties. [28] uses breed agents to group the same kind of devices (i.e. with same vulnerabilities/security policies) in order to easily set and/or to specify model variables.

4 Reference scenario modelling framework

This chapter describes the reference scenario modelling framework on which different models which use different tools and formalisms are instantiated. Such a modelling framework allows to describe in a formal way SCADA and corporate network(CCI) elements, messages and message routes, vulnerabilities, states, attack and consequences scenarios, as well as influence of incorrect functioning on quality of service indicators. The framework tends to model not only cyber attack spreading or electric infrastructure functioning, but namely the cyber attack influence on the functioning of electric infrastructure controlled by vulnerable SCADA control centre over vulnerable communication infrastructure. This chapter gives a general view of modelling framework. Specific cyber attacks, their characteristics and when possible their consequences are specified in dedicated chapters of this document.

Particularly, tools and formalisms instantiated in the reference scenario modeling framework:

- to model and analyze malware propagation in relation to the adopted SCADA & CCI security policies, we use NetLogo, a programmable modeling environment for simulating natural and social phenomena;
- to compute FISR performances as a consequence of Denial of Service (DoS) and Man In The Middle (MITM) attacks on specific SCADA & corporate network devices, we use NS2, an open source tool for simulating communication networks and computing performances;
- to calculate QoS values, giving an indispensible information to estimate risk for final electrical customers, we use RAO.

4.1 System elements and connections

From cyber attack modelling point of view the case study can be considered as constituted of three layers - pure electrical infrastructure (without RTUs), HMI of SCADA and CCI and SCADA elements in between serving for information transmission (Figure 15).

The modelling framework for electric grid was proposed in MICIE project (see deliverables of WP2000) and the simulation model of reference scenario fragment of the grid (two feeders, Zuriel and Hanita with connected MV distribution grid fragment) was successfully implemented based on this framework. No modifications are to be done to this framework with respect to new cyber security aspects of CockpitCI project. The reason for this is that electric grid components (poles, lines, feeders, switches, ...) are not subject to cyber attacks.

So, the modelling framework discussed here includes the modelling framework of electric grid from MICIE project with no modifications. We will now concentrate on SCADA and CCI elements modelling under cyber attacks.

These elements serve for information transmission. From cyber security point of view they form a network that can be modelled by a graph with nodes representing SCADA and CCI elements (FIUs, RTUs, gateways, computers, routers, ...) and arcs representing links between elements (fiber optic cables, wires, radio). We postulate that only elements can be subject to cyber attacks, not links. Elements are of different types with respect to their functions and particularities.

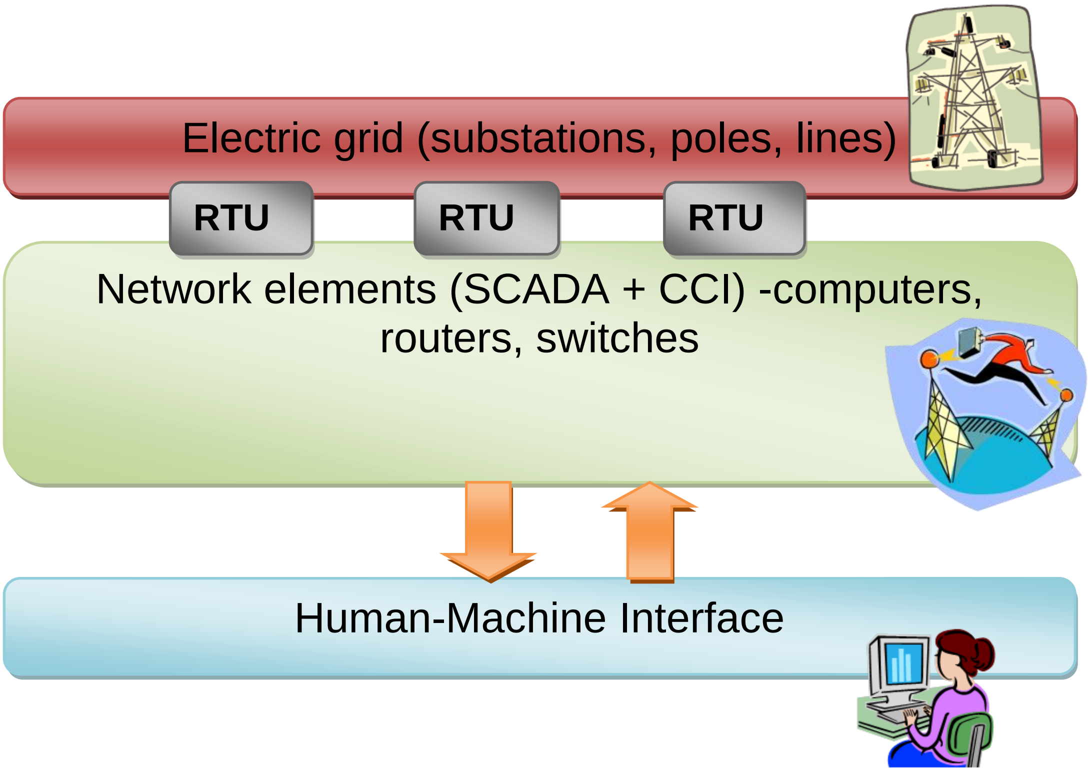

Electric grid (substations, poles, lines)
RTU
RTU
RTU
Network elements (SCADA + CCI) -computers, routers, switches
Human-Machine Interface

Figure 15: Three layers of the system under consideration

4.2 Message and routes

Information exchanged while system functioning consists of messages. Messages are of different type (status data, commands, acknowledges,..). They are sent by sender to destination along specified routes (Figure 16).

Messages have head and payload parts. Message head contain the route and type. Message of a given type has fixed set of parameters representing the payload (the information transported).

A route is a path in the graph representing SCADA and CCI structure. The route is given by a sequence of identification numbers of elements belonging to the route, starting from sender and ending by destination element. Routes are in general case dynamic, i.e. can be redefined at any moment by management system. In general, routes are defined for each sender and can be different for messages of different types.

The information flow is graphically represented by vertical green arrow in Figure 16.

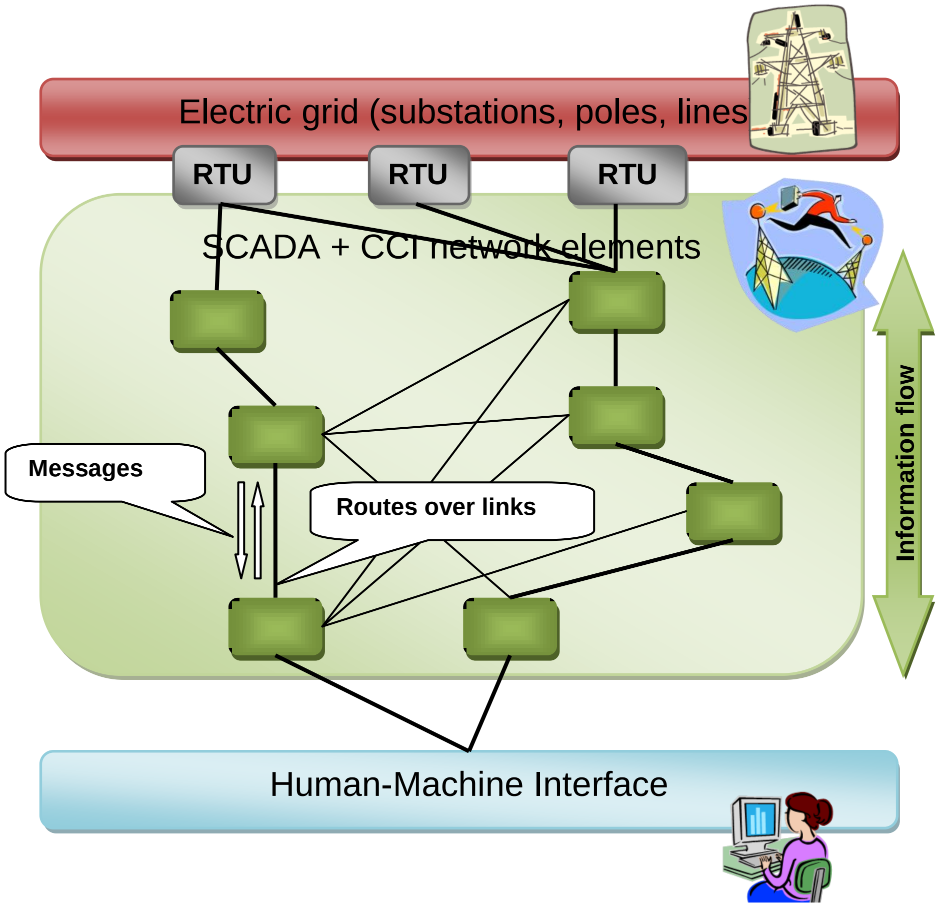


Figure 16: Messages and routes

4.3 Attacks and vulnerabilities

Cyber attack spreading over the system can be presented as horizontal (transversal) flow (orange arrow in Figure 17). This flow infects some vulnerable elements, causing them:

- Spread the attack
- Alter the messages in different ways.

Depending on vulnerabilities (V) of element targeted by cyber attack, the element might be compromised (changes in state S). This potentially leads to two types of consequences (Figure 18):

- First, depending on what is the compromised state, the element can become a source of secondary attack to all connected vulnerable elements.
- Second, the compromised element can issue messages as they were sent by SCADA or alter passing messages in different ways, such as destruction, wrong routing, changing payload, etc.

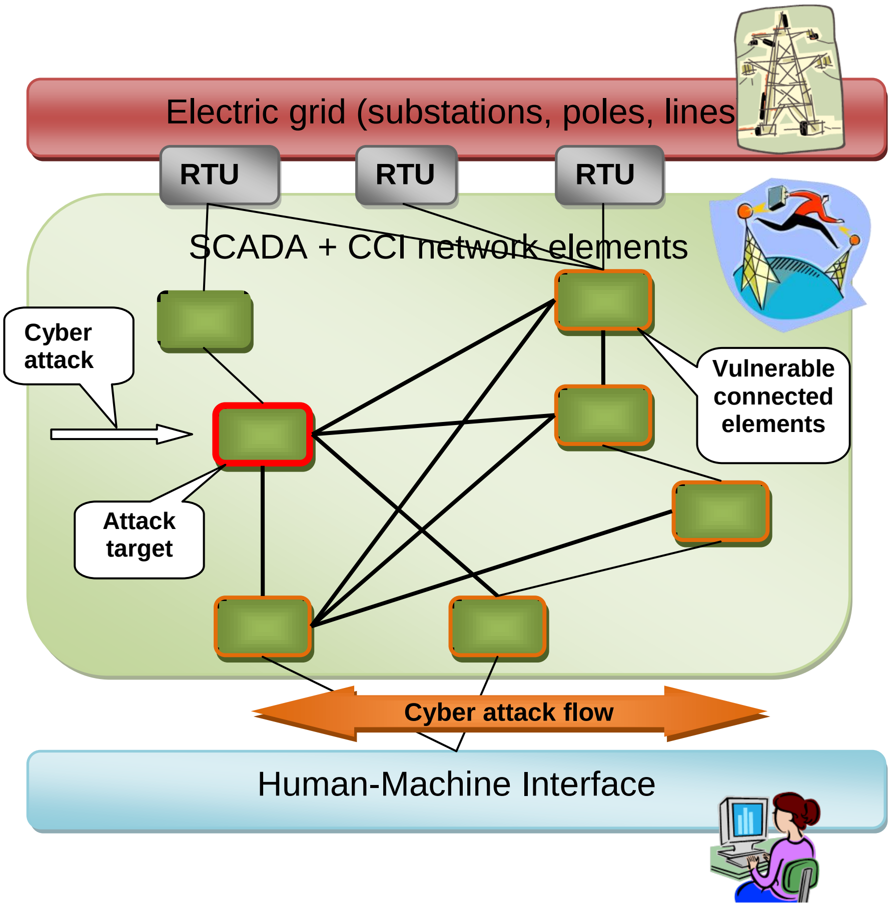

Electric grid (substations, poles, lines
RTU
RTU
RTU
SCADA + CCI network elements
Cyber attack
Vulnerable connected elements
Attack target
Cyber attack flow
Human-Machine Interface

Figure 17: Cyber attack spreading

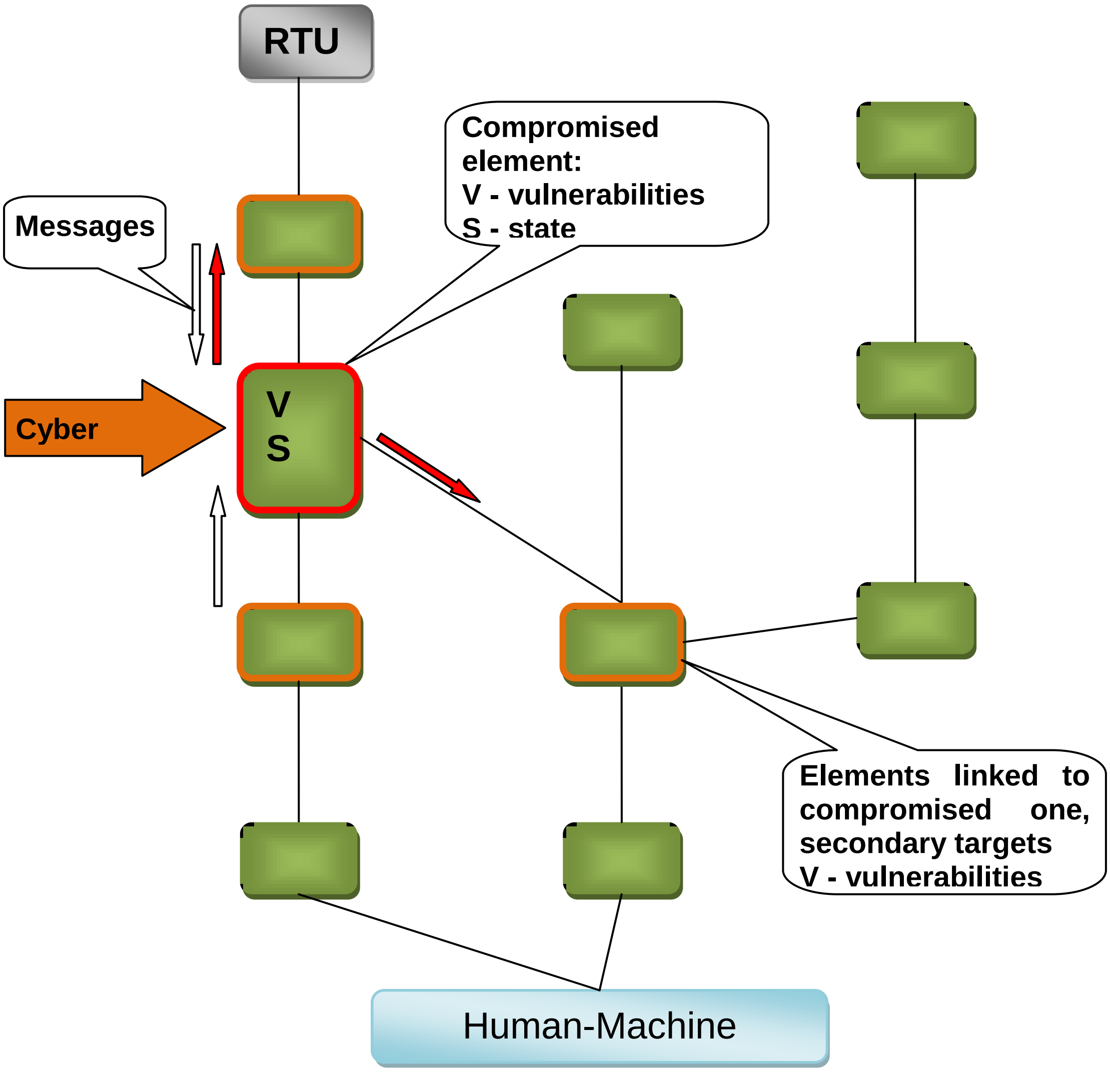


Figure 18: Cyber attack - vulnerabilities - state change - consequences

4.4 Attack altering elements states

Zooming on an element, we can model the compromising process and altered behaviour of the element. Cyber attack acts on some vulnerability of the element with respect to this type of attack. If the element is vulnerable to a given attack type, it becomes compromised (may be with some probability or after some time) and its state changes following a set of rules describing element state changes under cyber attacks of different types. A compromised element can either become active in spreading the attack (secondary cyber attack), or change its behaviour with respect to passing messages, or both. This depends on type of element and type of attack.

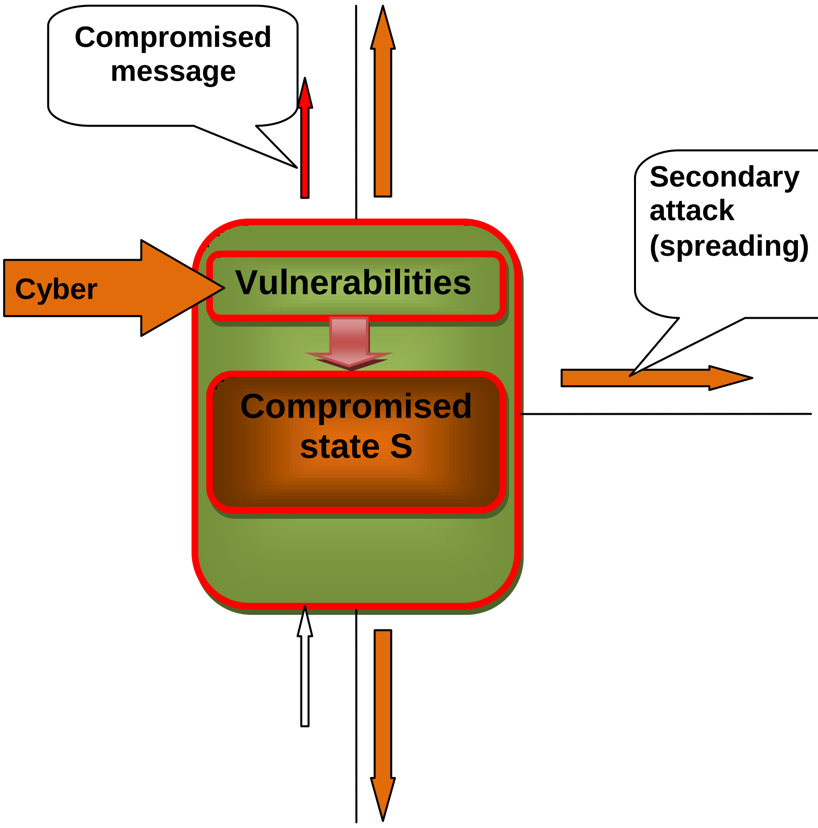


Figure 19: Compromised element behaviour

As an example, the final attack target could be an RTU, but primary target - a FIU. The contaminated FIU doesn't change passing messages with SCADA payload, but spreads the attack to RTUs. Once an RTU is compromised, it starts to alter messages with SCADA payload, but does not spread the attack further.

Depending on the attack type, state variables describing element vulnerabilities and further attack spreading parameters should be introduced. Some examples are given below.

- For malware spreading
  - state variables reflecting malware presence in CCI and SCADA elements
  - state variables reflecting presence and type of anti-malware
  - malware spreading links and probabilities (all physical links between elements, not only links belonging to paths and used in normal functioning)
  - entrance point for malware
- For Denial of Service (DoS) attack
  - state variables describing attack (packet size, packets number, interval between transmission, etc.)
  - packet buffers size and processing rate
  - all possible attack targets for a given element
- For Man In The Middle (MITM) attack
  - vulnerability to this type of attack

- all possible attack targets for a given element

The model includes also the logic linking CCI and SCADA elements state under current stage of complex cyber attack propagation (malware presence, number of packets during DoS attack, MITM manipulation logic, etc.) with changes in elements behavior regarding packets transmission.

4.1 Attack spreading over the system

Being compromised, an element can become a secondary source of cyber attack. Depending on element state and its logic of malfunction, it can attack connected elements, successfully or not. The process repeats itself with new compromised elements, thus representing attack spreading over the whole system.

Note that this representation allows one to describe a complex cyber attack. For example, attack starts with malware spreading over the system, then at some time compromised elements start a DDoS attack to a target which is not vulnerable to malware spreading. Another example, malware spreading can open at some level a way to MITM attack.

4.2 Messages alteration by compromised elements

For compromised behavior of CCI and SCADA elements under cyber attack, the following types of consequences (compromised behavior with respect to passing messages with SCADA payload) are introduced in the framework (can be extended by modeller if needed):

- messages pass (otherwise they are lost) unchanged with given probability (unstable service) and, when they pass, messages are delayed to some extent
- messages are manipulated (changed) with given probability in different way with given probability of each possible manipulation (changing payload, keeping it readable or not)
- messages are rerouted to wrong paths (with given probability)
- messages spontaneously issued (under secondary DoS attack, for instance) with given probability of each possible type/destination of messages issued
- a mix of changes above, for instance messages can be randomly lost, messages not lost can experience time delay and/or can be manipulated with given probability in different ways

To model such a compromised behavior with regards to messages processing and transmission, new parameters (state variables) CCI and SCADA elements have dedicated state variables like time delay, lost probability, probability of manipulation, probability of manipulation in a given way. Logic of packets processing, manipulation, transmission, taking into account malicious behavior, should be described in the model.

4.3 Physical and cyber objects to be modeled

In our framework the modelling requires description of following physical or cyber objects:

- ECI modelling framework from MICIE project [35]

- SCADA and CCI elements by type
- links between elements (network)
- messages with SCADA payload by type
- paths, possibly by message type, between all senders and destinations
- cyber attack types and targets
- vulnerabilities of elements, depending on type
- state (state variables) describing altered behaviour of element, attack propagation parameters and messages altering parameters, depending on type
- messages altering types and logic
- element behaviour logic if compromised with respect to further attack spreading and messages with SCADA payload altering, depending on type and state

Logic of elements behaviour can be described in form of production rules, state diagrams, algorithms etc.

5 Cyber Attacks

5.1 Kinds of cyber attacks

In reference scenario [67], three kinds of cyber attacks are considered:

1. Malware spreading;
2. Denial of Service (DoS);
3. Man in the Middle (MITM).

In the following, each attack, distinguished according to the three above typologies, will be specified in terms of characteristics, attack initiation sources, attack targets and expected consequences and referred to the topology and main devices of SCADA and corporate network of reference scenario [67], as reported in Figure 20 for convenience. Particularly, the criteria adopted in specifying the initiation sources and the targets, are:

- Attack initiation sources are chosen following the criteria of fully covering all the kind of devices involved in controlling power grid: SCADA devices, corporate network devices and, by means of internet connection, possible devices from outside.
- Attack targets are chosen following the criteria of obtaining a maximum number of damaged SCADA devices as a consequence of a successful attack on a single device. According to such a criteria, one chosen target will be the “MOSCAD front end”. “MOSCAD front end” is located before the wireless communication links to the remote terminal units, and its outage has immediate consequences on the Loss of Control (LoC) and the Loss of View (LoV) of all the Remote Terminal Units.

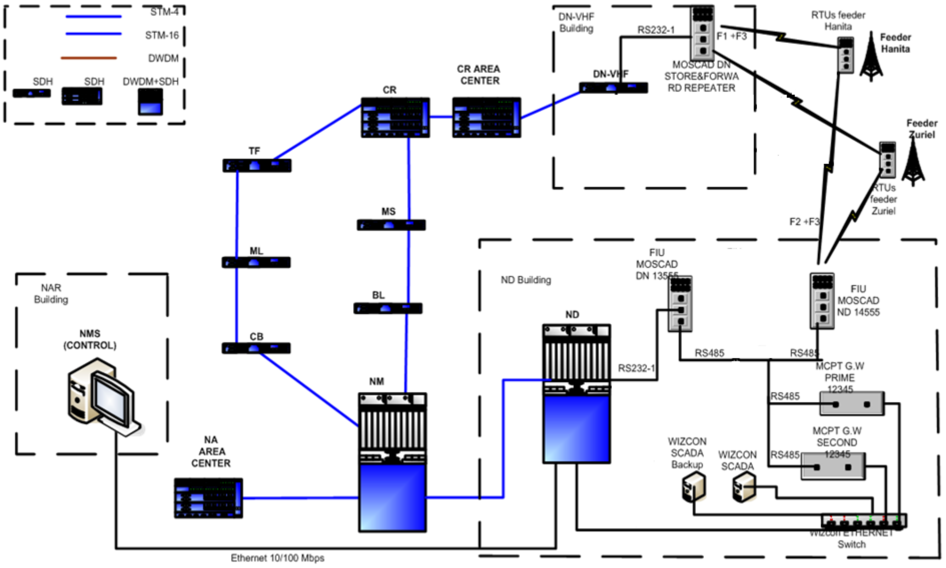


Figure 20: Main devices of SCADA and corporate network

On occurrence of any of above three typologies of cyber attacks (Malware spreading, DoS or MITM), four different phases of SCADA and corporate network behavior will be distinguished and analyzed along time:

1. normal conditions, before the attack;
2. anomalous conditions, during the attack;
3. a possible tail of anomalous conditions, after the attack;
4. return to normal conditions, as before the attack.

The impact of the attacks on the Power Grid could be then evaluated by means of engineering judgement or even by using a Power Grid simulator.

5.2 Expected consequences

A cyber attack could cause possible alterations on the configuration of SCADA and some of these could be the following:

- Making unauthorized changes to programmed instruction in local processors to take control of master-slave relationships between Master Terminal Units (MTUs) and RTUs.
- Modifying SCADA software or configuration settings.
- Infecting SCADA software with malware.
- Simultaneous failures of multiple systems.

A cyber attack could lead to miss-operation of SCADA operators:

- Attacker sent inaccurate/false information to system operators, either to disguise unauthorized changes or to cause the operator to initiate inappropriate actions.
- Unauthorized changing or disabling alarm thresholds.
- Blocking data or sending false information to operators to prevent them from being aware of conditions or to initiate inappropriate actions.
- Overtaxing staff resources due to simultaneous failures of multiple systems.

A cyber attack could lead to possible damages on the electrical grid:

- Interfering with the operation of plant equipment, which can cause modification to safety settings.
- Blocking or delaying the flow of information through ICS networks, which could disrupt ICS operation.
- Simultaneous failures of multiple systems

5.3 Indicators

The consequence of any of the above cyber attacks on SCADA and corporate network could be the lack or alteration of observability and controllability of the electrical grid and in turn the impossibility to execute adequate commands from SCADA. The consequences of a cyber attack are on SCADA and in turn on the electrical grid then, we distinguish SCADA quality of service indicators and grid QoS indicators.

SCADA QoS indicators

- *Loss of View (LoV) per cent,* if the SCADA Control Centre can't receive packets from the RTUs:

$$LoV\% = \frac{drop_{RTUs \to HMI}}{sent_{RTUs \to HMI}} \times 100$$

$$\frac{drop_{RTUs \to HMI}}{RTU_{sent}} * 100$$

- *Loss of Control (LoC) per cent*, if the RTUs can't receive packets from the SCADA Control Centre:

$$LoC\ \% = \frac{drop_{HMI \to RTUs}}{sent_{HMI \to RTUs}} \times 100$$

Moreover, packets can be dropped under an attack. A possible indicator could be:

- Total Drop packets (T*DP) per cent*, which gives a global vision of how many packets are missing on the network:

$$TDP\% = \frac{total\ packets\ drop}{total\ packets\ sent} \times 100$$

$$\frac{total\ packets\ drop}{total\ packets\ sent} * 100$$

It is also expected a variation of:

- *Transmission Time between two packets*;
- *Round Trip Time (RTT)*, composed by *TCP transmission time* plus *ACK transmission time*
- *Packets routing*
- Time Response of SCADA in executing FISR procedure

Electrical grid QoS indicators

The consequences of cyber attacks on the electrical grid could be the degradation of reliability, resilience, safety and quality of electricity to customers, typically regulated by a National Electric Authority. QoS indicators:

- The duration of electrical interruptions for customer for year
- The number of long/short electrical interruptions for customer per year
- SAIDI - System Average Interruption Duration
- SAIFI - System Average Frequency Interruption
- CAIDI - Customer Average Interruption Duration
- Overvoltage values and duration dangerous levels - damages to equipment or to customers.

5.4 Malware spreading

We consider the occurrence of a cyber attack which injects malware at a corporate network device of the reference scenario: Network Management System (NMS) of Figure 21, and the infection spreading throughout SCADA and Corporate devices. That with the aim to evaluate the possible disconnection of the communication between SCADA Control Centre and its RTUs and the consequent degradation of the quality of FISR service.

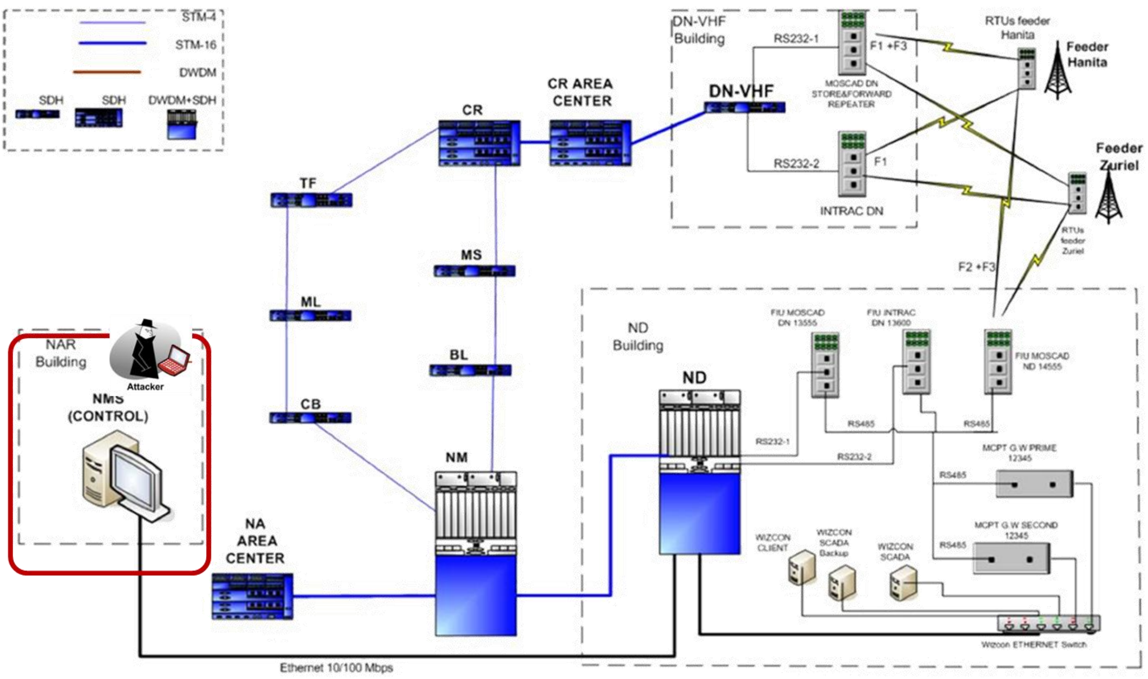


Figure 21: Malware injection on Corporate network device of reference scenario

We assume that the security polices of each device (or element, or node) of SCADA and corporate networks are dependent upon the criticality of such a device (or element or node). The rationale is that corporate devices with a larger bandwidth will be more protected and thus more expensive to be destroyed from a malware injected by an attacker. SCADA devices are not as critical as the devices of corporate network. Thus the latter will be more protected than the former. Accordingly, attacking the latter kind of nodes will be more expensive than attacking other less important (and thus less protected) nodes. As a consequence corporate devices are more vulnerable than SCADA devices because corporate devices are more “public”. The antivirus policy on corporate devices is more efficient that the antivirus policy on SCADA devices. Within corporate network, the antivirus policy of Point of Presence devices is more efficient than the antivirus policy of Transmission Exchange devices, in turn more efficient than antivirus policy of Local Exchange devices. Within SCADA system, the computers of SCADA Control Centre are more protected than the other SCADA devices.

To represent this attack and its consequences, the following security policies will be accounted:

- Antivirus-check: it indicates how many time units (days: 1- 365) occur between two consecutive antivirus checks (or everything that can help to find a malware).

- Virus-spread: the virus (malware) spreads along SCADA and Corporate network at certain rate: [1,365] day(s).

- Probability of infection from a device to its neighbor.

- Probability that the antivirus discovers the malware and cleans the device.

The virus propagates throughout SCADA and Corporate devices following its specific properties and accounting their security policies.

The **Target** is to get out of service the redundant (primary and secondary) connections between SCADA Control Centre and RTUs.

The malware is injected at the Network Management System device of the corporate network of the reference scenario (**Source**, Figure 21).

The **expected consequences** are the loss of Observability and Controllability of Power Distribution Grid from SCADA Control Centre.

In **Attack cases** list, *Case 0*, the malware is injected at Network Management System device and spreads on SCADA and corporate network devices.

5.5 Denial of Service (DoS) attacks

A Denial of Service (DoS) attack in which a malicious agent exploits the weakness of network protocols to flood a target node and exhaust its resources will be investigated.

The main **characteristics** of the DoS attack will be specified, in terms of packet size, interval between packet transmission, number of packets sent during the attack, the protocol followed by the flood attack.

A possible **attack target** will be the MOSCAD front end before the wireless communication links towards Remote Terminal Units. Particularly, the attack target will be the MOSCAD-DN when the attack is coming from corporate network and the SCADA is working on the alternate path (otherwise the attack has no effect). The attack target will be MOSCAD-ND when the attack is coming from an external devices connected to SCADA/corporate network throughout Internet.

According to the criteria specified in the previous sections, attacks can start from corporate network devices and from external devices connected to SCADA/corporate network throughout Internet (**Attack sources**).

When the attack starts from a device of the corporate network, where the bandwidth is higher than the ones of SCADA (RTUs and MOSCADs), the attack is expected to be more effective. The buffers of the SCADA devices (MOSCADs) saturate faster than the buffers of corporate network devices due to the different bandwidth of SCADA cables respect to the bandwidth of the Corporate network cables.

When the attack starts from Internet, it is expected that it causes less damage than the previous ones due to the lower bandwidth of the connection between the attacker and Internet and between Internet and the SCADA Ethernet bus.

In Figure 22, Jolly Rogers indicate the possible attack sources.

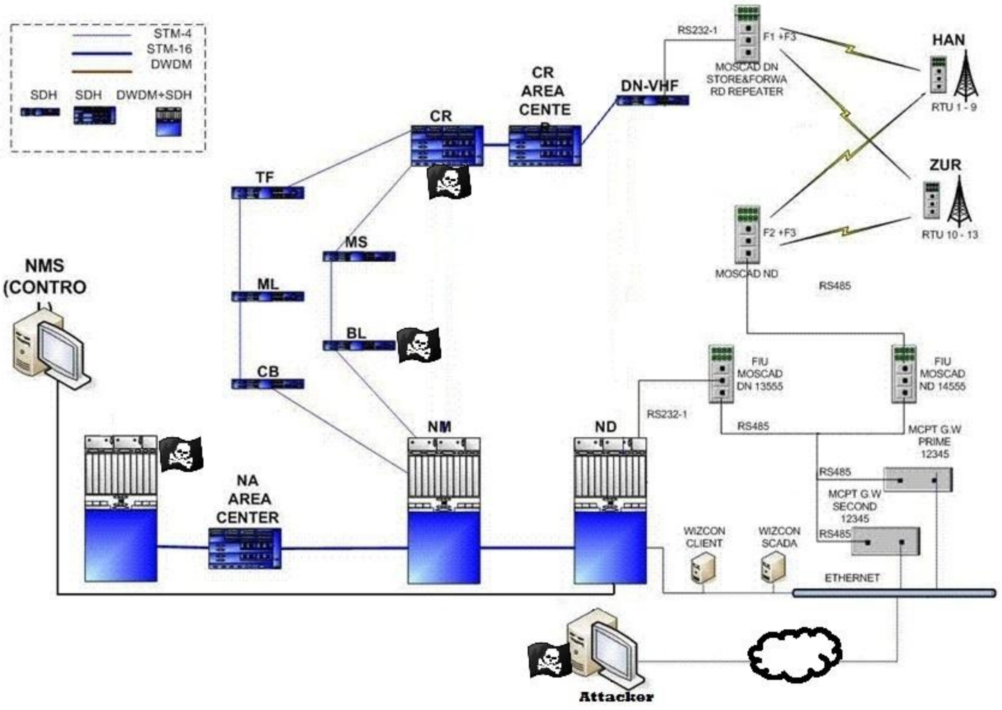


Figure 22: Reference scenario with possible DoS attack sources

The **expected consequences** are changes in value of SCADA and grid QoS indicators, as defined in subsection "Indicators".

The following sources of DoS **attack cases** are proposed:

- *Case 1*, DoS attack from the TeX-CR

- *Case 2*, DoS attack from the LeX-BL
- *Case 3*, DoS attack from the PoP
- *Case 4*, DoS attack from an external source

The Cases from 1 to 3 are attacks initiated inside Corporate Network.

Case 4 is a DoS attack from an external source. The possibility of a direct high-rate DoS flooding from a remote source (Case 4) is limited by factors such as the Maximum Transfer Unit (MTU) of the communications medium and encapsulating VPN technology, as well from the available bandwidth. Nonetheless, it remains a possibility that can and should be properly evaluated, even if they can be easily dealt with using rate-limiting techniques deployed in switch, routers and firewall devices.

5.6 Man in the Middle (MITM) attacks

In MITM attack, it is expected a change of normal packet routes, and packet routing could be considered one of the possible indicator of the attack. For such a reason, the devices of reference scenario [67] are numbered and a correspondence table between device name and number is also included as in Table 1.

Table 1: Correspondence between device names of SCADA/corporate network and device numbers

Device Name	Device Number
FIU-DN	0
FIU-ND	1
MOSCAD-DN	2
MOSCAD-ND	3
RTU-HAN-1	4
RTU-HAN-2	5
RTU-HAN-3	6
RTU-HAN-4	7
RTU-HAN-5	8
RTU-HAN-6	9
RTU-HAN-7	10
RTU-HAN-8	11
RTU-HAN-9	12
RTU-ZUR-10	13
RTU-ZUR-11	14
RTU-ZUR-12	15
RTU-ZUR-13	16
TeX-CR-AREA-CENTRE	17
TeX-CR	18
TeX-NA-AREA-CENTRE	19
LeX-TF	20
LeX-MS	21
LeX-ML	22
LeX-BL	23

LeX-CB	24
LeX-DN-VHF	25
WIZCON CLIENT	26
WIZCON SCADA	27
NMS CONTRO	28
GATEWAY PRIME	29
GATEWAY SECOND	30
PoP	31
PoP-NM	32
PoP-ND	33
BUS Ethernet	34
INTERNET	35
INTERNET	36
MITM	38

The main **characteristics** of the MITM attack are:

- The attacker intercepts the traffic;
- Once the traffic is intercepted, the attacker injects new commands/information that override the original ones. The injection occurs by means of packets, in the same format of the normal SCADA packets, but with an higher frequency than the frequency of the normal SCADA packets between the SCADA Control Centre and the RTU victim. That is because it is believed that the higher frequency of the MITM packets facilitates the override of normal SCADA packets;
- The attacker doesn't modify the payload of normal SCADA packets;
- The attacker connects to SCADA devices or corporate network devices through a Ethernet cable at the same speed of the Ethernet of the reference scenario;
- When the attacker intercepts the VHF communication, (s)he uses a VHF antenna, the propagation time between MOSCAD and MITM and from MITM and RTU is halved.

A possible **attack target** will be the MOSCAD front end before the wireless communication links towards Remote Terminal Units. Particularly, the attack target will be MOSCAD-DN when the attack is coming from corporate network and the SCADA is working on the alternate path (otherwise the attack has no effect). The attack target will be MOSCAD-ND when the attack is coming from an external devices connected to SCADA/corporate network throughout Internet.

According to the criteria specified in the previous sections, an attack could start from a corporate network device and from an external devices connected to SCADA/corporate network throughout Internet (**attack sources**). The following sources of MITM attacks are proposed:

- Between Ethernet bus and the two gateways, in the SCADA Control Centre;
- Between TeX-CR and TeX-CR Area Centre;
- Between MOSCAD-DN and RTU;

- Throughout Internet (e.g. via VPN).

The **expected consequences** are on the following possible numeric indicators of MITM attack:

- *LoV* (Loss of View). In this case SCADA Control Centre receives false information/data from MITM attacker. The consequent false Observability of Power grid status from SCADA Control Centre may induce in tricky behaviour of SCADA operator on managing grid operations. $\frac{drop_{RTUs\rightarrow HMI}}{RTU_{sent}} * 100$
- *LoC* (Loss of Control). In this case receives false commands from MITM attacker instead of SCADA Control Centre. The consequences of such false commands on the Power Grid should be verified by engineering judgment or even by a Power Grid simulator.
- *Change of Packets routing*

It could be also expected a light variation of :

- *Transmission Time between packets*
- *Packet Round Trip Time (RTT),* composed by *TCP transmission time* plus *ACK transmission time*

The following cases of MITM attacks (**attack cases**) are proposed:

- *Case 6*, attacker between TeX-CR-AREA-Centre and TeX-CR;
- *Case 7*, attacker between BUS and GW Prime;
- *Case 8*, attacker between MOSCAD1 and RTU1;
- *Case 9*, attacker on Internet.

Specifically, the *Case 7* and *8* represent attacks initiated inside SCADA, while the *Case 6* represents an attack initiated inside Corporate Network.

Case 9 is a MITM Attacker on Internet. The use of Virtual Private Network (VPN) technologies reduces the possibility of an MITM attack coming from the inter-site links, since it protects the integrity and confidentiality such interconnects in such a way that it only becomes possible to compromise if inherent flaws are found in the supporting encryption protocols or negotiation mechanisms. Outside the scope of the proposed inter-LAN VPN scenario, an internet-initiated MITM attack is a possibility that can be avoided, to a higher extent, by adopting a combination of strict remote access policies, combined with encryption, AAA technologies and protocol and content filtering mechanisms.

6 Composing epidemic and performance models

In [64] a worst case cyber attack initiated on a corporate network device and that puts out of service the redundant (primary and secondary) connections between SCADA Control Centre and its remote devices is investigated combining NETLOGO and NS2 tools.

Along the different phases of the attack the Fault Isolation and System Restoration (FISR) service, performed by SCADA, has degraded time responses which affect the quality of power to grid customers. [64] discuss a model implemented by means of NETLOGO to represent the occurrence of the cyber attack targeted at a specific system of the corporate network, the Network Management System, which spreads the infection throughout SCADA and ICT nodes up to disconnect the communication between SCADA Control Centre and its remote devices, resulting in the SCADA QoS degradation.

Model parameters include the probability of infection of a node, the virus spread rate, the intrusion detection rate of corrupted SCADA/ICT servers or remote devices and keep into account of the potentiality of the attacks, the vulnerabilities and security policies of the single SCADA and ICT elements.

The infection spreading affects FISR service and in turn the quality of power to grid customers as computed by a QoS prediction model, implemented by NS2 simulator.

6.1 FISR service

FISR service is delivered by SCADA operators, according to procedures which may vary from one case to another one. To represent FISR we account the following procedure [65, 40]. Initially, MV power grid Figure 23 is in its operative conditions. Then, randomly, a permanent failure occurs on any electrical section of the sub grid fed by either CB or HF sub-stations. As a consequence the Protection breaker at the substation will trip. After two automatic reclosing attempts, the Protection breaker remains open, de-energizing all the sub grid. The loss of power is sensed by each RTUs, that using its backup battery, opens the correspondent Circuit breaker. At this point, the "failure detection process” starts, by re-closing progressively all breakers, starting from the one closest to the electrical substation in order to detect the failed section. On the attempt of re-closure of the breaker at the head of the failed section, its RTU senses the loss of power and immediately re-opens the breaker and sends an alerting message to SCC, that acknowledges it. The re-opening of this breaker ensures that the failure is isolated. During failure detection and isolation process all the customers included in the portion of the sub grid between the head of the failed section and the N.O. (Normally Open) Tie switches remain de-energized. At this point the "power restoration process" starts, remotely performed by SCADA operator. The actual implementation of such a process depends upon the location of the failure inside the sub grid, and may include, if necessary, the closure of N.O. Tie switches (thanks to request/response messages between SCC and RTUs) in order that all customers, except those included in the failed section, can be energized by the other substation. After the repair of the failed section, the grid can be reported to its initial configuration.

6.2 A single heterogeneous network supporting FISR service

SCADA system, MV power grid and corporate network constitute a single heterogeneous network that supports FISR service, as shown in Figure 23. In Figure 23, a green box bounds the main SCADA devices (except RTUs and radio links), a yellow box bounds Telco devices that support a SCADA redundant link, and a red box bounds the Power grid. There are several interconnections among these networks. SCADA system interacts with MV Power grid by means of SCADA RTUs. Also SCADA devices, including RTUs and devices in Control Centres, are powered by the same MV Power grid. SCADA system also interacts with Telco network by means of the redundant link, which traverse PoP ND and LeX DN-VHF devices, and by the communication link between SCADA Control Centre and PoP ND.

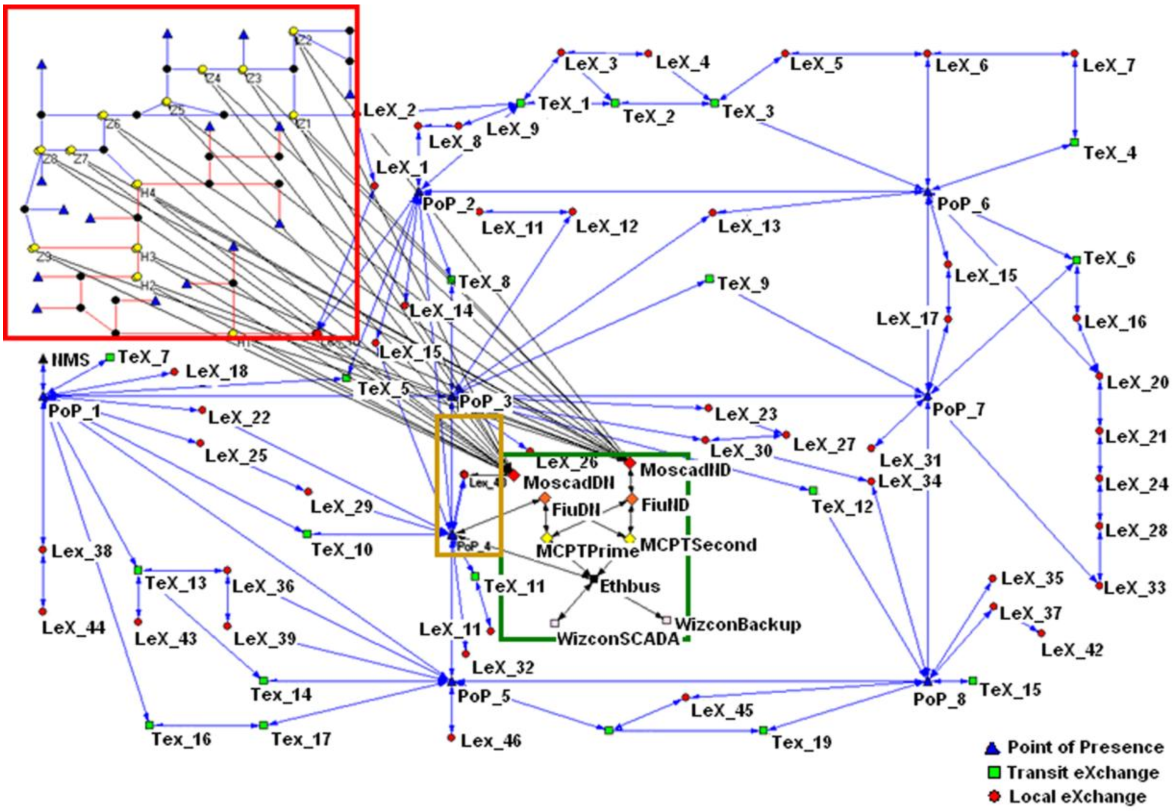


Figure 23: A single heterogeneous network supporting FISR service

6.3 Models

We represent SCADA and corporate network under the occurrence of three different kinds of cyber attacks:

1. Malware spreading
2. Denial of Service

3. Man In The Middle

6.3.1 Malware propagation

Malware propagation model is based on SIR (Susceptible, Infected, Resistant) mathematical formalism, for disease spread over individuals [27]. Each individual could be in one of the three states: Susceptible, Infected or Resistant. There isn't the possibility that an individual could belong to more than one of the states. The passage between each state is governed by several variables. To represent SCADA and corporate network we got a SIR net, described by a graph. We said that each device is a node, and there is an arc if two nodes can communicate each other (the arcs are symmetric). The virus infection of the original SIR formalism, in our case is the malware. A node can move from S, the Susceptible group, to I, the Infected group, when it comes in contact with an infected node. What qualifies a contact depends on the virus. Each infected node contacts the neighbour nodes in each step of time. Each contact may not result in transmission of the virus, only a percent of the contacts result in transmission.

For each j node (j=1,…,N), we define d_j as the number of the neighbours of the node j of which the fraction α may result infected; so, we assume that the virus spread itself, every step of time, on a fraction $\beta_j = \alpha \bullet d_j$ of nodes. We justify such an assumption thinking to deal with a stealth virus. A stealth virus doesn't infect too much nodes every time, because otherwise, it could be more easily detected for instance looking at the increased traffic value. Moreover, we assume that each node has different probability to contract the virus: γ_j. The virus doesn't disappear after a certain period of time, but just after periodically running the antivirus or after maintenance operation, k_j is the rate of the antivirus scan. Depending on the virus, there is the possibility that the antivirus can find it and know how to remove it, φ_j is that probability. At each point of time, we have three groups of nodes and a specific numbers of nodes in a group. Particularly, S(t), I(t) and R(t) are, respectively, the number of Susceptible, Infected and Recovered nodes in the network at time t. Given N, the network size, correspondingly, we define the three groups as fractions of the total population N in lower case:

- s(t) = S(t)/N (the susceptible fraction of the nodes of the network at time t)
- i(t) = I(t)/N (the infected fraction of the nodes of the network at time t)
- r(t) = R(t)/N (the recovered fraction of the nodes of the network at time t)

Each node is in one of the three groups. Thus:

$$S(t) + I(t) + R(t) = N \quad (1)$$

and

$$s(t) + i(t) + r(t) = 1 \quad (2)$$

At the time (t + 1):

$$S(t+1)= S(t) - s \bullet \beta \bullet \gamma \bullet I(t) \quad (3)$$

$$R(t+1)= R(t) + k \bullet \varphi \bullet I(t) \quad (4)$$

$$I(t+1)= I(t)+s\bullet\beta \bullet \gamma \bullet I(t) - k \bullet \varphi \bullet I(t) \quad (5)$$

We have used NetLogo to create SCADA & corporate network model, to set SIR variables and to represent the occurrence of a cyber attack on a corporate network device (Network Management System). NetLogo is an agent-based modelling tool for simulating natural and social phenomena. It is particularly well suited for modelling complex systems developing over time. In our model, malware spreads throughout the corporate network and SCADA devices up to disconnect the communication between SCADA Control Centre and RTUs. We assume that the security polices of SCADA and corporate network are dependent upon the criticality of their devices. The rationale is that the corporate network devices with a larger bandwidth will be more protected and thus more expensive to be destroyed from an attacker. SCADA devices are not as critical as the ICT devices of corporate network. Thus, the latter will be more protected than the former. Accordingly, attacking the latter kind of nodes will be more expensive than attacking other less important (and thus less protected) nodes. On the other side, corporate network devices are more vulnerable than SCADA devices because corporate network devices are more “public”. The antivirus policy on corporate network devices is more efficient that the antivirus policy on SCADA devices. Within corporate network, the antivirus policy of Point of Presence devices is more efficient than the antivirus policy of Transmission Exchange devices, in turn more efficient than antivirus policy of Local Exchange devices. Within SCADA system, the computers of SCADA Control Centre are more protected than the other SCADA devices.

In SIR model of SCADA and corporate network, we use the following variables:

- Alfa: it is a measure of how many neighbours the virus sends the infection to. Its range is [0, 100] %.

- Antivirus-check: it indicates how many time units occur to perform an antivirus check (or everything that can help to find a malware). Its range is [1, 365] days.

- Virus-spread-timer: the virus (malware) can spread itself along the network at various rates. We assume that an infected node may infect just a fraction of its neighbours (an exception is the Wizcon Ethernet bus, that just transmits the infection). Its range is [1, 365] days.

Figure 24 shows the screenshot, at time t=0, of SIR model of SCADA and corporate network. The infection starts on Network Management System device of the corporate network, named HMI-NMS_CONTRO in Figure 24.

Along the infection spreading, each node of SIR model can be in one of the three states:

- Susceptible (S): the node is healthy (in green colour) and it can be infected by a malware;

- Infected (I): the node is infected (in red colour): at some rate it can infect neighbour nodes;

- Recovered (R): the antivirus scan got success in removing the infection (in gray colour).

The links among corporate network nodes are depicted in red colour while the links among SCADA nodes are depicted in blue colour. Ticks, up on the left of Figure 24, shows the simulation time.

Another output of the simulation (not included in Figure 24) is the percentage of nodes in the different states.

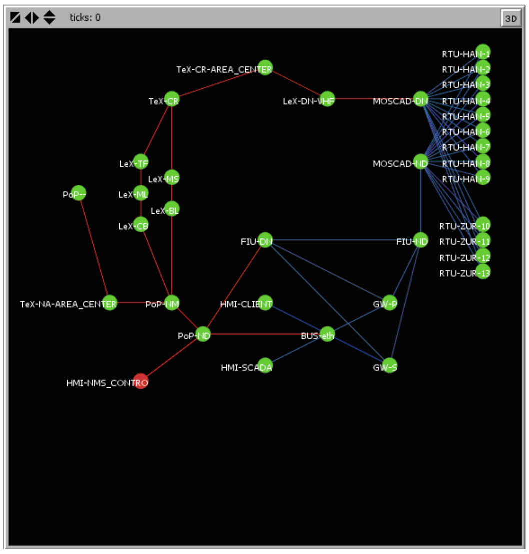


Figure 24: The infection starts on NMS device of corporate network

According to the modelling assumptions on the infection spreading, the virus propagates throughout PoP-ND and PoP-NM devices (respectively at time step=1 and at time step=2) and in turn on the GW-P device (at time step =4) of the primary SCADA Control Centre-RTUs connection, Figure 25.

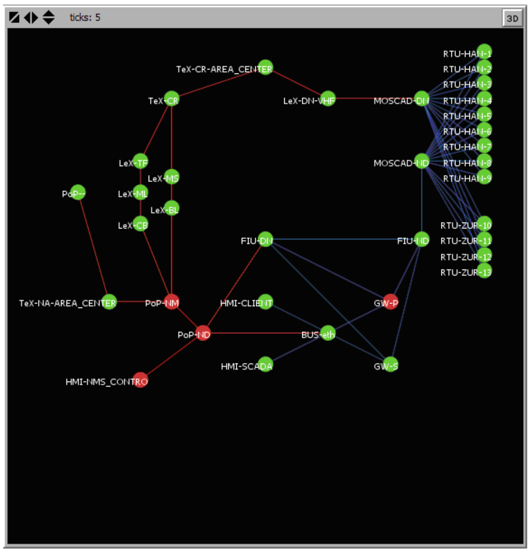


Figure 25: The infection spreads on corporate network and SCADA devices

The virus does not spread throughout the redundant computers of SCADA Control Centre due to their strict cyber security policy. Then the virus spreads on LeX-CB and FIU-ND (time step= 5). The infection of FIU-ND node gets out of service the primary connection between SCADA Control Centre and Remote Terminal Units. The strict antivirus policy on the PoPs of the ICT network discovers and cleans the malware (time step=11 and 22 respectively) on PoP-NM and PoP-ND respectively. At such a stage, the SCADA operator, has still a full observability and operability of the electrical grid, by means of the secondary communication between SCADA Control Centre and RTUs.

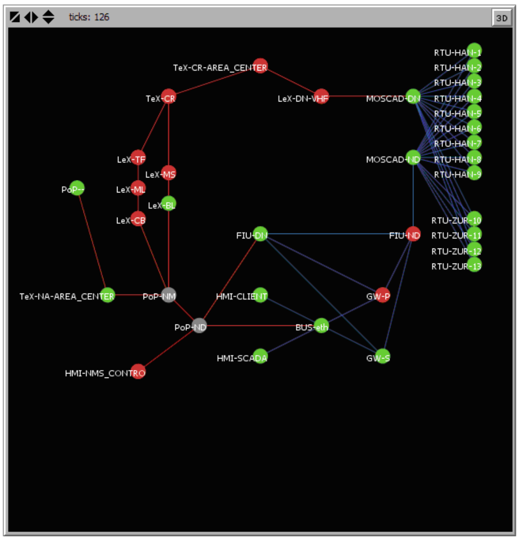


Figure 26: SCADA operator looses grid Observability.

At the time step = 52, the virus also infects TeX-CR node. At this stage (Figure 26), SCADA operator completely looses the observability and operability of the electrical grid. If a permanent electrical failure occurs on the grid, SCADA operator cannot act remotely the Fault Isolation and System Restoration Service.

More details of SIR model of SCADA & corporate network are in [62].

6.3.2 DoS attacks

DoS attacks will be performed with the aim to saturate the bandwidth of the carrier used for the communication between SCC and its RTU. Specifically, the SCC polls the RTUs from RTU-1 to RTU-13 and as a consequence it is expected that the transmission duration time from SCADA to RTUs increase from the first RTU to the last one. The ACKs transmission duration time is the same for all the RTUs. The MOSCAD front end of SCADA is under consideration to be chosen, as attack target, according to the criteria of causing a maximum number of damaged SCADA devices as a consequence of a successful attack. In fact, MOSCAD front end outage has immediate consequences on the Loss of Control and on the Loss of View of all the RTUs. Particularly, MOSCAD-DN will be chosen as attack target when the attack comes from the corporate network and the SCADA is working on the alternate path (otherwise the attack has no effects) and MOSCAD-ND when the attack comes from an external device connected to SCADA by means of Internet.

Four different attack initiation sources, named attack cases, will be chosen:

1. DoS attack from the TeX-CR;

2. DoS attack from the LeX-BL;

3. DoS attack from the PoP;

4. DoS attack from an external source.

The main parameters of the DoS attacks will be specified in terms of packet size, interval between packet transmission, number of packets sent during the attack, the protocol of the flood attack. Table 2 shows the value of such parameters we are going to use for all the DoS attack cases.

Table 2: DoS attack parameters.

Packet size	*10 B*
Interval	*10 µs*
N. of packets sent during the attack	*4 600 000 000*
Flood attack protocol	*UDP protocol with CBR*

6.3.3 MITM attacks

The main characteristics of the MITM attacks are the following:

- the attacker intercepts the traffic;

- once the traffic is intercepted, the attacker injects new commands/information that override the original ones. The injection occurs by means of packets between the SCADA Control Center and the RTU victim, with the same format of the normal SCADA packets, but with an higher frequency. The rationale is that a higher frequency of MITM packets facilitates the override of normal SCADA packets;

- the attacker doesn't modify the payload of normal SCADA packets;

- the attacker connects to SCADA devices or corporate network devices through a Ethernet cable at the same speed of the Ethernet of the reference scenario;

- when the attacker intercepts the VHF communication, he uses a VHF antenna, the propagation time between MOSCAD and MITM and from MITM and RTU is halved.

Also here, MOSCAD front end of SCADA will be chosen, as attack target. Particularly, MOSCAD-DN when the attack will come from corporate network and SCADA is working on the alternate path (otherwise the attack has no effects); MOSCAD-ND when the attack will come from an external devices connected to SCADA system by means of Internet.

The following sources of MITM attacks will be chosen:

1. Between TeX-CR and TeX-CR Area Center

2. Between Ethernet bus and the gateway, in the SCADA Control Center

3. Between MOSCAD-DN and RTU-HAN-2

4. Throughout Internet (e.g. via VPN).

6.3.4 Impact of cyber attacks on FISR service

Performance of FISR is computed on the single heterogeneous network of Figure 23, in which continuous and discrete parameters coexist. Power grid mainly consists of elements that are typically modelled by continuous equations. Physical laws that dictate the behaviour of electro mechanic elements of power grid are described by differential equations with some discrete dynamics needed to represent circuit breakers. Then, to represent power grids typically continuous simulators are used. On the way around, SCADA and corporate networks are packet switching networks, then they need to be represented by discrete event simulators. In our case, to compute performance of FISR, we need a full scale representation of SCADA and corporate network which act to implement FISR on the power grid and a proper representation of the power grid itself, as it is observable by the SCADA Control Centre, for the service under consideration. In general, the Power grid is observable by SCC in terms of bus voltages, line, generator & transformer flows (MW, MVAR & Amperes, transformer taps & breaker status as well as other generator parameters (e.g. limits), frequency, requiring a full scale simulator for its representation. The concept of observability of power grid from SCC, limited to FISR service, can be simplified, by just representing the topology of the grid (substations, trunks, loads, junctions, RTU breakers), and the events involved in FISR service (remote On/Off operation of RTU breakers from SCC, presence/absence of the electrical flow from the feeding Substations to loads, according to electrical Junctions and RTU breakers positions, and occurrence of possible failures in any electrical section of the grid). For such a limited representation of the Power grid, we may resort to a discrete event simulator. Among discrete event simulators we choose NS2, one of the most widely used open source network simulators [66]. NS2 allows to simulate packet based local/wide area networks and wired/wireless networks and then it may well represent SCADA and corporate networks. First, we built a separate NS2 script for SCADA system, corporate network and electrical grid, than we integrate them to have a whole FISR model that relies on the heterogeneous network shown in Figure 23.

6.3.5 Performance and routing of FISR Service

The quality of FISR service is critical because it is strictly correlated to the quality of power supplied to customers. There are different indicators of the quality of power supplied to customers, such as the duration of power interruptions for customer for year, the number of long/short power interruptions for customer per year, etc. Values of such indicators are typically regulated by a National Electric Authority. A timely actuation of FISR service, consequential to a permanent failure of the grid, reduces the outage duration and then contributes to keep indicators of quality of power supplied to customers within pre-fixed values. On the contrary, a delayed actuation of FISR service makes such indicators worse. We investigated s-t dynamical path and s-t Round Trip Time as basic indicators of FISR performance. S-t dynamical path is intended as the path of nodes traversed by a packet, from a source to a destination. It dynamically changes in consequence of network re-configuration caused by network congestion or link/node failures. It is computed between SCADA Control Centre and RTUs. S-t packet Round Trip Time (RTT) is intended as the packet transmission time, from a source to a destination plus the ACK time (from destination to source) for TCP-IP protocols. It is computed between SCADA Control Centre and RTUs. Then, we investigated FISR response time intended as the time between the occurrence of loss of power supply to customers (due to a grid failure) and the restoration of power supply to customers. We correlated it with the duration of grid outage and the percentage of customers affected by the outage.

6.4 Assumptions and input data

6.4.1 SCADA System

The communication between main SCADA Control Centre and RTUs is implemented by a request/response application protocol that relies on the TCP/IP transport layer protocol. The radio links between RTUs on one side and RF modem ND or RF modem DNS&F on the other side, were implemented as ideal wireless links by means of a no loss model (i.e. no shadowing, fading,…). Connections and IP traffic among Control Centre nodes and RTUs as well as among the nodes within the Control Centre were implemented with reference to [60]. Connections between the SCADA Control Centre and the RTUs were implemented by installing a TCP agent over each RTU, and a symmetrical TCP agent over each node representing the Control Centre. Then on each TCP agent we locate a CBR (Constant Bit Rate) traffic source that transmits a packet of 255 bytes length, conform to [60], with regular intervals of 30 sec, to simulate exchange of messages among RTUs and Control Centre nodes. The request/response mechanism between SCADA Control Centre and RTUs is implemented by means of the simulator's scheduler. Each Control Centre request to the RTU activates a CBR source on the RTUs as soon as the request is completely received. The TCP agent on the generic RTU will begin to transmit the response messages after a time interval, during which the RTU processes field data from the related electrical section of the grid. Table 3 summarizes the input data of SCADA communications links.

Table 3: Input data of SCADA communication links

Link Type	Ethernet	RS-485	RS-232	VHF-Radio
Capacity	100 Mbps	19.2 Kbps	19.2 Kbps	4.8 Kbps
s-t Node	SCADA - MCP_T- PoP	MCP_T-FIU FIU- RF modem	RF modem - Telco Nodes	RF modem - RTU
Traffic type	DLC (TCP) + TCP	DLC (TCP)	DLC (TCP)	DLC (TCP)
Traffic bit rate	256 bytes /30 sec	256 bytes /30 sec	256 bytes /30sec	256 byte /30 sec

Occurrence of failures on electrical sections of the Power grid are detected and transmitted by RTUs and Substations to SCADA Control Centre. SCADA sub model represents the main path and the backup path between SCADA CC and RTUs. In case of failure of the main SCADA unit, the backup SCADA unit is enabled. In case of failure of the main FIU and/or of the main Gateway, data are re-routed on a backup path. Queue types and buffer sizes (the maximum number of packets can be stored before dropping) of links are defined as for corporate network sub model.

6.4.2 Corporate Network

To generate a realistic traffic over the corporate network, we consider that it hosts reliable traffic (i.e. for real time control devices and equipments, including SCADA) and less reliable traffic (i.e. corporate traffic). We assume traffic generation according to two types of transport layer protocols of the IP (Internet Protocol) family, TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) that provide network services for applications and application layer protocols packet delivery. UDP is a connectionless, unreliable message delivery protocol. The routing policy of the network nodes is a DV (Distance Vector) type. The queues of each link are Drop-Tail, which implements FCFS (First Come First Served) scheduling and drop-on-overflow buffer management. Table 4 summarizes the input data of corporate network.

Table 4: Input data of corporate network

Link Type	**Backbone (DWDM)**	**TeX (STM-16)**	**LeX (STM-4)**
Capacity	10 Gbps	2.5 Gbps	600 Mbps
s-t Node	PoP-PoP	PoP-TeX, TeX-TeX	PoP-LeX, TeX-LeX, LeX-LeX
Traffic Type	TCP+UDP	TCP	TCP

Traffic Bit-Rate	12GB(TCP)
	+8GB(UDP)
		12 GB	12 GB

Type of Agents	CBR for UDP		FTP for TCP
N of Agents	100 for UDP		100 for TCP

6.4.3 Electrical grid

NS2 sub model of the electrical grid consists of 49 elements: 2 substations, 13 Circuit breakers driven by correspondent SCADA RTUs, 19 junction nodes and 15 loads. In normal operation, the grid is separated into two sub-grids by means of N.O. Tie switches. One sub grid, energized by TF substation, feeds 6 loads, while the other sub grid, energized by CB substation, feeds 9 loads. NS2 sub model of the MV power grid is interconnected with NS2 sub model of the SCADA system by means of Circuit breakers/RTUs. The status of Protection breakers at substations TF and CB is monitored by the SCADA Control Centre that can also actuate their remote re-closure. To build a worthwhile NS2 sub model of the grid, a careful attention has been paid to translate events and object of the electrical domain into adequate discrete modelling assumptions. The electrical current flowing from substations to loads is represented by CBR (Constant Bit Rate) packets trans-mitted from substation to loads by means of a static (source) routing protocol and an UDP transport layer protocol. UDP packets are sent almost continuously by NS2 node representing the substation to NS2 nodes representing loads. On/off operations on the N. C. Circuit breakers, remotely driven by SCADA sub model, are represented by the occurrence of up/down events on the link that represent the electrical section posed immediately after (according to the direction of the electrical current in nominal conditions) the breaker. Along FISR implementation procedure, the direction of the electrical current on the grid may change changes upon operations on N. C. Circuit breakers or on N. O. Tie switches. Then the static routing policy of the NS2 model of the grid changes accordingly.

6.5 Expected numerical results

To compute SCADA performances and in turn the quality of electrical power to customers, as consequences of each cyber attack, we have used NS2 simulator to build, run and predict related indicators under attacks. NS2 is one of the most widely used open source network simulators; it is driven by discrete events and allows to simulate packet based local/wide area networks and wired/wireless networks as well. The NS2 model of the single heterogeneous network (SCADA & corporate network & Electrical grid) supporting FISR service is under implementation considering cyber attacks as specified above. Communications between SCC and its RTUs are under implementation with reference to (IEC 60870-5), as well as the packet traffic on the network. IEC 60870-5 is a standard developed in a hierarchical manner and published in a number of sub-paths which completely define an open protocol for SCADA communications. The protocol is defined in terms of the Open Systems Interconnection (OSI) model, using a minimum sub-set of the layers: physical, data link, and application layers. Detailed definition of message structure at the data link level, and a set of application level data structures are included to develop interoperable systems. We are investigating how cyber threats, vulnerabilities and attacks might change the performances of SCADA and corporate network devices, as described in reference scenario [67] which, in turn, might lead to outages of the electrical grid. We are representing SCADA and corporate network under malware propagation, Denial of Service and Man In The Middle cyber attacks, by using the heterogeneous NetLogo and NS2 tools to predict the consequent values of performance indicators along the different attack phases.

6.6 Simulated DoS attacks

Table 5 summarizes the main parameters of simulated DoS attacks on SCADA system and their impact on SCADA performance.

Table 5: Simulated DoS attacks on SCADA system

Attack source	PoP	TeX-CR	LeX-BL	Internet
Attack target	Moscad DN	Moscad DN	Moscad DN	Moscad ND
Start time (sec)	55	55	55	55
Stop time (sec)	101	101	101	101
LoV	NA	NA	NA	0/17
LoC	57/57	57/57	57/57	59/93
RTT Max/Min (sec)	Inf / inf	Inf / inf	Inf / inf	Inf /1792
DPR	57/57	57/57	57/57	59/93
Simulation time (sec)	200	200	200	200
Computation time (min)	21	15	17	15

Particularly, the first four rows of the table specify the attack parameters: attack source, attack target, start and stop attack time. The following four rows report the attacks consequences, while the last two rows report the simulation and computation time, respectively. The computation time grows from 15 minutes to 21 minutes. This is due to the source of the attack; the more hops a communication involves, the longer is the time needed to complete the communication between devices; also if the packets are dropped near the source of the attack, such packets no longer need to be transmitted.

6.6.1 DoS attack from the TeX-CR and LeX-BL

In Figure 27 and Figure 28, two examples of the results of NS-2 simulation are shown. They represent the packets exchange between the SCADA Control Centre and the RTU-1, under DoS attack coming from TeX-CR and LeX-BL, respectively. The messages exchanged between the SCADA and RTU-1 are distinguished by five colors:

1. Black represents commands from the SCC to the RTU-1;

2. Blue: Acks from the RTU-1 to the SCC;
3. Red: the start time of the flood attack (55s);
4. Green: the end time of the flood attack (101s);
5. Magenta: RTT of the exchanged packets.

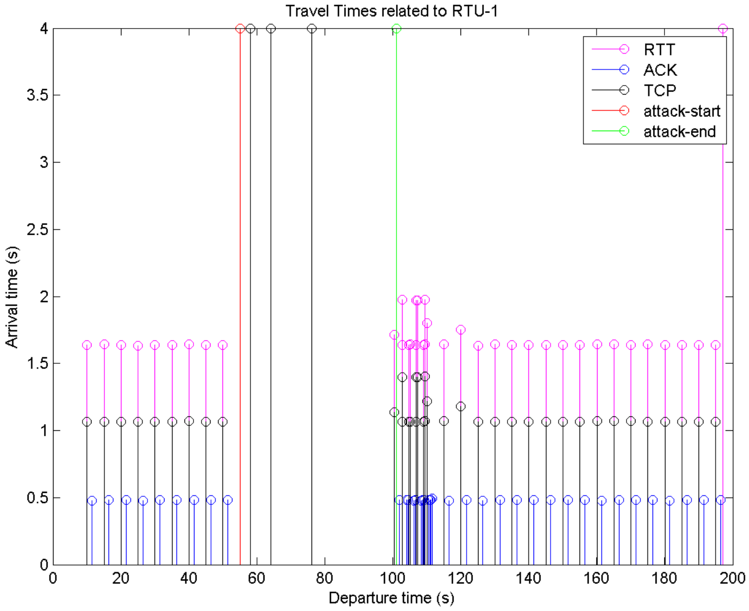


Figure 27: Travel Times of SCADA packets on RTU-1 when an attack comes from TeX-CR

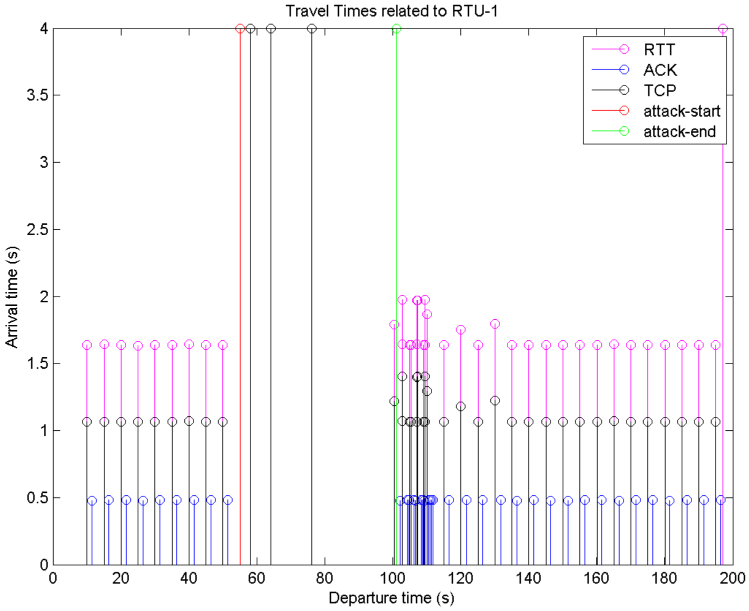


Figure 28: Travel Times of SCADA packets on RTU-1 when an attack comes from LeX-BL

In Figure 27 and Figure 28, we can distinguish four attack phases:

1. Before the attack: SCADA packets flow, without problem, from the SCC to the RTU-1 and come back normally. RTT (computed time) is the sum of the TCP travel time (measured time) plus the ACK travel time (measured time).
2. During the attack: the flood starts to increase the occupancy of all the buffers of the devices flooded by attack, up to saturate them (buffer size: 10 packets). If the SCADA packets flow to those devices, the time to reach the RTU is increased. The travel time of the Ack does not change because the links are full-duplex and the attack floods in the opposite direction. When a packet is dropped, TCP message interval time increases (i.e. more or less the double), increasing in turn the RTT.
3. A tail after the attack: SCADA messages go in de-synchronization. That is due to the fact that the saturated queue is emptied at a rate that is different from the nominal packet transmission rate, the packets are transmitted at lower intervals. Such intervals depend upon the elaboration time of each device. The time to reach the destination is unpredictable due to the fact that the buffers start to empty and there are still some flood packets that have to be sent.
4. Return to normal condition: flood problems end and the operative conditions come back to normal ones.

6.6.2 DoS attack from the PoP

Figure 29 shows the "travel times" of SCADA packets to the RTU-13, when the initiation source of a DoS attack is the PoP.

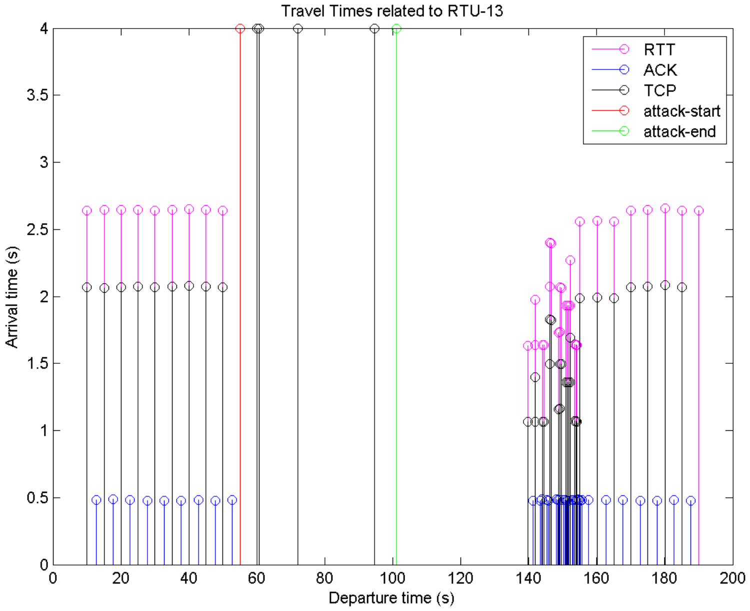


Figure 29: Travel Times of SCADA packets on RTU-12 when an attack comes from PoP

Also in this case, four attack phases are represented and the only difference, with the case of DoS attacks from TeX-CR or LeX-BL, is in the phase 2. In fact, in this case, just before the end of the attack, a packet reaches the designed RTU. At 60s of the simulation, two packets are sent (the bold line at 60s), the packet that reaches the RTU is one of the two, the other one still can't reach the RTU. When a packet is dropped, TCP message interval time increases (i.e. more or less the double). The first packet is related to the one scheduled at 55s, the second one is related to the one scheduled at 60s.

6.6.3 DoS attack from an external source

A study case in which the DoS attack source is an external source is represented in Figure 30 and, as in the previous figures, the four phases are shown.

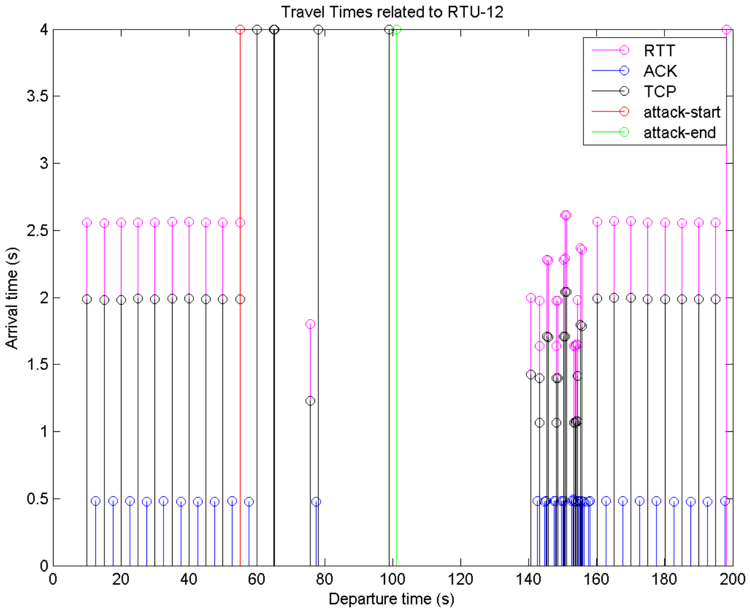


Figure 30: Travel Times of SCADA packets on RTU-11 when an attack comes from an external source

6.6.4 Lessons learned by SCADA system under DoS attacks

The lessons learned regard two major aspects:

1. The modification of RTT along the attack phases
2. The number of packets lost due to the attack.

As far as the RTT, here are the lessons learned:

- Attacks moving from devices with higher bandwidth than the SCADA will provoke major damage to the continuity of the service. This is due to the fact that the attack that comes from the internet is de-powered always by the low bandwidth while attacks that starts from the corporate network pass through decreasing bandwidth;
- One comment about phase 2 and phase 3. During phase 2, the transmission interval between packets grows according to the Tahoe (congestion control algorithm)

version of the implemented TCP; the value of growth depends upon the RTT value. The greater is the RTT value, the greater is the growth of the transmission interval between packets. During phase 3, instead, the packets are sent more frequently, then the transmission interval is reduced in such a way to retrieve that packets lost during the attack. This particular effect is due to the well known TCP mechanism, named AIMD (Additive Increase/Multiple Decrease). When a congestion occurs, AIMD works as follows: when there is a congestion (phase 2) the transmission frequency decreases in multiplicative way; when there is no congestion (phase 3), the transmission frequency grows in an additive way;

- One comment should be made about the phase 3 of the attack. During this phase the values of the RTT are different compared to the normal ones. This is caused by TCP retransmission mechanism. In fact in normal condition the transmission times of packets depend upon the scheduling mechanism that polls the RTUs from the first one (RTU-1) to the last one (RTU-13). After the congestion all the packets of all the RTUs that are still in the transmission queue are resent with no respect to the scheduling mechanism. The minimum value of the RTT in this phase, see RTU-13, is similar to the normal RTT of RTU-1. That is because packets of RTU-13 do not need to wait the other packets related to the other RTU. RTU-1 gets higher RTT because its packets can’t accelerate during this phase and they can only find more traffic on the transmission channel and that will bring more transmission delay.

As far as the packets lost intended as LoC and LoV here are the lesson learned:

- Impact: the worst case attack scenarios are the ones that bring a complete Loss of Control (LoC) of the SCADA (Table 5). This occurs when the attack starts from the corporate network, in this case the LoC is total (57/57), while if the attack comes from the external attacker from Internet, the LoC is partial (59/76). That is probably due to the fact that even with a massive attack flow is lost by different bandwidth bottlenecks: the first one is the Ethernet bus within SCADA Control Center, the second one is the path between gateways and Moscad FIU. We have a LoC and not a LoV because the direction of the attack is towards the RTUs and the communication links are full-duplex. If the attack is towards SCADA Control Center we got a LoV instead then the LoC in a similar way. This situation is in a normal polling condition of SCC to RTUs. On the contrary if we have an anomalous condition of Power Grid detected by RTUs, the RTUs send an interrupt to the SCC to alert it. If we have an attack towards RTUs, due to the direction of the flood (from the corporate devices to the MOSCAD) and the duplex-links, a possible interrupt from the RTU can reach the SCC without problems because it travels on the opposite direction.

6.7 Simulated MITM attack

For each simulation, in the following sub sections, we present the computational time and all the elements involved in the communication between SCC and RTU-2, along the four phases of the MITM attack. The reference number of each device of the two networks is shown in Table 6 where, on the first column there are the names of the devices and in the second column there are their related numbers.

Table 6: Reference number of the devices on the SCADA and corporate networks

Device Name	Related Number
FIU-DN	0
FIU-ND	1
MOSCAD-DN	2
MOSCAD-ND	3
RTU-HAN-1	4
RTU-HAN-2	5
RTU-HAN-3	6
RTU-HAN-4	7
RTU-HAN-5	8
RTU-HAN-6	9
RTU-HAN-7	10
RTU-HAN-8	11
RTU-HAN-9	12
RTU-ZUR-10	13
RTU-ZUR-11	14
RTU-ZUR-12	15
RTU-ZUR-13	16
TeX-CR-AREA-CENTER	17
TeX-CR	18
TeX-NA-AREA-CENTER	19
LeX-TF	20
LeX-MS	21
LeX-ML	22
LeX-BL	23
LeX-CB	24
LeX-DN-VHF	25
WIZCON CLIENT	26
WIZCON SCADA	27
NMS CONTRO	28
GATEWAY PRIME	29
GATEWAY SECOND	30
PoP	31
PoP-NM	32
PoP-ND	33
BUS Ethernet	34
INTERNET	35
INTERNET	36
MITM	38

For each row of the Tables, reported in the following, the first bullet shows the route taken by SCADA packets from the SCC (n. 27) to the RTU-HAN-2 (n. 5); the second bullet shows the route from RTU-HAN-2 (n.5) to the SCC. The MITM node (n. 38) is highlighted by the bold font and it is underlined.

The node that represents Internet got two numbers, because in NS-2 it was helpful while developing the attack from Internet. In such an attack, we had to trick the shortest path algorithm. To perform the trick, we had exploded Internet in two nodes in such a way as not to pass on the same link twice. The details of such a need are well shown in the following pictures: Figure 31 and Figure 32.
The black links are the physical cables; the red arrows are the route that a packet should take according to the attack. The direction of the arrows indicates a communication from the SCC to the RTUs. A similar argument can be done for the reverse communication.

As one can see in Figure 31, the routing algorithm should pass twice to the same links (bus-internet and internet-MITM). This solution is not allowed, so we have split the cloud Internet in two nodes, now we can modify the route to perform our attack.

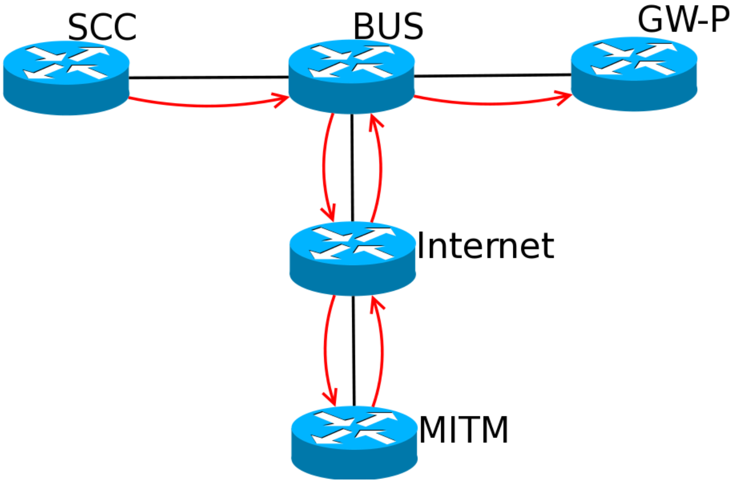


Figure 31: MITM with one Internet node

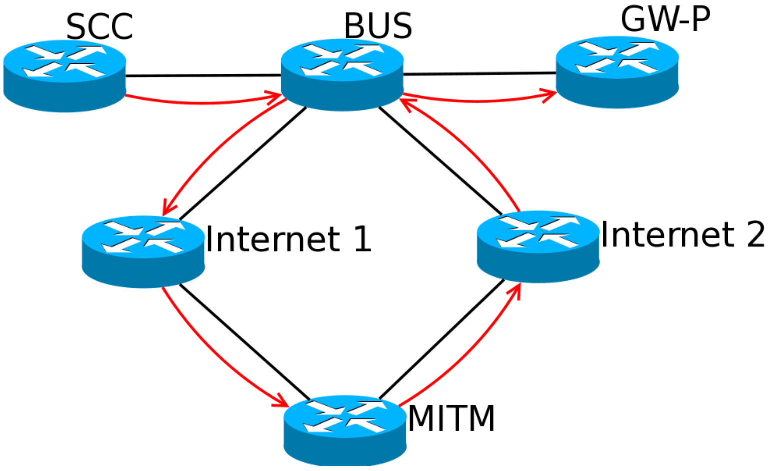


Figure 32: MITM with two Internet nodes

6.7.1 MITM attack between TeX-CR and TeX-CR Area Center

In the following Table 7 we present the obtained results in the case of MITM attack between TeX-CR and TeX-CR Area Center. The computational time is reported in the last row while the traversed devices before the attack, during the attack and after the attack are shown respectively in the other three rows.

Table 7: MITM attack between TeX-CR and TeX-CR Area Center

	Traversed devices
before attack	• 27 34 29 0 33 32 24 22 20 18 17 25 2 5 • 5 2 25 17 18 21 23 32 33 0 29 34 27
during attack	• 27 34 29 0 33 32 24 22 20 18 **38** 17 25 2 5 • 5 2 25 17 **38** 18 21 23 32 33 0 29 34 27
after attack	• 27 34 29 0 33 32 24 22 20 18 17 25 2 5 • 5 2 25 17 18 21 23 32 33 0 29 34 27
Computational time	**6 seconds**

We can notice, in the above Table 7, that the path taken to return to the SCC is different from the one taken to go to the RTU. This is caused by the routing algorithm.

In Figure 33 the travel times (RTT, ACK and TCP) of the communication from SCC to RTU-2 under a MITM attack between TeX-CR and TeX-CR-AREA-Center are shown.

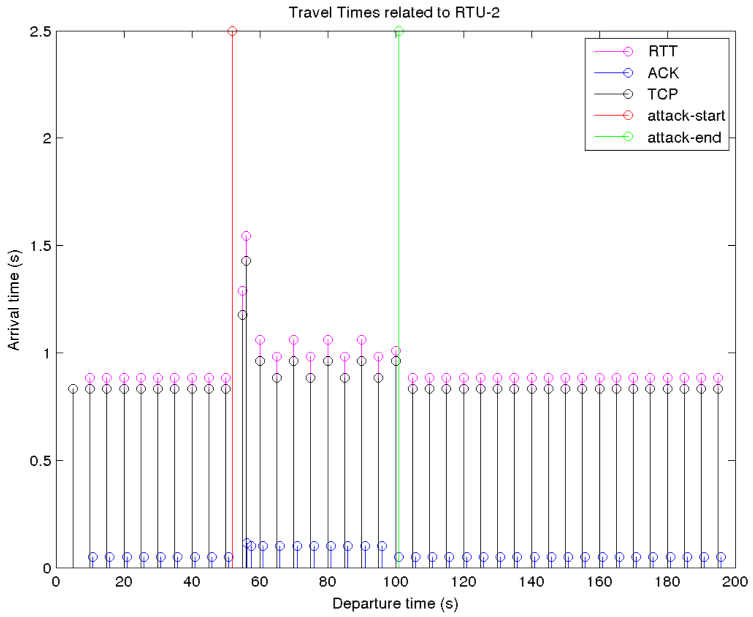


Figure 33: Arrival times (RTT, ACK and TCP) from SCC to RTU-2 with an attacker between TeX-CR and TeX-CR-AREA-Center

The attacker enters in the network with a cable with different delay values (the ones of Ethernet cable) respect to the delay values of optical fiber links. That causes a sensible delay that is measurable by means of RTT variation.

The increased RTTs near the start of the attack are caused by NS-2's TCP version. The protocol waits for the ACK for a certain time. The time is the one computed according to the previous communications. If the ACK is not received, and it is the case, the TCP protocol resends a packet with the same identifier assuming that the previous packet is lost. Instead the first packet was not lost but just delayed for the presence of the MITM node, then we have the transmission of two TCP packets with the same identifier and consequently two ACKs with the same identifier. TCP resends a packet thinking that the message went lost while it is simply delayed. When the ACKs are received, the protocol tunes the waiting time to receive the ACKs to a new value.

In the following Figure 34, the arrival times (RTT, ACK and TCP) of the packets sent by MITM node to RTU-2 are shown.

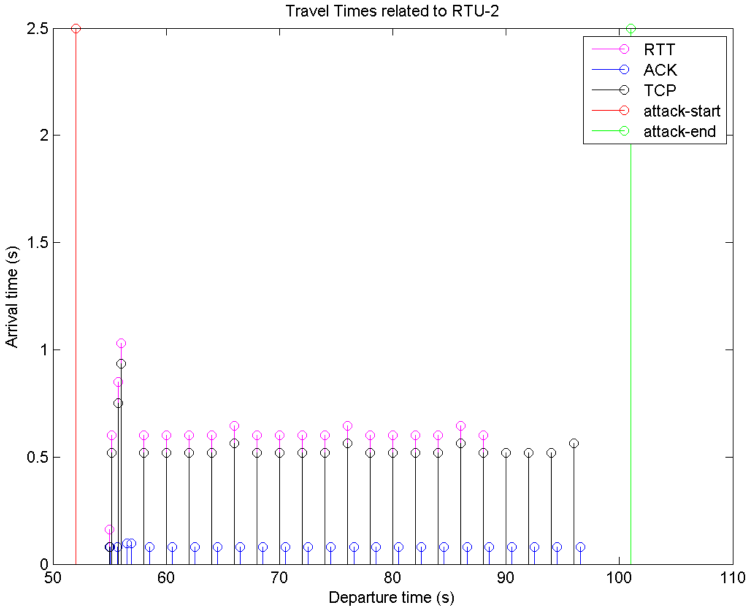


Figure 34: MITM packets to RTU-2

6.7.2 MITM attack between Ethernet bus and the gateway

Table 8 presents the obtained results in the case in which the attacker is between bus and gateway. In the last row the computational time is reported, while in the other three rows, the traversed devices before the attack, during the attack and after the attack are shown respectively.

Table 8: MITM attack between bus and gateway

	Traversed devices
before attack	• 27 34 29 1 3 5 • 5 3 1 29 34 27
during attack	• 27 34 **38** 29 1 3 5 • 5 3 1 29 **38** 34 27
after attack	• 27 34 29 1 3 5 • 5 3 1 29 34 27
Computational time	**6 seconds**

In Figure 35 the RTT, ACK and TCP from SCC to RTU-2 are shown and in Figure 36 the travel times of the packet send by the MITM node to RTU-2 are shown.

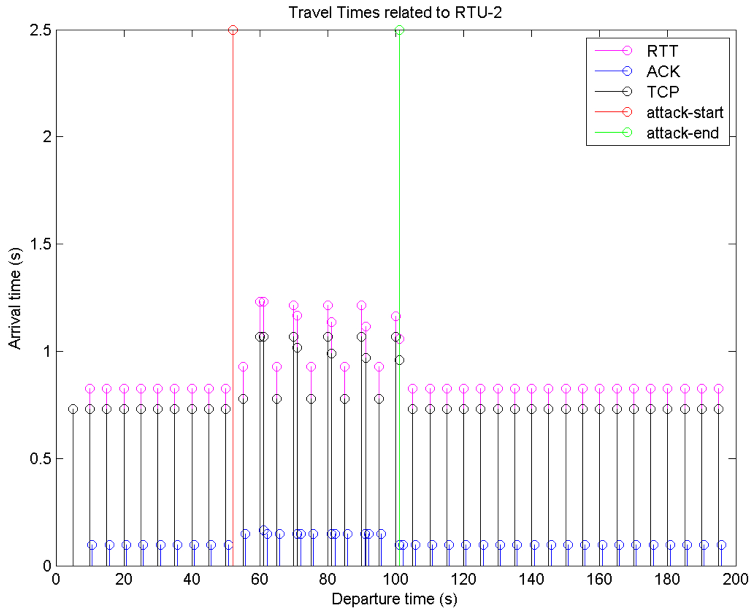


Figure 35: RTT from SCC to RTU-2 with an attacker that is between the Ethernet bus and Gateway

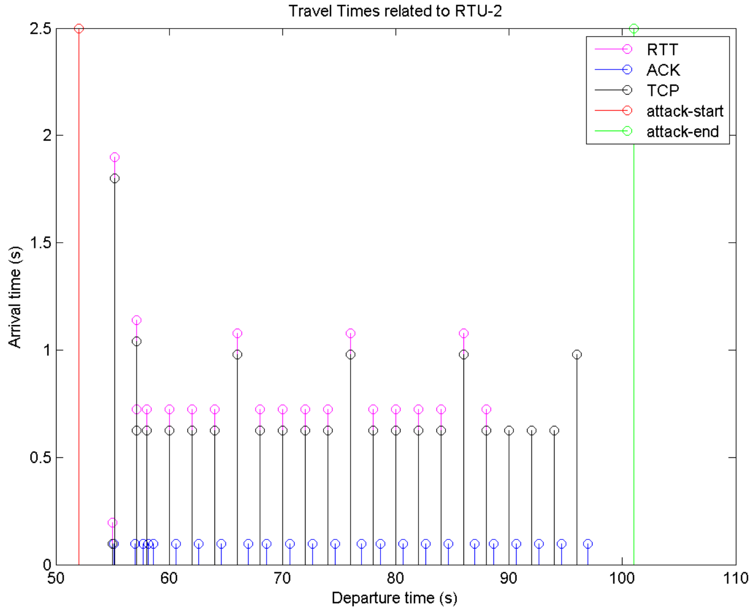


Figure 36: MITM packets to RTU-2

6.7.3 MITM attack between Moscad-ND and RTU-2

Table 9 presents the results in the case where the attack is between Moscad-ND and RTU-2. In the last row the computational time is reported, while in the other three rows, the traversed devices before the attack, during the attack and after the attack are shown respectively.

Table 9: MITM attack between Moscad-ND and RTU-2

	Traversed devices
before attack	• 27 34 29 1 3 5 • 5 3 1 29 34 27
during attack	• 27 34 29 1 3 **38** 5 • 5 **38** 3 1 29 34 27
after attack	• 27 34 29 1 3 5 • 5 3 1 29 34 27
Computational time	**4 seconds**

Remember that the relationship between the numbers and devices of SCADA and corporate network of Table 9 is shown in Table 6.

Figure 37 shows the arrival times (RTT, ACK and TCP) of the communication from SCC to RTU-2 under MITM attack which occurs between Moscad-ND and RTU-2 and in Figure 38 are shown the relative "travel times" of the packets send by the MITM node to RTU-2.

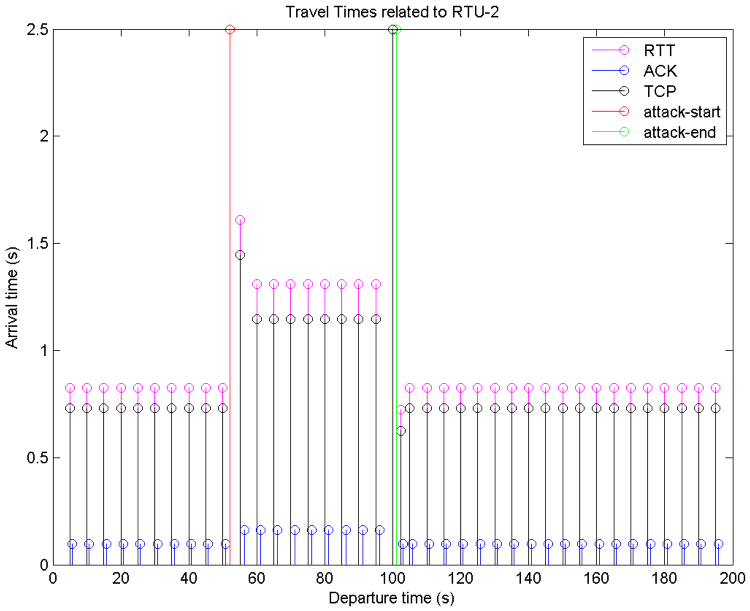


Figure 37: RTT from SCC to RTU-2 with an attacker that is between Moscad-ND and RTU-2

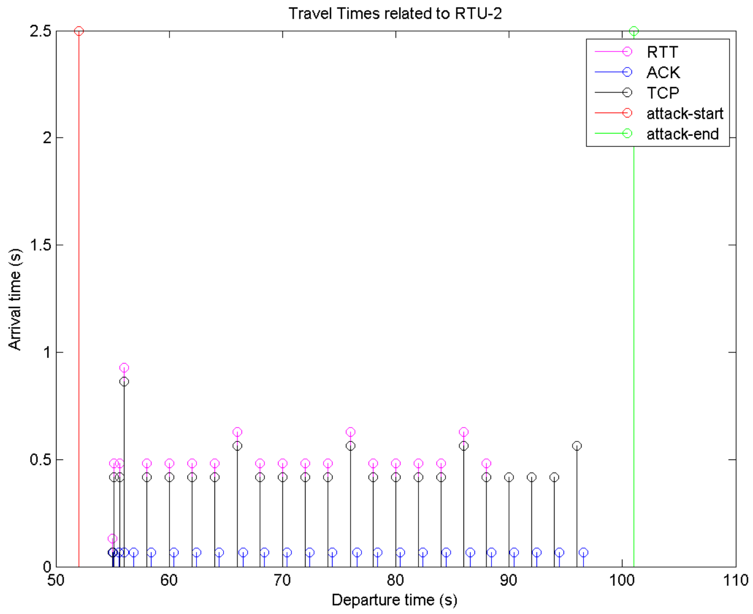


Figure 38: MITM packets to RTU-2

6.7.4 MITM attack on Internet

Figure 39 and Figure 40 show the communication from SCC and RTU-2 when a MITM attack occurs on Internet, in terms of packet "travel times" from the MITM node to RTU-2, and respectively.

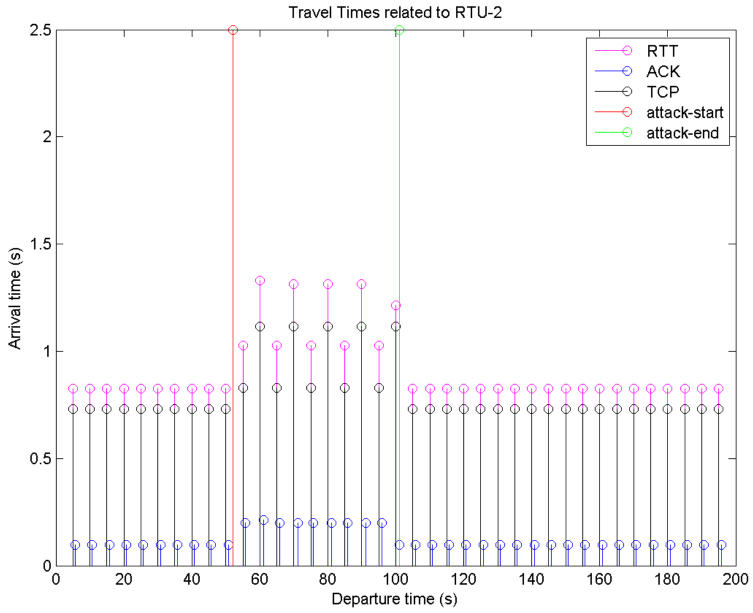


Figure 39: RTT from SCC to RTU-2 with an attacker that is on Internet

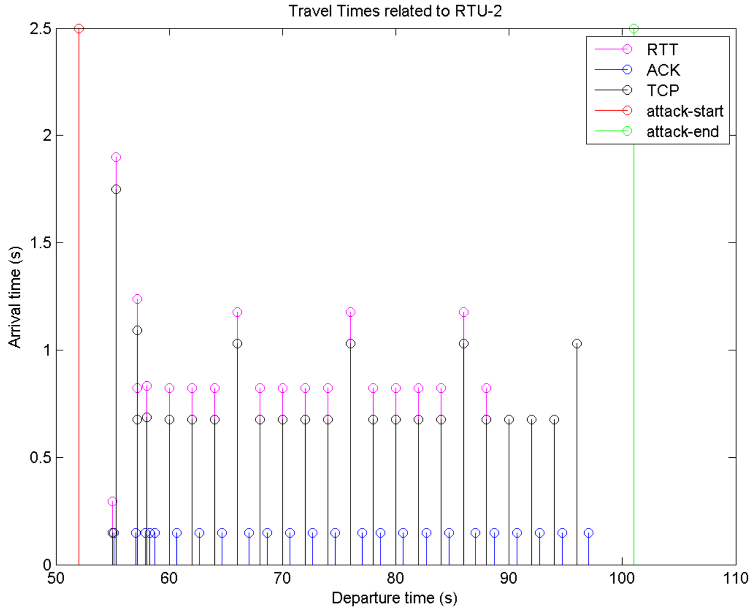


Figure 40: MITM packets to RTU-2

Table 10 reports the computational time and all the traversed devices the communication between SCC and RTU-2, in case of MITM attack on internet.

Table 10: MITM attack on Internet

	Traversed devices
before attack	• 27 34 29 1 3 5 • 5 3 1 29 34 27
during attack	• 27 34 35 **38** 36 29 1 3 5 • 5 3 1 29 36 **38** 35 34 27
after attack	• 27 34 29 1 3 5 • 5 3 1 29 34 27
Computational time	**6 seconds**

6.7.5 Differences between DoS and MITM attacks

Some differences between DoS and MITM attacks, related to LoC and LoV indicators, presence of the attack tail after the end of the attack, packet route modification, packets transmission time/frequency variation along attack phases and its end, have been observed:

- In the case of DoS, the LoC or the LoV is relevant dependently on the flow direction of the attack; in MITM there are no evidence of LoC or LoV.
- In case of DoS there is a tail, its length depends on the scheduling; in case of MITM there is no tail.

- In DoS there seems to be no route modification, if there is, it has no effect; in MITM there is a route modification. The modification of the route contains the position of the MITM attacker that is a new node with respect to the set of nodes that constitute SCADA plus corporate network devices.

- In DoS there is packets transmission time/frequency variation due to the congestion and the consequent activation of the AIMD mechanism; in MITM the AIMD mechanism is not activated.

7 RAO: Agent based models

The chapter describes simulation model for quality of service (QoS) indicators calculation for CockpitCI reference scenario. The model is developed using the Intelligent RAO Simulator presented in deliverable D2.1. The model is based on the modelling framework presented in deliverable D2.2. As a reminder, the reference scenario consists of Fault Isolation and System Restoration (FISR) process executed by SCADA control centre on a feeder with faulty line and of a cyber attack scenario. The model was executed with numerous scenarios of cyber attacks to illustrate how these attack scenarios influence QoS while executing FISR. As a reminder, the simulation model presented here extends and completes the ECI and FISR model developed in MICIE project and described in MICIE deliverable D2.2.3.

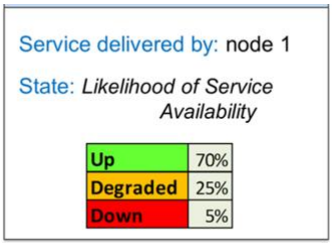
7.1 Additional element type parameters

To simulate behaviour of CCI/SCADA elements under cyber attacks, the following parameters were added to all CCI/SCADA element types:

- *CS_Up* - current element state rating "Up"
- *CS_Degraded* - current element state rating "Degraded"
- *CS_Down* - current element state rating "Down"; note that one of these three parameters must have value 1.0 and others must be 0.0

- *MC_CS_Up* - current Monte-Carlo run state rating "Up" of the element
- *MC_CS_Degraded* - current Monte-Carlo run state rating "Degraded" of the element
- *MC_CS_Down* - current Monte-Carlo run state rating "Down" of the element"
  Note that one of these three parameters must have value 1.0 and others must be 0.0. For each run these values are generated randomly on the basis of base three state ratings of the element (see below)

- *BS_Up* - base element state rating "Up" for Monte-Carlo simulation
- *BS_Degraded* - base element state rating "Degraded" for Monte-Carlo simulation
- *BS_Down* - base element state rating "Down" for Monte-Carlo simulation



Service delivered by: node 1

State: *Likelihood of Service Availability*

Up	70%
Degraded	25%
Down	5%

We suppose that if current element state ranking "Up" is equal to 1.0, the element is not affected by cyber attack and it receives and sends messages normally, with no delay nor manipulation. If current element state ranking "Down" is equal to 1.0, the element is completely out of service, so does not process messages; incoming messages are simply lost. If current element state ranking "Degraded" is equal to 1.0 we should define what the degraded behaviour is. The model supports the following types of degraded behaviour: messages are processed without compromising them but with delay, fixed or random. To describe this, following parameters are added:

- *Behaviour_degraded* - type of degraded behaviour (fixed or random delay)
- *Delay_degraded_min* - minimum value of delay if degraded
- *Delay_degraded_max* - maximum value of delay if degraded

Following parameters are also added to manage Monte-Carlo simulations:

- *i_Monte_Carlo* - number of current Monte-Carlo run
- *i_fault* - number of faulty segment; remember we have 7 segments in our reference scenario electrical greed, so in each simulation we need to simulate a fault in each of 7 segments to calculate QoS

Following parameters are introduced to collect some statistics about elements functioning under cyber attacks:

- *Mes_sent* - counter of messages sent by element
- *Mes_received* - counter of messages received by element
- *Mes_passed_OK* - counter of messages processed by element without any manipulation nor delay
- *Mes_delayed* - counter of messages delayed by element; note that if the message is delayed and changed and/or derouted, all corresponding counters are increased by 1
- *Mes_stopped* - counter of messages received by element but not sent to next element in the message route
- *Mes_changed* - counter of messages changed (compromised, manipulated) by element
- *Mes_derouted*- counter of messages derouted by element, i.e. sent to wrong next element
- *Mes_destroyed* - counter of messages destroyed by element

It is possible on the basis of values of counters at the end of simulation to calculate statistics about element behaviour. Not all these counters are used by current version of simulation model.

Finally, three more parameters are added to collect statistics while doing Monte-Carlo simulations:

- *Count_Up* - counter of Monte-Carlo simulations with state ranking "Up" of element equal to 1.0
- *Count_Degraded* - counter of Monte-Carlo simulations with state ranking "Degraded" of element equal to 1.0
- *Count_Down* - counter of Monte-Carlo simulations with state ranking "Down" of element equal to 1.0

The parameters described above are added to all types of objects representing CCI/SCADA elements, namely to following types:

- an_HV_MV_substation - for communication room of HV/MV substation
- a_Feeder - for feeder RTU
- a_Grid_node - for field RTU
- a_SCADA - for SCADA HMI
- a_Gateway - gateways
- an_FIU - field interface units
- a_Radio_VHF_Unit - radio frequency units
- a_CCI_element - other CCI/ SCADA elements not covered by previous types

7.2 Model instantiation (system elements)

The simulation model instantiates CCI/SCADA structure of CockpitCI reference scenario, given on Figure 41. So, the model includes:

- 1 WIZCON SCADA HMI
- 2 MCPT gateways (primary and backup)
- 2 FIU MOSCAD (local and remote)
- 2 radio VHF units
- 8 CCI/SCADA elements (TeX-CR-AREA-CENTER, TeX-NA-AREA-CENTER, LeX-DN-VHF, WIZCON CLIENT, NMS CONTRO, PoP, PoP-ND, BUS Ethernet)
- 7 HV/MV substations (TF, CB, NM, BL, MS, CR, ML)
- 2 feeders (Zuriel and Hanita)
- 13 field RTU installed on greed nodes (10 for Hanita feeder and 3 for Zuriel)

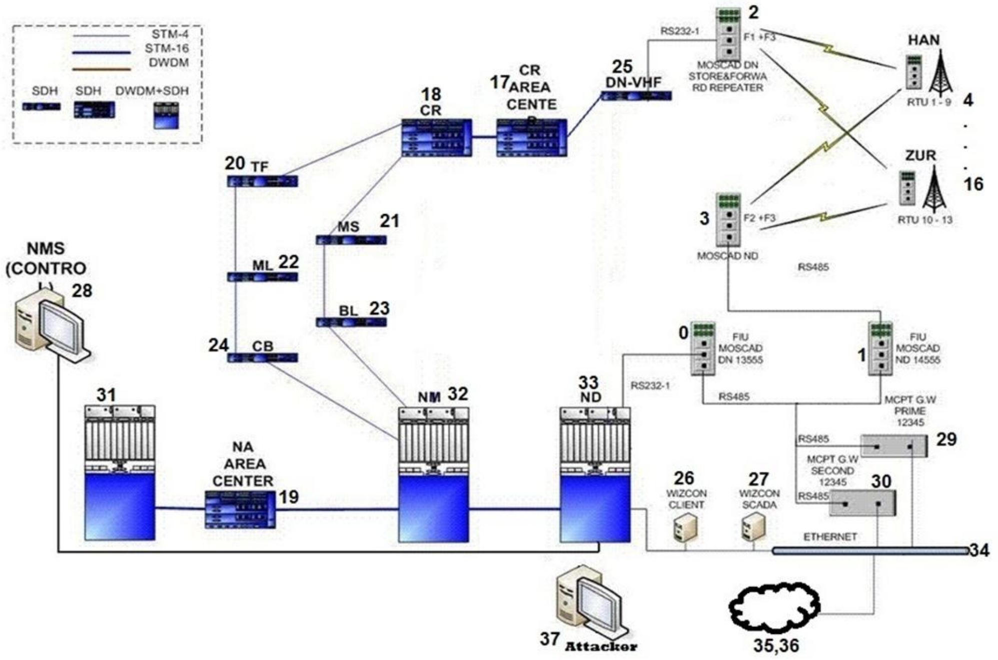


Figure 41: CCI/SCADA structure of CockpitCI reference scenario

Below, the table of the correspondence between element number in the model and its denomination has been reported again for a better understanding of the work.

Element number	Denomination
0	FIU_DN
1	FIU_ND
2	MOSCAD_DN

3	MOSCAD_ND
4	RTU_HAN_1
5	RTU_HAN_2
6	RTU_HAN_3
7	RTU_HAN_4
8	RTU_HAN_5
9	RTU_HAN_6
10	RTU_HAN_7
11	RTU_HAN_8
12	RTU_HAN_9
13	RTU_ZUR_10
14	RTU_ZUR_11
15	RTU_ZUR_12
16	RTU_ZUR_13
17	Tex_CR_AREA_CENTER
18	TeX_CR
19	Tex_NA_AREA_CENTER
20	LeX_TF
21	LeX_MS
22	LeX_ML
23	LeX_BL
24	LeX_CB
25	Lex_DN_VHF
26	WIZCON_CLIENT
27	WIZCON_SCADA
28	NMS_CONTRO

29	GATEWAY_PRIME
30	GATEWAY_SECOND
31	PoP
32	PoP_NM
33	PoP_ND
34	BUS_Ethernet
35	INTERNET
36	INTERNET_
38	MITM
41	RTU_Zuriel
42	RTU_Hanita

7.3 Messages and routes

Messages are presented in the model by objects of corresponding type a_Message (temporary objects), having the following important parameters.

- *Number* - unique message number
- *Creation_time* - message creation time
- *Sender_Element* - type of element who sends the message
- *Sender_Element_num* - number of element who sends the message (number of element in the set of elements of the same type)
- *Sender_Element_ICT_num* - unique number of element who sends the message in CCI/SCADA (numbers of elements are given in table above)
- *Dest_Element* - type of destination element for the message
- *Dest_Element_num* - number of destination element for the message (number of element in the set of elements of the same type)
- *Dest_Element_ICT_num* - unique number of element who sends the message in CCI/SCADA (numbers of elements are given in table above)
- *Message_type* - type of message; types currently introduced in the model are: RTU_switch_status, RTU_battery_status, RTU_AC_loss, RTU_Command
- *Route_number* - number of route currently selected for message delivery
- *Route_step* - counter of nodes along the route already passed by message
- *Next_node* - number of next node along the route (unique number of element who sends the message in CCI/SCADA)

Following parameters are dedicated to represent SCADA commands sent to RTU or feeder_RTU while executing FISR, they reflect message payload for SCADA commands as well as some specific information about manual operation of field switches:

- *Execution_order* - the order in which commands are to be executed regardless their delivery order; this is important because of technical and structural constraints;
- *SS_number* - number of substation to which belongs the feeder concerned by current SCADA procedure;
- *Feeder_number* - number of the feeder;
- *Element* - corresponding element to act on, whether it is a breaker or a switch;
- *Element_number* - element (breaker or switch) number;
- *Action* - what to do with the element - to open or to close;
- *Team_allocated* - number of maintenance team assigned to do the action in case of manual operation
- *Manual_delivery_duration* - time it takes to execute the action manually (this is mainly travelling time to the field switch)

Common message parameters continued:

- *State* - message state, may have many values reflecting message state during its life, namely issued, delivered, processed, in_delivery, in_manual_delivery, waiting_for_execution, executed, etc.
- *ICT_Number* -current node number where the message is (number in CCI/SCADA structure)
- *Delay_at_cur_ICT_el* - delivery delay in current node (zero if no cyber attack)
- *Destiny* - indicates what happened to the message, may have values arrived_OK, stopped, arrived_changed, destroyed, derouted
- *Delivery_duration* - time it took to deliver the message
- *Processing_time* - time when the action is executed

To represent routes, an object type a_Route is introduced in the model with following parameters:

- *Number* - route number for identification
- *Message_type* - type of messages sent through this route; message types currently introduced in the model are: RTU_switch_status, RTU_battery_status, RTU_AC_loss, RTU_Command
- *ICT_Number_sender* - unique number of element who sends the message in CCI/SCADA (numbers of elements are given in table above)
- *ICT_Number_dest* - unique number of element who sends the message in CCI/SCADA (numbers of elements are given in table above)
- *ICT_Number_route_01 ... ICT_Number_route_15*: - parameters storing node numbers of the route, up to maximum 15

And following parameters serve to collect some statistics about messages delivered by a given route.

- *Messages_sent* - number of messages sent through the route
- *Messages_received* - number of messages received by destination element
- *Messages_lost* - number of messages lost due to compromised behaviour of one of the nodes of the route
- *Delivery_duration* - total delivery duration for all messages delivered

Example of routes instantiation for messages of type Command sent by SCADA HMI to field RTUs 1, 2 and 3 are given below. Node numbers correspond toFigure 41: CCI/SCADA structure of CockpitCI reference scenarioFigure 41 and the table above.

```
Route_101 : a_Route 101 RTU_Command 27  4 34 29  0 33 32 24 22 20 18 17 25
2 -1 -1 -1 * * * *

Route_102 : a_Route 102 RTU_Command 27  5 34 29  1  3 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 * * * *

Route_103 : a_Route 103 RTU_Command 27  6 34 30  0 33 32 23 21 18 17 25  2
-1 -1 -1 -1 * * * *
```

7.4 Cyber attack simulation

Cyber attacks representation consists of a set of elementary attacks altering state rankings of just one CCI/SCADA element. Elementary attack is described by objects of type a_CyberAttack, having the following parameters:

- *Number* - the elementary attack sequential number for identification;
- *Annee, Date, Mois, Heure, Minute* - the time of the attack occurrence, correspondingly year, day, month, hour and minute;
- *Step* - model time step number when the elementary attack occurred;
- *ICT_number* - unique number of element in CCI/SCADA (numbers of elements are given in table above) who's state ranking change

- *CS_Up* - new element state rating after attack: "Up"
- *CS_Degraded* - new element state rating after attack: "Degraded"
- *CS_Down* - new element state rating after attack: "Down"; note that one of these three parameters must have value 1.0 and others must be 0.0

- *Status* - status of the elementary attack, one of the following values: to_happen (the attack is yet to happen), happened (the attack has happen with corresponding changes in the element state rankings)

- *BS_Up*- new base element state rating "Up" for Monte-Carlo simulation
- *BS_Degraded* - new base element state rating "Degraded" for Monte-Carlo simulation
- *BS_Down* - new base element state rating "Down" for Monte-Carlo simulation

Following parameters are also added to manage Monte-Carlo simulations:

- *i_Monte_Carlo* - number of current Monte-Carlo run
- *i_fault* - number of faulty segment; remember we have 7 segments in our reference scenario electrical greed, so in each simulation we need to simulate the attack while executing FISR in each of 7 segments to calculate QoS

Three more parameters are also added to collect statistics while doing Monte-Carlo simulations:

- *Count_Up* - counter of Monte-Carlo simulations with state ranking "Up" of the corresponding element, generated randomly, equal to 1.0

- *Count_Degraded* - counter of Monte-Carlo simulations with state ranking "Degraded" of the corresponding element, generated randomly, equal to 1.0
- *Count_Down* - counter of Monte-Carlo simulations with state ranking "Down" of the corresponding element, generated randomly, equal to 1.0

Now, a cyber attack, including possibly countermeasures, is presented by a set of objects described above. For example, let us consider the attack in table below.

Time step	**CCI/SCADA element**	**Up**	**Degraded**	**Down**
6	2 (Radio VHF Unit 1)	0.0	1.0	0.0
8	3 (Radio VHF Unit 2)	0.0	1.0	0.0
10	2 (Radio VHF Unit 1)	1.0	0.0	0.0
12	2 (Radio VHF Unit 1)	0.0	1.0	0.0

One can see that the element number 2 (Radio VHF Unit 1) is compromised at time step 6 (Degraded = 1.0) and measures are done at time step 10 to repair the element (Up = 1.0). This attack is presented by the following objects:

```
CyberAttack_31_Radio_VHF_Unit_1 : a_CyberAttack 3 0 0 0 0 0  6  2 0.0 1.0
0.0 * 0.0 1.0 0.0 * * * * *

CyberAttack_32_Radio_VHF_Unit_2 : a_CyberAttack 3 0 0 0 0 0  8  3 1.0 0.0
0.0 * 0.4 0.4 0.2 * * * * *

CyberAttack_33_Radio_VHF_Unit_1 : a_CyberAttack 3 0 0 0 0 0 10  2 1.0 0.0
0.0 * 0.6 0.3 0.1 * * * * *

CyberAttack_34_Radio_VHF_Unit_1 : a_CyberAttack 3 0 0 0 0 0 12  2 0.0 0.0
1.0 * 0.0 0.0 1.0 * * * * *
```

Overall, the simulation model consists of:

- a set of objects describing system composition and initial state (Data base)
    - 220 permanent objects (temporary objects are created while simulating) belonging to 20 object types (substation, breaker, line, FIU, gateway, SCADA, message, route, etc.)
- a set of activities describing system behaviour (Knowledge base)
    - 240 activities of 132 types (toggle breaker state, send a message, repair a line, transmit a message, etc.)
- Animation description to illustrate system state
    - 5 main screens

7.5 Animation screen: elements state rankings and cyber attack progress

The animation screen showing current state rankings for all elements is given on Figure 42. The value of "Up" ranking is given on green background, the value of "Degraded" ranking is shown on yellow background and the "Down" ranking is on the red one.

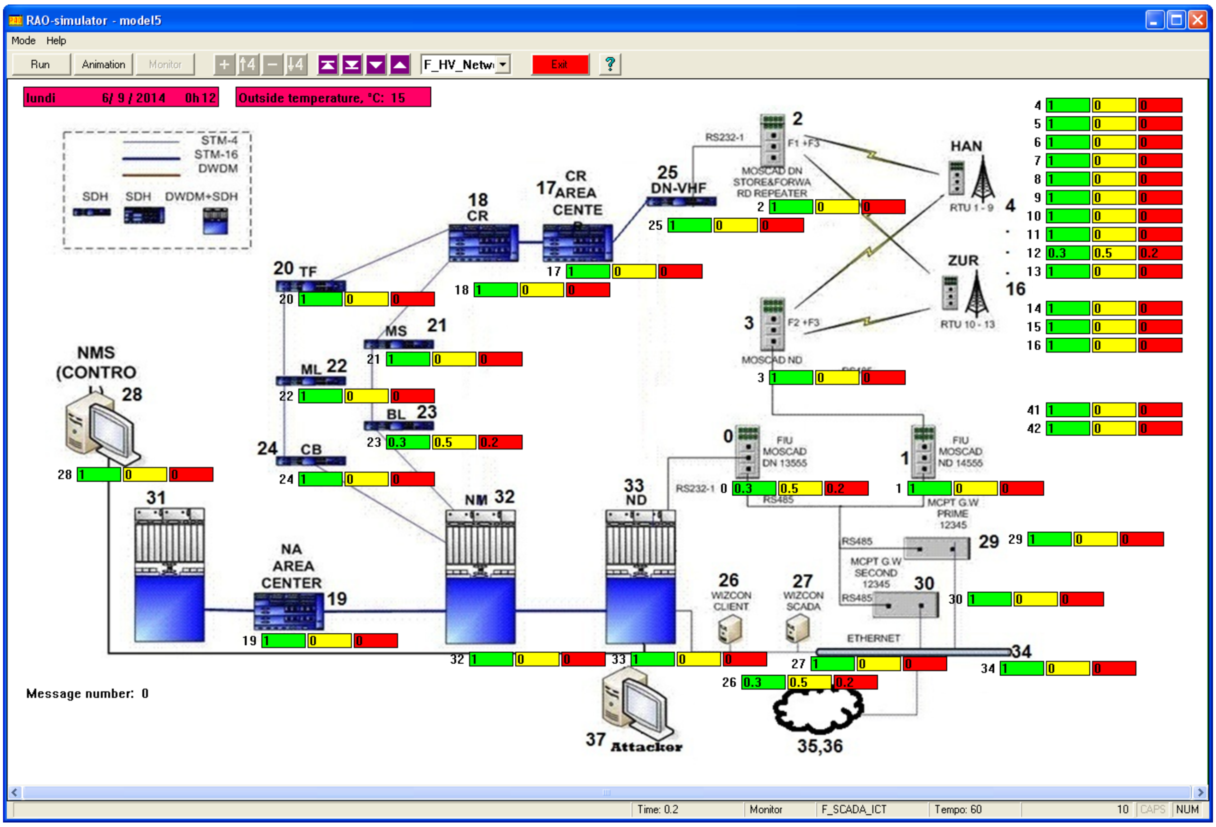


Figure 42: CCI/SCADA elements state rankings animation screen

7.6 Test simulation runs

First of all, we simulate the FISR execution in normal CCI/SCADA operation.

Table 11 shows the results of one FISR execution simulation on all 7 segments of the reference scenario grid. Quality of service indicator, used by IEC, is Tn.

Besides the Tn, table gives the values of other indicators allowing to better understand system functioning: overall FISR duration in minutes, percentage of time when each customer was de-energized related to overall FISR duration, number of SCADA messages (representing FISR commands) sent and average messages delivery time in minutes.

Table 11. No cyber attack

Indicator	**Segment number**						
	1	2	3	4	5	6	7
Duration, min	10	11	12	13	14	52	16
Tn, min	**5.28**	**8.19**	**6.9**	**7.1**	**7.26**	**20.87**	**8.58**
Customer 1	54.5%	45.5%	0%	46.2%	50%	86.5%	56.3%
Customer 2	54.5%	0%	50%	53.8%	57.1%	88.5%	62.5%
Customer 3	54.5%	45.5%	41.7%	38.5%	35.7%	9.6%	0%
Customer 4	54.5%	45.5%	41.7%	38.5%	35.7%	0%	37.5%
Commands sent	6	10	16	12	17	19	13
Delivery time, min	0	0	0	0	0	1.95	0

Note that even in normal CCI/SCADA functioning the average messages delivery time for FISR in segment number 6 is not zero. This is due to the fact that the FISR for this segment involves one not remotely controlled switch, so there is manual switching and travel time for maintenance team.

7.6.1 Static cyber attacks

A cyber attack happened before the beginning of FISR process may be called static. In this case we have elements state rankings affected by cyber attack, but not changing any more during FISR process execution.

Let us consider the following attacks:

Time	**Element**	**Up**	**Degraded**	**Down**
0	2 (Radio VHF Unit 1)	0.0	1.0	0.0

Time	**Element**	**Up**	**Degraded**	**Down**
0	3 (Radio VHF Unit 2)	0.0	1.0	0.0

Here, one of Radio VHF Units is under cyber attack, causing the "Degraded" state ranking.

Table 12 and Table 13 give simulation results for this attack. Remember we suppose that Radio VHF Unit delays messages by 2 minutes once it is in "Degraded" state.

Table 12. Compromised state of Radio VHF Unit 1

Indicator	**Segment number**						
	1	2	3	4	5	6	7
Duration, min	16	18	20	16	20	59	20
Tn, min	9.28	14.03	12.52	10.13	10.64	25.9	11.77
Customer 1	43.75%	38.9%	0%	31.3%	45%	78%	45%
Customer 2	43.75%	0%	45%	43.8%	55%	81.4%	55%
Customer 3	43.75%	38.9%	35%	31.3%	35%	11.9%	0%
Customer 4	43.75%	38.9%	35%	31.3%	35%	3.4%	35%
Commands sent	6	10	16	12	17	19	13
Delivery time, min	1.333	1.6	1.25	0.67	0.71	2.89	0.92

Table 13. Compromised state of Radio VHF Unit 2

Indicator	**Segment number**						
	1	2	3	4	5	6	7
Duration, min	10	11	16	20	22	61	20
Tn, min	5.28	8.19	8.18	10.59	11.03	25.28	10.48
Customer 1	54.5%	45.5%	0%	45%	50%	82%	55%
Customer 2	54.5%	0%	62.5%	60%	63.6%	86.9%	70%
Customer 3	54.5%	45.5%	43.8%	35%	31.8%	11.5%	0%
Customer 4	54.5%	45.5%	43.8%	35%	31.8%	0%	25%
Commands sent	6	10	16	12	17	19	13
Delivery time, min	0	0	0.5	1	1.06	3	0.92

One can see significant increase in Tn values in this case. This is quite in line with what we would anticipate intuitively, but the model gives precise values, allowing to estimate potential damage and thus allowing better risk management. The QoS degradation is even more pronounced in case of cyber attack on both radio VHF stations, as below:

Time	**Element**	**Up**	**Degraded**	**Down**
0	2 (Radio VHF Unit 1)	0.0	1.0	0.0
0	3 (Radio VHF Unit 2)	0.0	1.0	0.0

Table 14 gives the results for this case.

Table 14. Compromised state of both Radio VHF Unit 1 and 2

Indicator	**Segment number**						
	1	2	3	4	5	6	7
Duration, min	16	18	20	22	24	63	24
Tn, min	9.28	14.03	12.52	13.04	13.47	27.73	13.28
Customer 1	43.75%	38.9%	0%	40.9%	45.8%	79.4%	54.2%
Customer 2	43.75%	0%	45%	50%	54.2%	82.5%	62.5%
Customer 3	43.75%	38.9%	35%	31.8%	29.2%	11.1%	0%
Customer 4	43.75%	38.9%	35%	31.8%	29.2%	0%	29.2%
Commands sent	6	10	16	12	17	19	13
Delivery time, min	1.333	1.6	1.75	1.67	1.76	3.74	1.85

7.6.2 Dynamic cyber attacks

Dynamic attack is the one which is still spreading during the FISR process itself. In this case, some SCADA commands can pass instantly, other can be delayed or lost. This complicates the model. For example, the attack presented in 7.4 gives the results in Table 15.

Table 15. Dynamic cyber attack 1

Indicator	**Segment number**						
	1	2	3	4	5	6	7
Duration, min	10	17	17	21	23	62	23
Tn, min	5.28	12.97	9.47	11.98	12.42	26.68	12.22
Customer 1	54.5%	41.2%	0%	42.9%	47.8%	80.6%	56.5%

Customer 2	54.5%	0%	52.9%	52.4%	56.5%	83.9%	65.2%
Customer 3	54.5%	41.2%	41.2%	33.3%	30.4%	11.3%	0%
Customer 4	54.5%	58.8%	41.2%	33.3%	30.4%	0%	30.4%
Commands sent	6	10	16	12	17	19	13
Delivery time, min	0	1.4	1.25	1.5	1.65	3.63	1.69

If the radio VHF unit 1 after recovering to normal status at time step 10 is compromised at time step 12 by the attacker to status down, as below, then the QoS indicators are those presented in Table 16.

Time step	**CCI/SCADA element**	**Up**	**Degraded**	**Down**
6	2 (Radio VHF Unit 1)	0.0	1.0	0.0
8	3 (Radio VHF Unit 2)	0.0	1.0	0.0
10	2 (Radio VHF Unit 1)	1.0	0.0	0.0
12	2 (Radio VHF Unit 1)	0.0	0.0	1.0

Table 16. Dynamic cyber attack 2

Indicator	**Segment number**						
	1	2	3	4	5	6	7
Duration, min	10	17	17	21	23	62	23
Tn, min	5.28	12.97	9.47	11.98	12.42	26.68	12.22
Customer 1	54.5%	41.2%	0%	42.9%	47.8%	80.6%	56.5%
Customer 2	54.5%	0%	52.9%	52.4%	56.5%	83.9%	65.2%
Customer 3	54.5%	41.2%	41.2%	33.3%	30.4%	11.3%	0%
Customer 4	54.5%	58.8%	41.2%	33.3%	30.4%	0%	30.4%
Commands sent	6	10	16	12	17	19	13
Delivery time, min	0	1.4	1.25	1.5	1.65	3.63	1.69

Note that the results are worse for all segments but number 1. This is because the FISR lasts 10 minutes for this segment, so the radio VHF unit 1 becoming out of service at time 12 does not influence the QoS.

8 Attack and defence trees

In order to formally specify the way in which an ICT system can be attacked, we need a way to model threats against computer systems. If we can understand all the different ways in which a system can be attacked, we can likely design countermeasures to thwart those attacks. And if we can understand who the attackers are maybe we can install the proper countermeasures to deal with the real threats. Needless to say, the characterization of the attack and the choice of the countermeasures require new conceptual approach and extended analytical tools as described in [2,69].

Attack trees provide a formal, methodological way of describing the security of systems, based on varying attacks. Basically, attacks against a system can be represented in a tree structure, with the goal as the root node and different ways of achieving that goal as hierarchies of events and leaf nodes.

An attack tree is a multi-level hierarchical structure based on logical AND and OR operators. The top node is the ultimate goal with the grouping of different subgoals. The grouping can be composed with a number of attack leaves that are attributed with logic operators AND or OR [70]. In drawing attack trees we use the symbols commonly used in Fault Tree analysis [71] even if this way of representing attack trees is not standard and different representations can be found in the literature [72,73,74].

The methodology of Attack Trees has been also applied to SCADA systems [72]. Once the attack tree is generated, attack events can be considered as binary events: present or non-present. Any Boolean value can be assigned to the leaf nodes and then propagated up the tree structure to determine which combination of attack events lead to the final goal. Borrowing the terminology from fault tree analysis, we can identify the list of the minimal cut sets (mcs) as the list of the minimal combinations of elementary events that lead to the final goal (attack). The number of elementary events in an mcs is called order of the mcs. However, we can proceed to a more quantitative view of the attack phenomenon if we are able to assign probabilities to the elementary attack events to be present and effective.

Different attack strategies may have different costs so that it would be better to show exactly how expensive an attack is. It is then possible to assign continuous values to nodes. This opportunity can be exploited by assigning a cost to the leaf nodes, representing the cost of implementing the attack specified by the leaf. Like Boolean node values, these can propagate up the tree as well, so that the cost of possible attack scenarios can be evaluated. Often the value of an attack is not only measured with the cost of implementing the attack itself, but also with the (economic) impact that the specific attack has on the system. Hence, a second cost function can be associated to the tree called impact cost. The choice for an attacker can be a trade off between the cost, the impact and the probability of success of the attack. Up to now, the basic formalism of attack trees does not include defence mechanisms.

Defence trees [75] incorporate defence mechanisms or countermeasures [73] in attack trees so that the point of view of the attacker as well as the point of view of the defender can be analysed. Also in defence trees, the cost of implementing the countermeasures can be added to the system and included in the analysis.

Following we start with a preliminary example, to better explain the theory under the attack and defence tree, the goal of the attack, represented by the root of the attack tree, is the acquisition from an unauthorized user (hacker) of the root password of a Unix server with consequent possible attack to the system.

Attacks against a system can be regarded as sequences of smaller attack steps that can be described and analysed by means of the AT.

Description of the terminal leaves.

- **Vulnerability v1** - breaches in the login procedure to a server.
    - v1:1 - user password not correctly protected;
    - v1:2 - default password found in a user manual;
    - LG - a unauthorized user tries to login on the server (LOGGING-IN);
- **Vulnerability v2** - breaches in the protection of the root password
    - CG - a unauthorized user tries to crack the root password (CRACKING);
    - GG - a unauthorized user tries to guess the root password (GUESSING);

Vulnerability v1 may be decomposed in many different vulnerabilities. We show, for the sake of illustration, two common vulnerabilities like v1:1 that indicates that a user password is not correctly protected (as for instance written in plain characters on a sticker on the screen) or v1:2 that indicates that the user has never changed the default password assigned by the vendor and written in the user manual [76].

The sequence of attack steps occurs through the following intermediate events:

- LD - A unauthorized user tries to login AND _finds a breach v1, he can login (event LOGGED-IN);
- CR2 - A unauthorized user is logged-in AND tries to crack the root password (event CR2);
- CR1 – A unauthorized user is logged-in AND _finds a breach v2 (event CR1);
- CD - CR1 AND CR2 are true, the unauthorized user cracks the password (event CRACKED);
- GD - A unauthorized user is logged-in AND guesses the root password (event GUESSED);
- If the root password is cracked OR guessed the attack is successful and the ROOT of the attack tree is reached.

8.1 Qualitative analysis

The qualitative analysis is intended to find the combinations of elementary events that lead to the root. In the present case we have:

$ROOT = CD \vee GD = (CR_1 \wedge CR_2) \vee (LD \wedge GG) = (V_2 \wedge CG \wedge LD) \vee (LD \wedge GG) =$

$$= (V_2 \wedge CG \wedge V_1 \wedge LG) \vee (V_1 \wedge LG \wedge GG)$$

$$= (V_2 \wedge CG \wedge (V_{1:1} \vee V_{1:2}) \wedge LG) \vee ((V_{1:1} \vee V_{1:2}) \wedge LG \wedge GG) \qquad (1)$$

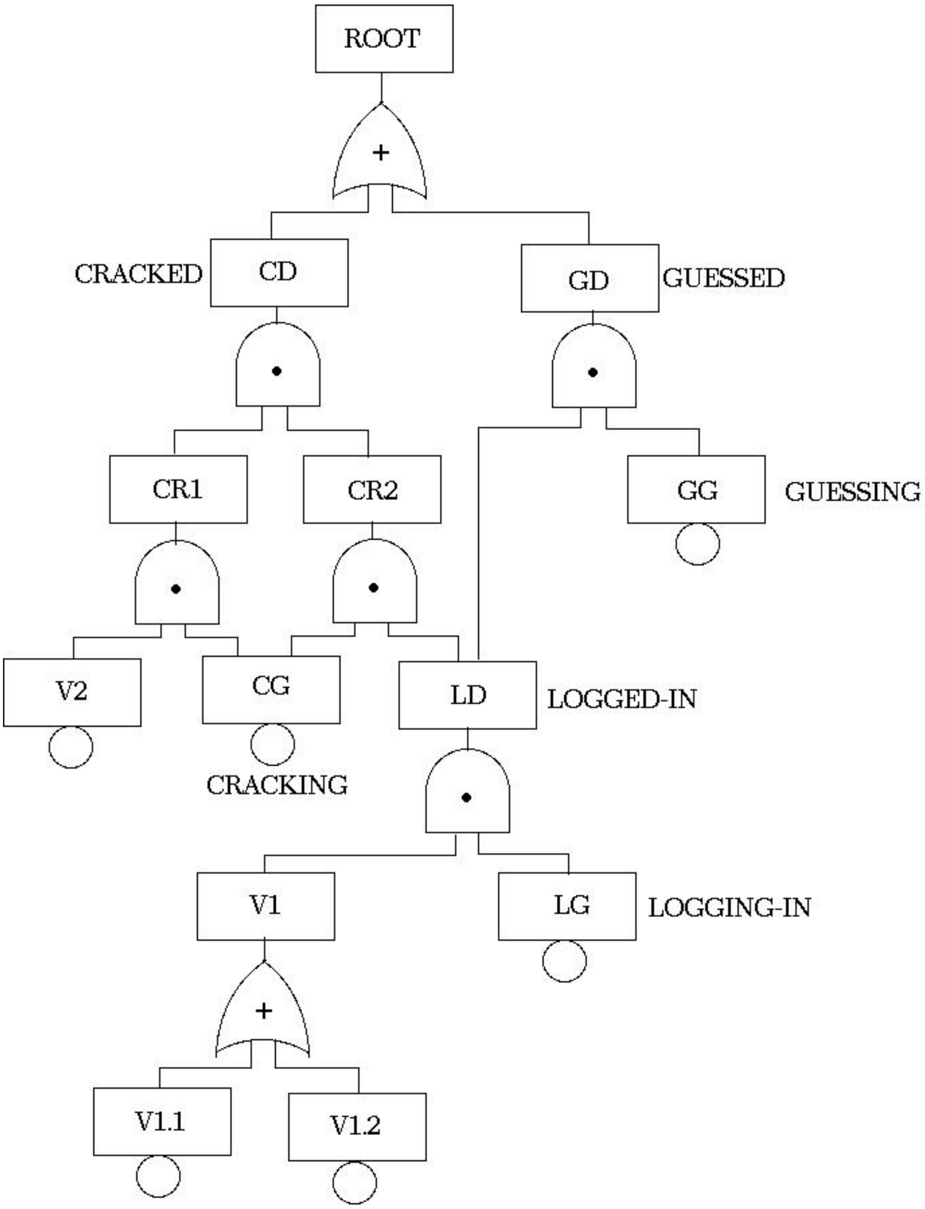


Figure 43: Cracking a Unix server

The analysis of equation (1) shows that there are 4 minimal combinations of events that lead to the final attack goal, the so called *mcs*.

The 4 *mcs* can be expressed as the following combinations of basic attack leaves:

mcs1 $V_{2} \wedge CG \wedge V_{1:1} \wedge LG$

mcs2 $V_{2} \wedge CG \wedge V_{1:2} \wedge LG$

mcs3 $V_{1:1} \wedge LG \wedge GG$

mcs4 $V_{1:2} \wedge LG \wedge GG$ (2)

The first two, *mcs1* and *mcs2*, have order 4, whereas *mcs3* and *mcs4* have order 3. The events represented by the terminal leaves can be assumed to be Boolean variables with two possible states (*true=1* and *false=0*). In this case, the goal of the attack, represented by the root event ROOT in Figure 43, is a Boolean function that can be suitably represented and analysed by means of the corresponding BDD. The BDD of the function of equation (1) is reported in Figure 44. The variable that is used as pivot at each level of the decomposition is reported on the left of the Figure 44.

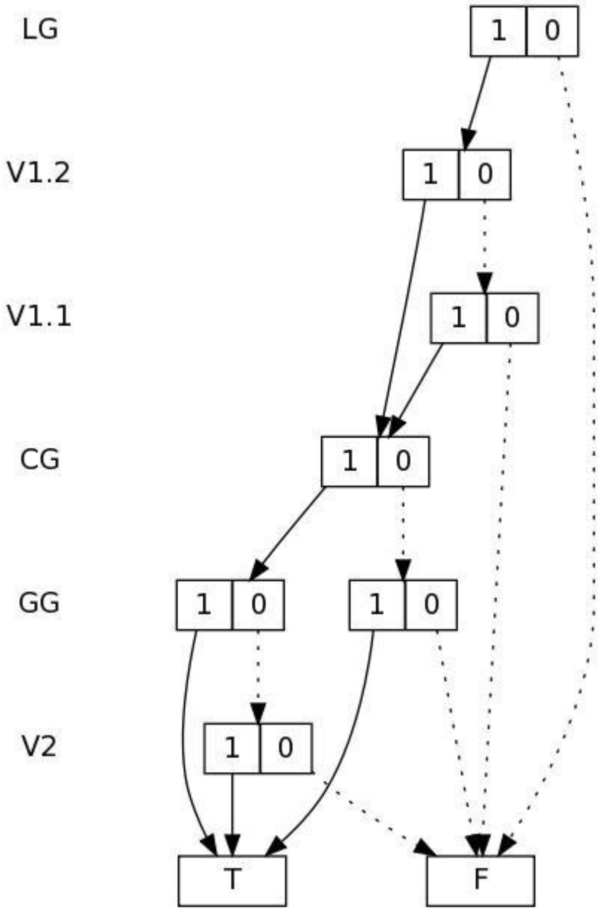


Figure 44: BDD of the AT of Figure 43

All the paths on the BDD of Figure 44 that go from the initial node (LG) to the terminal node T, indicate the sequence of actions that can be followed to launch an attack and coincide with the mcs of the AT obtained in (2).

Structural Importance Measure

The basic attack events do not have the same criticality in determining the ROOT. Hence, it is important to rank the events according to some structural importance index, that is based on the knowledge of the Boolean function associated to the root (equation (1)) [48,77].

Given an attack tree with n attack leaves, we can build the Boolean function B($\underline{X}$) where $\underline{X}$ is the *n*-dimensional vector representing the status (0 or 1) of the terminal leaves. $\underline{X}$ has 2^n possible values by combining the 0 and 1 of the n variables. By definition, B($\underline{X}$)=1 when the attack is successful and B($\underline{X}$)=0 when the attack is unsuccessful.

For each attack leaf i∈n we can apply the so called Shannon decomposition by defining two values of the function B($\underline{X}$):B_1(x_i=1;$\underline{X}$) when the value of x_i is stuck to 1 and B_0(x_i=0;$\underline{X}$) when the value of x_i is stuck to 0.

$$B_1(x_i = 1; \underline{X}) = B(x_1; x_2; : : : ; x_{i-1}; 1; x_{i+1}; : : : ; x_n)$$

$$B_0(x_i = 0; \underline{X}) = B(x_1; x_2; : : : ; x_{i-1}; 0; x_{i+1}; : : : ; x_n) \qquad (3)$$

Based on equations (3) the Shannon decomposition becomes:

$$B(\underline{X}) = (x_i \wedge B_1(x_i = 1;)) \vee (x_i \wedge B_0(x_i = 0; \underline{X})) \qquad (4)$$

Note that the Shannon decomposition is also the basic rule for the construction of the BDD [56]. If we compute B_1(x_i = 1; $\underline{X}$) and B_0(x_i = 0; $\underline{X}$) for all the 2^n combinations of the variables, we can define the structural importance coefficient of variable x_i .

$$I_{x_i}^{St} = \frac{\sum_{X \in 2^n} B_1\left(x_i = 1, \underline{X}\right) - B_0\left(x_i = 0, \underline{X}\right)}{2^n} \qquad (5)$$

8.2 Quantitative analysis

The second step is the quantitative analysis. If we are able to assign a probability to all the basic leaves of the attack tree (with reference to Figure 43, the basic events *V 1:1; V 1:2; V 2; CG; LG; GG*) we can calculate the probability of reaching the root event of the attack tree, the probability of the minimal combination of events that lead to the attack tree and a coefficient of criticality. If pi is the probability associated to the event x_i = *1*, *(1 - p_i)* is the probability associated to the event $x_i = 0$. Utilizing the Shannon decomposition, we can write:

$$P\left(B\left(\underline{X}\right)\right) = p_i \cdot P\left(B_1\left(x_i = 1, \underline{X}\right)\right) + (1 - p_i) \cdot P(B_0\left(x_i = 0, \underline{X}\right)) \qquad (6)$$

In the present example we have assumed that the *ALs* assume the values listed in the second column of Table 17. Propagating equation (6) along the BDD of Figure 44 we can compute the probability of the ROOT and of any intermediate events as well as of any *mcs*.

Table 17: Probability, cost and impact for the basic attack leaves of Figure 43.

AL	probability	cost	impact
V1.1	*0.3*	*150*	*120*
V1.2	*0.2*	*20*	*100*
V2	*0.3*	*100*	*300*
CG	*0.1*	*80*	*280*
LG	*0.1*	*100*	*200*
GG	*0.05*	*260*	*350*

Probabilistic Importance Measure.

When probabilities of attack leaves are known a new criticality index can be defined to rank the importance of the various leaves in determining the occurrence of the final attack (root of the tree). This new measure is called Birnbaum coefficient after [88], and it is defined as:

$$I_{x_i}^{B} = \frac{P\left(Root\ (x_i{=}1)\right) - P(Root(x_i{=}0))}{P(Root)} \qquad (7)$$

where:

P(Root) is the probability of the root of the tree;
P(Root(x_i = 1)) is the probability of the root of the tree when leaf x_i is stuck to 1;
P(Root(x_i = 0)) is the probability of the root of the tree when leaf x_i is stuck to 0.

The Birnbaum importance measure of an attack event represents the change in the probability of attack at the root caused by the probability difference when the attack leaf is present ($x_i = 1$) or not present ($x_i = 0$).

Attack cost and attack impact

A more effective and detailed analysis of an attack sequence should also include the cost of implementing the attack and the impact (in terms of monetary value) that the attack may have on the attacked system. We call attack cost or simply cost the cost of implementing a specific attack, and impact cost or simply impact the monetary damage caused by the attack.

Table 18: Attack cost vs mass and cumulative probability

cost	*mass probability*	*cumulative probability*
0	0.996546	
300	0.000600	0.000600
380	0.000970	0.001570
430	0.000720	0.002290
510	0.001164	0.003454

The propagation of the cost in the AT occurs with the following rules [52,89].

- In the presence of an *AND* gate both the cost and the impact of the events in input to the gate are summed (all the input events should be realized in order for the gate to be true).

- In the presence of an OR gate the cost and the impact behave in a different way: for the cost the attacker chooses the event with the minimum cost, for the impact the attacker chooses the event with the maximum impact.

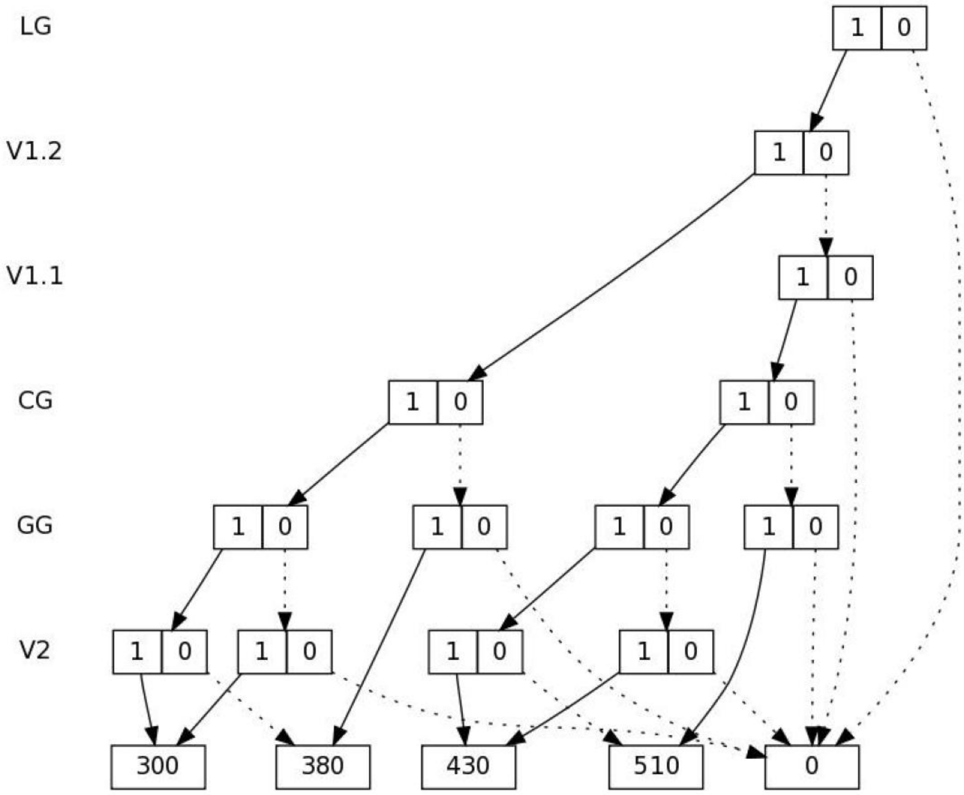


Figure 45: MTBDD of the cost function computed on the BDD of Figure 44

Columns 3 and 4 of Table 17 report the costs and the impacts that we have chosen for the different ALs (values inspired from [89]). The cost values can be added to the BDD by resorting to an extension of the BDD called MTBDD (Multi Terminal Binary Decision Diagrams) or Algebraic BDD [52,49]. MTBDD allows one to represent a real function of Boolean variables as a binary tree. While BDDs have only two terminal leaves 0 and 1, MTBDD can have more than two terminal leaves that identify all the possible values taken by the Boolean function along the paths from the root to the terminal leaves. Like BDD, MTBDD provide a compact representation of a weighted Boolean function by means of the Shannon's decomposition principle. Each node of the MTBDD represents a Boolean variable and has two successors: the left branch (in solid line) represents the value of the variable 1 and the right branch (in dotted line) represents the value of the variable 0. By adopting the propagation rules indicated above for the cost and for the impact, the terminal leaves of the MTBDD represent the minimum cost along that path or the maximum impact along that path.

By resorting to the MTBDD for the AT of Figure 43 the following measures can be obtained:

- When the AT is parameterized with the cost.
    - The minimum cost as a function of the probability of the attack. In other terms which is the probability that an attack with a total cost less than a given value can be successfully reached with a given probability.
    - The total minimum cost (and the probability) of the different *mcs*.
    - The probability mass distribution of the cost over the possible paths that lead to a successful attack.

The results are reported in Table 18 and in graphical form in the MTBDD of Figure 45. In Table 18, the first column reports the possible costs that can be incurred for a successful attack to reach the ROOT, the second column the mass probability of reaching the root with an attack of the corresponding cost and the third column the cumulative probability, i.e. the probability of successfully reaching the ROOT at a cost less than the value reported in the first column. The probability value 0.996546 at cost 0 indicated the probability that the attack is not successful.

- When the AT is parameterized with the impact.
    - The maximum impact as a function of the probability of the attack. In other terms which is the probability that an attack with a total impact greater than a given value can be successfully reached with a given probability.
    - The total maximum impact (and the probability) of the different mcs.
    - The probability mass distribution of the impact over the possible paths that lead to a successful attack.

The results for the impacts are reported in Table 19 and the corresponding MTBDD in Figure 46.

The meaning of the columns are the same as for the cost function, but we have added one more column indicating the survival probability (defined as the defective value minus the cumulative probability), that indicates the probability of having a successful attack with an impact greater than the corresponding value on the first column.

Table 19: Attack impact vs mass, cumulative and survival probability

impact	*mass probability*	*cumulative probability*	*survival probability*
0	0.996546		
650	0.000679	0.000679	0.002775
670	0.001455	0.002134	0.001320
880	0.000420	0.002554	0.000900
900	0.000900	0.003454	0.0

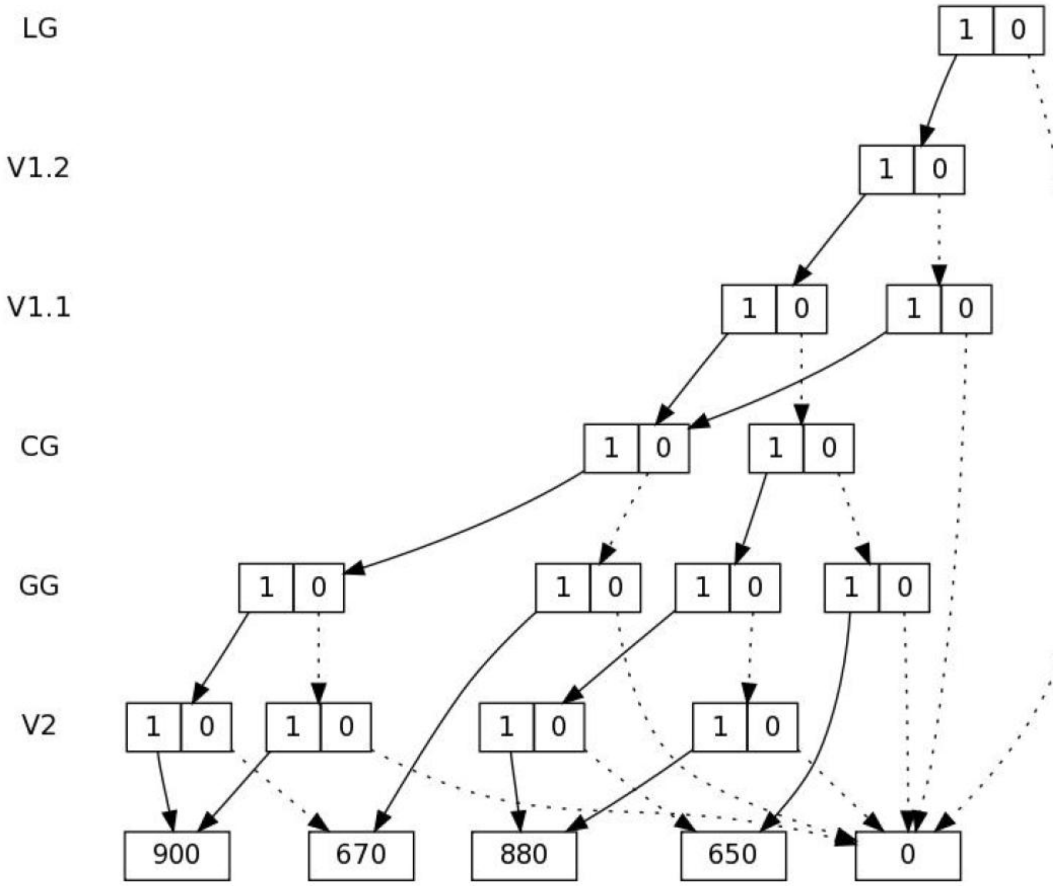


Figure 46: MTBDD of the impact function computed on the BDD of Figure

With the above measures the analysis provides a three dimensional image where the three axes are the probability, the cost and the impact, so that an attack can be launched by finding a convenient trade-off among the three mentioned parameters. But at the same time, the knowledge of the most convenient attack scenarios is the starting point to implement defense strategies.

Countermeasures and defense tree

Up to now we have studied how to model an attack. In this section we show how the knowledge of an attack allows the user to implement countermeasures. The idea is that a countermeasure hinders an attack event: it either prevents it altogether or reduces the probability that the event occurs [73,89].

A countermeasure appears in an AT as an input to an *AND* gate whose other inputs are the events that the countermeasure should inhibit [89]. The logical position and action of a countermeasure is represented in Figure 47. Since the countermeasure has an inhibition function, the attack may proceed only if the countermeasure is not effective.

This is the reason why in Figure 47 (a)) the event *C01* that represents the countermeasure is negated before its input into the *AND* gate. An equivalent representation is proposed in Figure 47 (b)) where the event *C01* is negated directly in the event box. If the probability that a countermeasure is effective is p_{C01} the probability that the countermeasure is not effective and the attack may proceed is ($1- p_{C01}$).

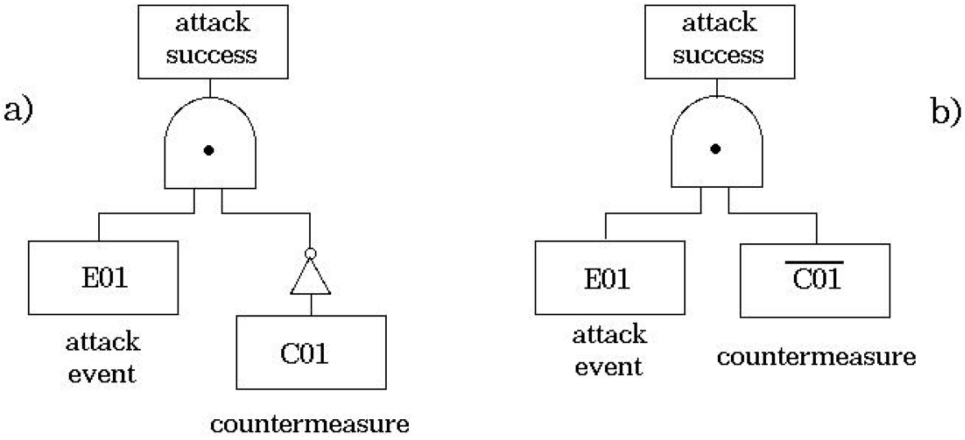


Figure 47: The action of a countermeasure

Note that for any single attack event we may have multiple countermeasures that are all logically in *AND* [89]. In the case of Figure 43 possible countermeasure are the following:

> *Event V 1:1* - enforce password protection (pwd of 8 alphanumeric characters, pwd age of 3 months);
> *Event V 1:2* - eliminate factory default password;
> *Event LG* - eliminate guest account, implement biometrics for authentication;
> *Event V 2* - Administrative rights to limited users;
> *Event CR* - network analyzers and intrusion detection mechanisms;
> *Event CG* - enforce password protection.

In a quantitative analysis of the ADT with countermeasures, a probability of success of the countermeasure should be assigned together with the cost of implementing the countermeasure.

8.3 Attack and defence tree applied to SCADA

Assuming as SCADA architecture the general ones shown in Figure 48, for which some more details are provided in [61], the characterizing elements are:

- The SCADA Control Centre (SCC) has a complete redundant backup.
- The primary LAN connects the SCC to different services and facilities like Web Server to customers and the central data base;
- SCC is connected to the RTU by means of an MTU and a network that in our specific case is composed by a proprietary WAN with a backup connection through a public corporate network (Figure 48).

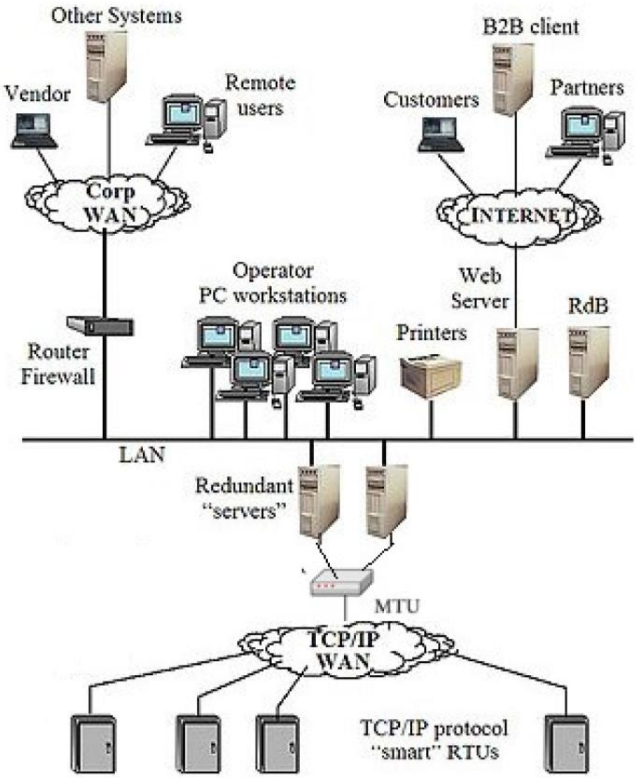


Figure 48: Typical SCADA architecture

On the base of the suggestions and analysis provided in [7,73,90,89], we assume as a ROOT of the AT, the event SCADA compromised (Gate G1) Figure 50. The attack may penetrate along three main lines:

- The RTUs, the MTU and the network that connects the RTUs to the MTU. The Master Terminal Unit (MTU) stores and processes the information from RTU (Remote Terminal Unit). The network, as in Figure 49, is composed by a proprietary private WAN with a redundant connection through a public corporate network (events E01 - E06).

- The second line of attack is through the primary control center (composed by two main blocks a SCC and a HMI) and its backup (composed by a switch and the backup SCC and HMI) (events E07 - E11)
- The third line of attack targets the central LAN and the equipment and facilities connected to the LAN, like the hystorian Data Base, and the Web service to the customers (events E12 - E15)

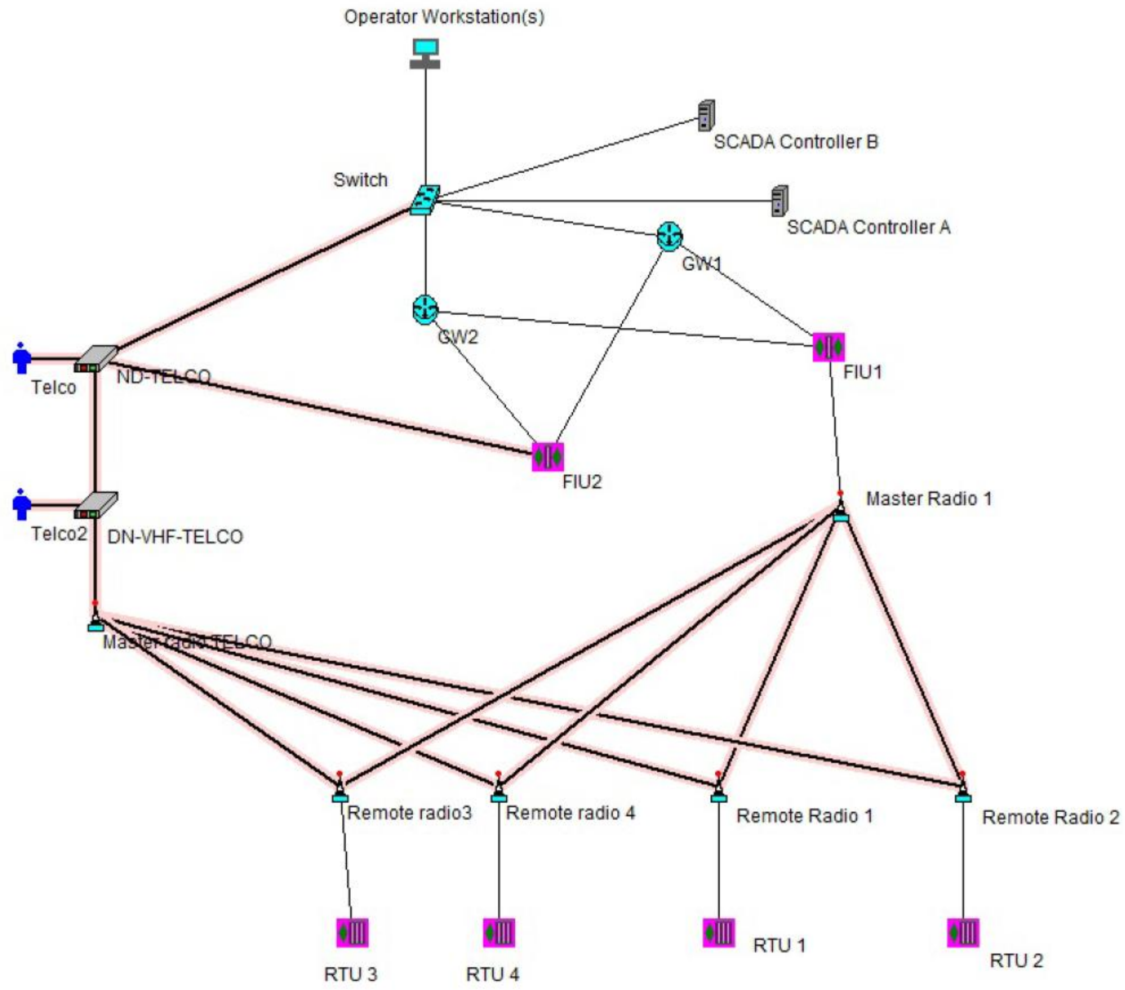


Figure 49: Layout of the network connecting the SCC to the RTU

With the above organization, AT of Figure 50 has 14 basic attack leaves (events E1 - E14), 8 intermediate gates (gates G2 - G9) and one ROOT (gate G1).

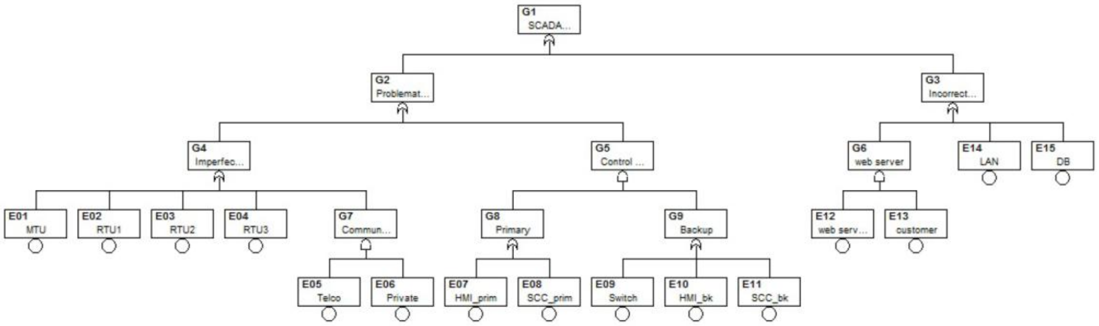


Figure 50: AT of the SCADA architecture

The basic events are assigned an attack rate (assuming that the time to attack is a random variable exponentially distributed), a cost and an impact according to the values reported in Table 20.

On the base of the attack rate data, reported in the third column of Table 20, we can compute the probability vs time of reaching the ROOT of the AT in Figure 50. The attack probability is computed by converting the Boolean structure of the AT into a BDD and performing the probability computations on the BDD, by applying iteratively equation (6). The Attack Probability as a function of time (in hours) is displayed in Figure 51.

Furthermore, by applying equation (7) we have computed the Birnbaum importance index whose value is reported in the last column of Table 20. The Birnbaum index ranks the AL according to their importance that takes into account both the probability and the position in the AT. Looking at the values, we see that the RTUs have the highest importance index.

We can integrate the attack probability with the cost of performing an attack. To show the results that can be obtained we fix a (rather arbitrary) mission time that however could be changed and parameterized. In the present case we have fixed a value TM = 1800 h (corresponding to 75 days). By attaching to each leaf of the AT of Figure 50 the attack cost reported in the fourth column of Table 20 we can convert the AT into a weighted AT, such that the path to reach the ROOT is now associated with a probability and an attack cost.

The results are reported in Table 21, where the columns have the following meaning:

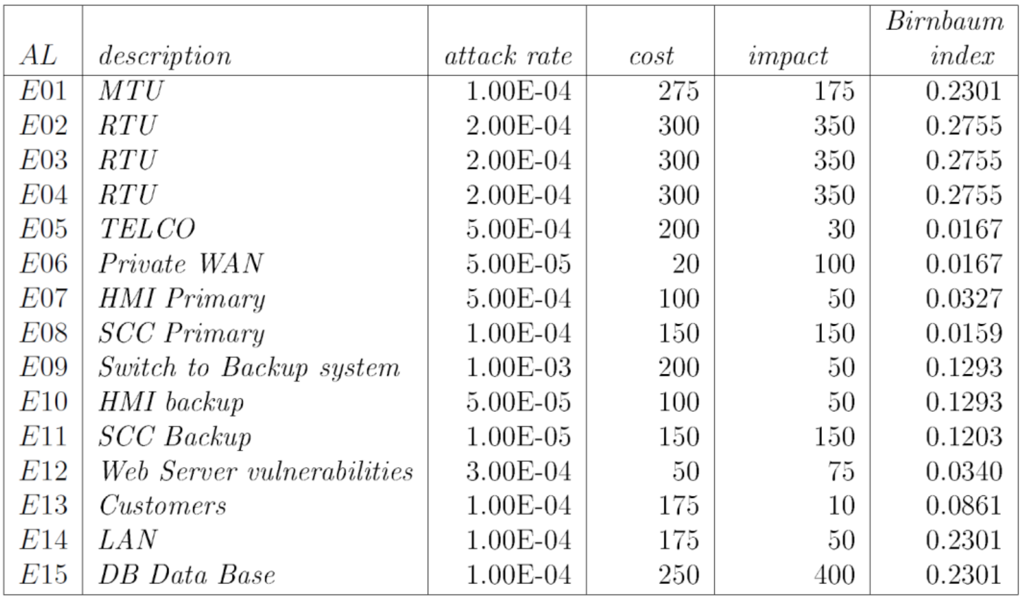
1. The first column reports the possible costs that can be incurred for a successful attack to reach the ROOT. According to the results of our analysis, the minimum cost of a successful attack is 200 cost units and the maximum is 375.

2. The second column reports the mass probability of reaching the root with an attack of the corresponding cost. The first row, corresponding to an attack cost equal to 0 represents the probability that the ROOT is not reached (the attack is not successful) in the fixed time span of TM = 1800 h. The second row means that we have a probability of 0,051076 of successfully reaching the ROOT at a cost of 200 cost unit. And so on.

3. The third column gives the cumulative probability, i.e. the probability of successfully reaching the ROOT at a cost less than the value reported in the first column. The cumulative probability is simply obtained by progressively summing up the mass probabilities in the second column. Note that the cumulative probability is defective

(does not reach the value 1) since there is a non null probability that the attack is not performed

Table 20: Description, attack rate, cost, impact and Birnbaum index for the attack leaves of Figure 50.



AL	*description*	*attack rate*	*cost*	*impact*	*Birnbaum index*
*E*01	*MTU*	1.00E-04	275	175	0.2301
*E*02	*RTU*	2.00E-04	300	350	0.2755
*E*03	*RTU*	2.00E-04	300	350	0.2755
*E*04	*RTU*	2.00E-04	300	350	0.2755
*E*05	*TELCO*	5.00E-04	200	30	0.0167
*E*06	*Private WAN*	5.00E-05	20	100	0.0167
*E*07	*HMI Primary*	5.00E-04	100	50	0.0327
*E*08	*SCC Primary*	1.00E-04	150	150	0.0159
*E*09	*Switch to Backup system*	1.00E-03	200	50	0.1293
*E*10	*HMI backup*	5.00E-05	100	50	0.1293
*E*11	*SCC Backup*	1.00E-05	150	150	0.1203
*E*12	*Web Server vulnerabilities*	3.00E-04	50	75	0.0340
*E*13	*Customers*	1.00E-04	175	10	0.0861
*E*14	*LAN*	1.00E-04	175	50	0.2301
*E*15	*DB Data Base*	1.00E-04	250	400	0.2301

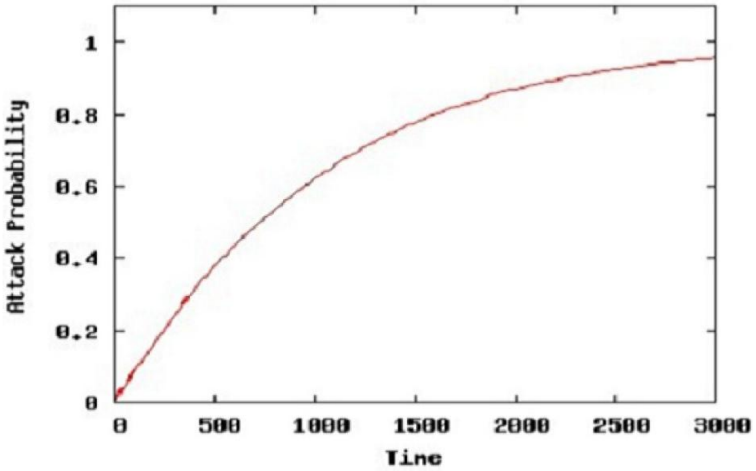


Figure 51: Attack probability vs time (in h)

If we look at the fourth row of Table 21 we find an attack cost of 250, a mass probability 0,157582 and a cumulative probability 0,273881. These figures mean that we have a probability equal to 0,157582 of reaching the ROOT spending exactly a cost of 250 and a cumulative probability equal to 0,273881 of reaching the ROOT with a cost less or equal to 250. A similar analysis can be carried out by looking at the attack impacts. In this case, the attacker is interested in procuring the maximum damage and hence, in the presence of an OR gate we pick up the maximum impact among the input events.

Table 21: Attack cost vs mass and cumulative probability

cost	*mass probability*	*cumulative probability*
0	0.161230	-
200	0.051076	0.051076
225	0.065223	0.116299
250	0.157582	0.273881
275	0.219523	0.493404
300	0.343209	0.836613
350	0.000954	0.837567
375	0.001203	0.838770

Following these rules we can construct the MTBDD for the impacts and the results are reported in Table 22. The meaning of the columns are the same as for the cost function, but we have Table 22: Attack impact vs mass, cumulative and survival probability added one more column indicating the survival probability (defined as the defective value minus the cumulative probability). The mass probability in the second column indicates the probability of having the impact reported in the first column, while the cumulative probability in the third column indicates the probability of having an impact less or equal to the value of the first column. The survival probability indicates the probability of having a successful attack with an impact greater than the corresponding value on the first column.

Table 22: Attack impact vs mass, cumulative and survival probability

impact	*mass probability*	*cumulative probability*	*survival probability*
0	0.161230	-	
80	0.001203	0.001203	0.837567
85	0.011989	0.013192	0.825578
100	0.015871	0.029063	0.809707
175	0.082459	0.111522	0.727248
200	0.010068	0.121590	0.717180
300	0.123749	0.245339	0.593431
350	0.428701	0.674040	0.164730
400	0.164730	0.838770	0.0

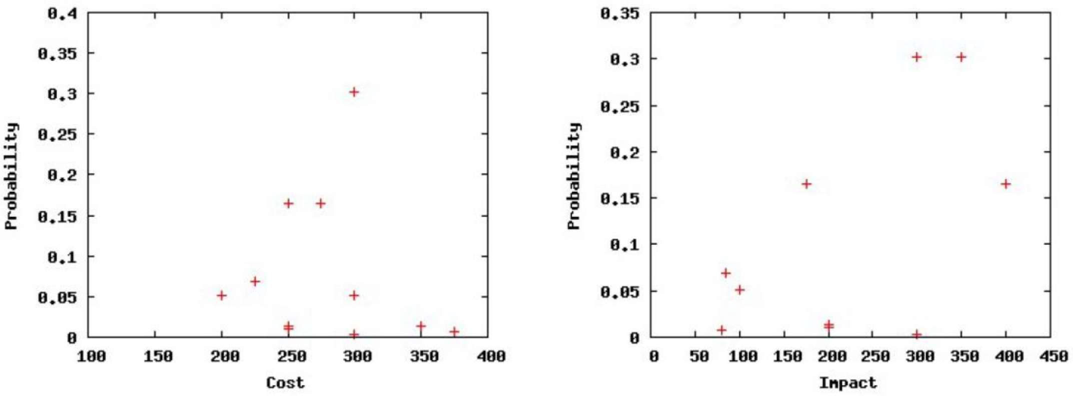
Table 21 and Table 22 show which are the probabilities of success of an attack with a given cost or a given impact, but do not indicate which are the strategies to follow to successfully perform an attack with a given cost or with a given impact.

Table 23: MCS, cost, impact and probability

mcs number	*AL forming mcs*	*impact*	*cost*	*probability*
mcs_1	E15	400	250	0.164730
mcs_2	E14	175	275	0.164730
mcs_3	E12 E13	85	225	0.068734
mcs_4	E09 E07	100	300	0.051076
mcs_5	E11 E07	200	250	0.010586
mcs_6	E10 E07	100	200	0.051076
mcs_7	E03	350	300	0.302324
mcs_8	E04	300	300	0.302324
mcs_9	E02	350	300	0.302324
mcs_{10}	E01	175	275	0.164730
mcs_{11}	E09 E08	200	350	0.014178
mcs_{12}	E11 E08	300	300	0.002939
mcs_{13}	E10 E08	200	250	0.014178
mcs_{14}	E05 E06	80	375	0.007408

We can further detail our study by analyzing the *mcs* one by one with their cost and their impacts. The results are reported in Table 23, where all the mcs are tabulated together with their cost, impact and probability of occurrence.

The results of Table 22 are also displayed in Figure 52 (a) for what concerns the cost and in Figure 52 (b) for what concerns the impact. The Figures help choosing the suitable strategy by trading off between high probability and low cost and high probability and high impact.



a) b)

Figure 52: *mcs* probability vs cost a) and impact b)

SCADA system with countermeasures

The qualitative and quantitative analysis performed in the previous paragraph give a preliminary information on which are the weakest points in the architecture and, hence, which are convenient directions to follow to reinforce the system. In particular, the mcs formed by a single AL indicate that an attack in a single point may arrive to the root, while the mcs with two (or more) ALs indicate that the attack needs to jointly start from two (or more) attack events.

The security of the SCADA system of the case study has been already analyzed utilizing the methodology and tools offered by the National Cyber Security Division of the US Department of Homeland Security by means of the Cyber Security Evaluation Tool (CSET) [91]. The tool CSET, applied in particular to the portion of the SCADA system of Figure 49, evidenced various vulnerabilities in the system and indicated various possible countermeasures to mitigate the vulnerabilities.

Inspired by CSET [91], we propose to evaluate the set of countermeasures displayed in Table 24. In Table 24 the acronym IDS/IPS means Intrusion Detection System and Intrusion Prevention System. The countermeasures are applied to the original AT of Figure 50 by means of the structure of Figure 47 (b). The countermeasure events are negated in the event box, and the probability reported in the fourth column of Table 24 is the failure probability, i.e. the probability that the countermeasure is not effective and the attack may proceed.

Table 24: Attack events from Table 20 with the implemented countermeasures and the related probability of failure

AL	*description*	*Countermeasure description*	*Failure Probability*
*E*01	*MTU*	*IDS/IPS*	0.2
*E*02	*RTU*	*Firewall*	0.3
*E*03	*RTU*	*Firewall*	0.3
*E*04	*RTU*	*Firewall*	0.3
*E*05	*TELCO*	*IDS/IPS*	0.2
*E*06	*Private WAN*	*IDS/IPS*	0.2
*E*07	*HMI Primary*	*Eliminate Guest Account*	0.5
		Implement Password Age	0.4
*E*08	*SCC Primary*	*Eliminate Factory Default Password*	0.3
*E*09	*Switch Backup system*		
*E*10	*HMI backup*	*Eliminate Guest Account*	0.5
		Implement Password Age	0.4
*E*11	*SCC Backup*	*Eliminate Factory Default Password*	0.3
*E*12	*Web Server vulner*	*Implement Digital Certificates*	0.5
*E*13	*Customers*	*Implement Biometric Authetication*	0.3
*E*14	*LAN*	*IDS/IPS*	0.2
*E*15	*DB Data Base*	*DMZ (Demilitarized Zone)*	0.2

The result of this application is reported in three Attack Subtrees starting from the gates G3, G4, G5 of the original AT of Figure 50. The subtree originated from gate G3 is reported in Figure 53. The subtree originated from gate G4 is reported in Figure 54. The subtree originated from gate G5 is reported in Figure 55. The probability of reaching a successful attack for the AT without countermeasures and the AT with countermeasures is compared in Figure 56, where the mitigation effect of the countermeasures is evidenced.

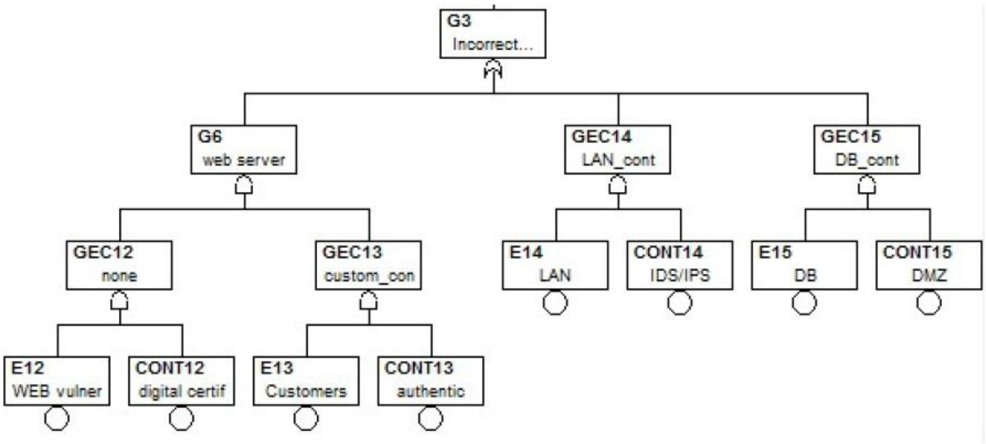


Figure 53: Subtree with countermeasures originated by Gate G3

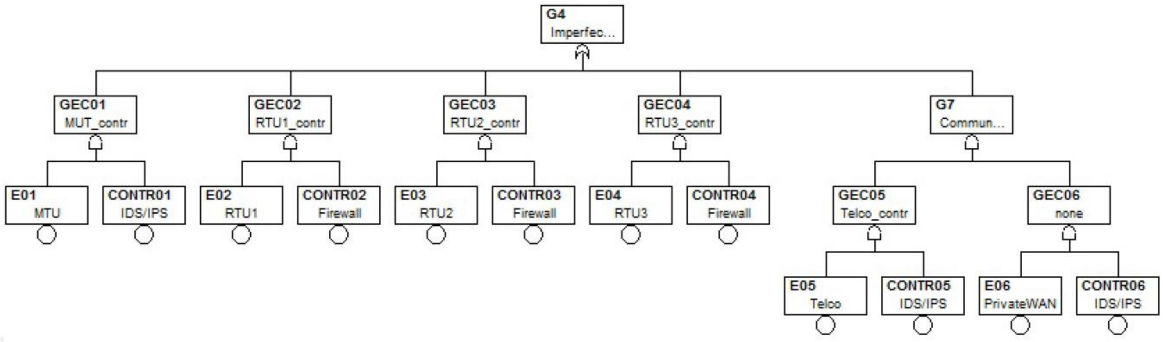


Figure 54: Subtree with countermeasures originated by Gate G4

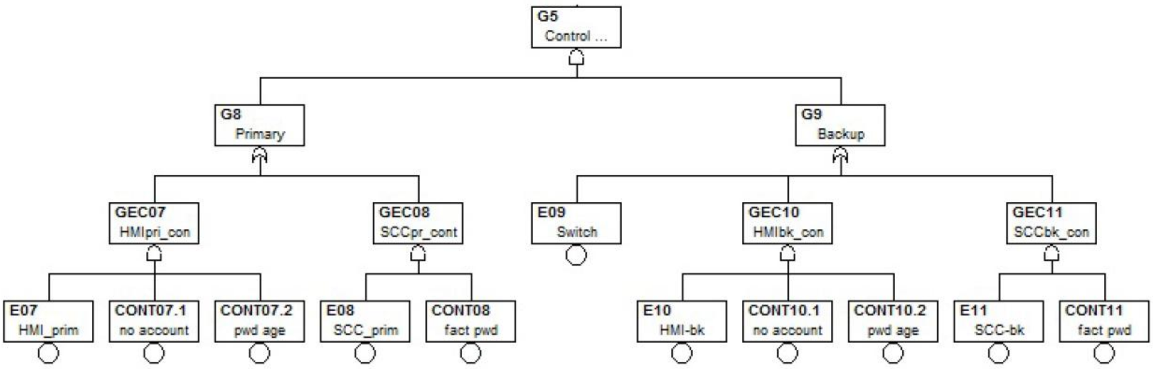


Figure 55: Subtree with countermeasures originated by Gate G5

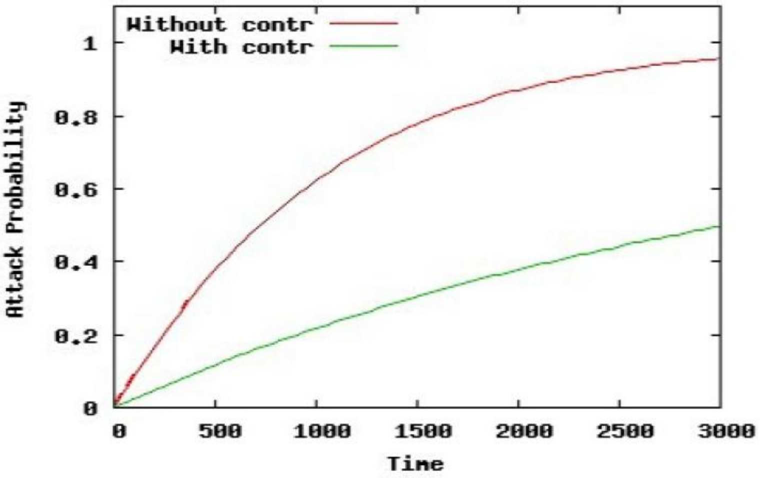


Figure 56: Comparison of AT probability vs time (in h) with and without countermeasures

9 Modelling versus Test Bed

While modelling is in charge of predicting consequences of cyber attacks on SCADA and the electrical grid, the test bed is in charge to reproduce cyber attacks and their propagation more realistically than modelling.

Performing security test on SCADA system consists of going to implement, in our case, three different cyber attacks above described. Conducting cyber attacks on an actual SCADA system is unthinkable because this action could cause even catastrophic failures. For this reason, research on SCADA security employs Test Bed (TB) to implement specific scenarios (Malware propagation, DoS, MITM and so on).

Specifically, to demonstrate CockpitCI project results the following HTB has been constructed (Figure 57) in IEC.

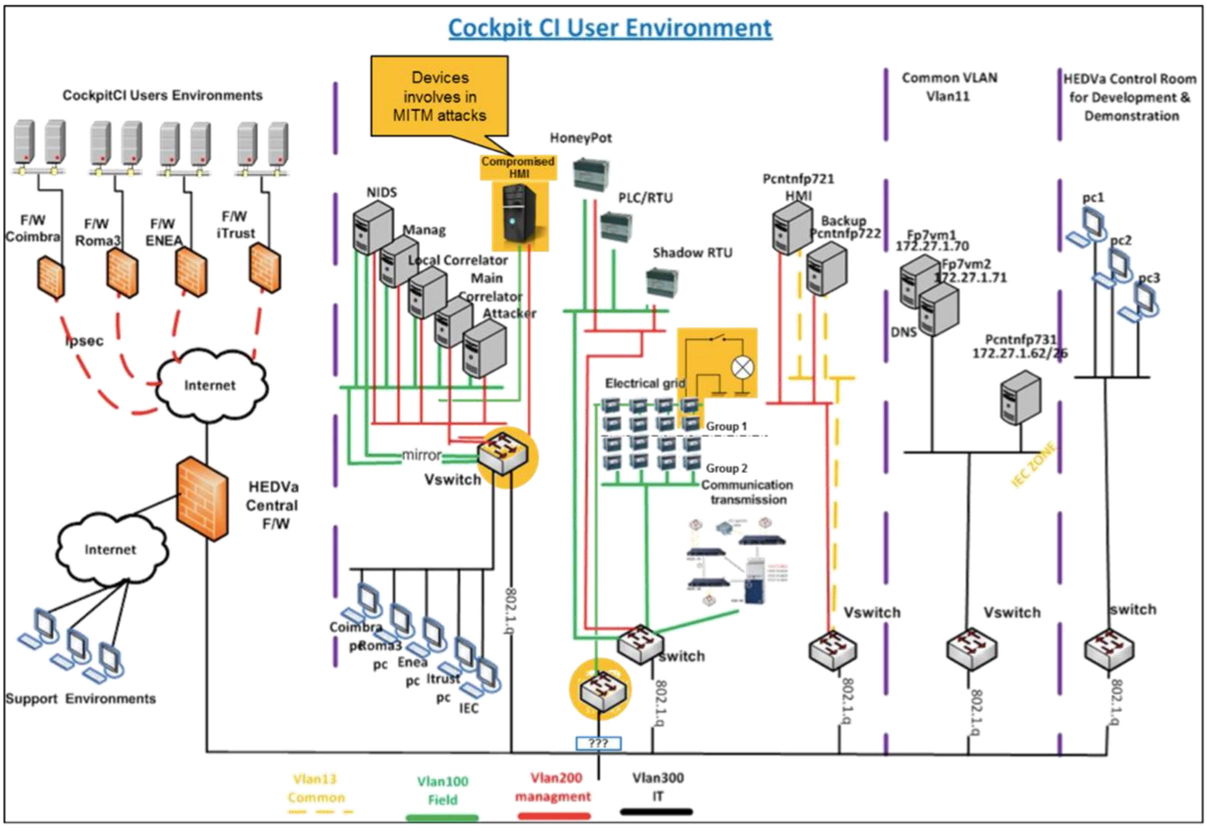


Figure 57: CockpitCI HTB

The hybrid test bed is constituted by the coexistence of actual and simulated systems and devices of SCADA, corporate network and the electrical grid. Actual SCADA devices consists of SCADA HMI and SCADA control server, which barely implements the FISR procedure. The circuit breakers operations (controlled by SCADA RTUs) are implemented by means of actual Programmable Logical Controller (PLC) devices. PLCs are connected, from one side, to the HMI and, from the other side, to the electrical breakers by actual standard connections. Actual devices allow to verify the effectiveness of actual cyber attacks.

The HTB is a distributed environment that provides the possibility for parallel operation of different users. It is constituted by the coexistence of actual and simulated systems and devices of SCADA, corporate network and electrical grid.

It provides the following capabilities:

- Simulate operation scenarios (power grid and telecom) based on real SCADA and Network Management System (NMS), physical components of electrical and telecom infrastructure and simulated elements of electrical and telecom infrastructure,
- Collect and analyse real network traffic of heterogeneous networks (power grid, telecom network, SCADA),
- Test models and components for cyber-attack detection and identification,
- Test models and components for mitigation of cyber-attack influence on critical infrastructure,
- Simulate cyber-attacks on different parts of CI,
- Identify and test vulnerable parts of CI with weak physical security and accessible to unauthorized people,
- Test effectiveness of countermeasure's plans,
- Test effectiveness of automatic reaction logics,
- Test CockpitCI system functionality,
- Define users with predefined access levels.

9.1 ENEA Remote Test Beds

Geographic dispersion and multi-site topologies are nowadays common in modern CIs. In this perspective, it was considered to be useful to somehow replicate a geographically dispersed topology involving multiple autonomous sites, both for research and testing purposes, allowing for interaction of SCADA with remote testers of reference scenario vulnerabilities and CockpitCI solution. Instead of building such environment from zero, it was considered an alternative approach of interconnecting involved partners using Virtual Private Networking (VPN) technologies, in a secure and effective fashion.

The use of VPN technologies to interconnect existing research and stakeholder Test Beds has the benefit of providing a cost-effective approach for interoperability testing, remote security assessment and requirement validation, while providing a good measure of the effectiveness of VPN technologies in CI scenarios, both in terms of security, protocol and functional impact and latency overhead (which is critical for real-time systems).

The reference VPN interconnection scenario include the responsible entities needed to establish and validate the requirements of the stakeholders, namely IEC (Israel Electrical Corporation) which has offered to provide access to its SCADA testbed. In an initial stage, it was agreed that the laboratory test beds at ENEA, Roma3, UC (University of Coimbra) and IEC would be safely interconnected in a common Layer-3 domain with the same IP addressing space, using secure VPN technologies for LAN-to-LAN communication. ??? illustrates the proposed topology:

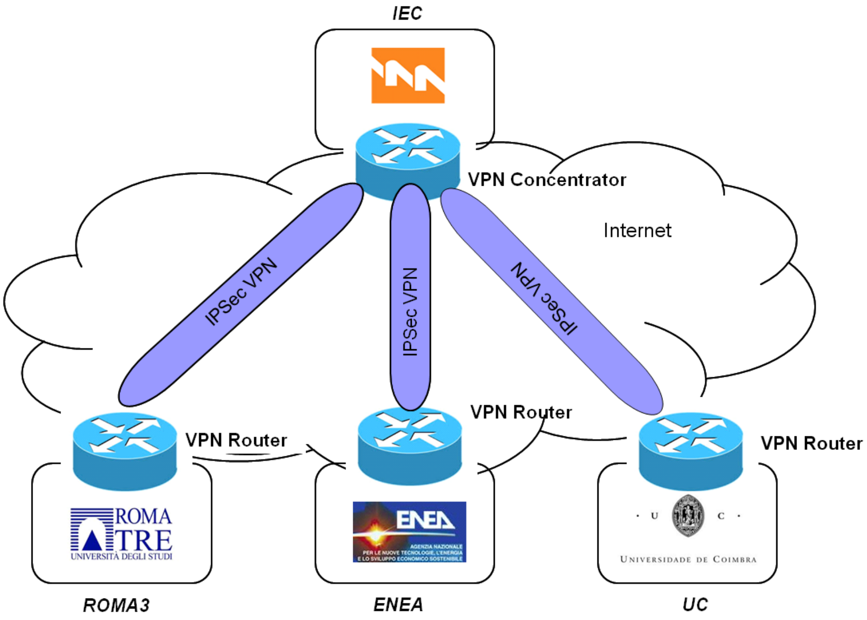


Figure 58: Logical VPN Topology for Testbed Aggregation

In this scenario, IPSec [93] or purpose-tuned SSL-based [94] VPN technologies are the best candidates for establishing the proposed VPN topology, because of its advantage in terms of communication latency over other alternatives such as PPTP [95] or OpenVPN [96], which commonly rely on userspace implementations (and therefore, more affected by factors such as context switching).

A VPN concentrator, placed in the IEC testbed network will be responsible for bridging the tunnels from the VPN routers on ENEA, UC and Roma3 with the physical network, therefore constituting a topology aggregating together all the networks behind each device (router or concentrator) in a single Layer-3 entity. The VPN concentrator can also be optionally configured to enable secure and authenticated permanent and ad-hoc accesses from isolated hosts or other networks.

This configuration must be fine-tuned to enable an adequate balance between security (use of low-overhead encryption mechanisms and reduced encryption key life) and latency overhead however, it is possible that some fine-tuning will be required (i.e., timeouts) to enable adequate testbed component interoperability across VPN links.

Both the VPN concentrator and the routers can be built using commodity hardware (a PC with two network interfaces) and software (a common Linux distribution has all the required means to implement this solution). In either case (VPN concentrator or VPN router), one physical network interface is to be connected to the physical network segment that is going to be bridged and the other one must be connected to the Internet, preferably with a public IP – as an alternative, if this interface is to be connected to a network that is behind a routing firewall, and IPSec is used, NAT-T (NAT Traversal [97]) must be enabled and configured accordingly.

As far as it concerns ENEA remote Test Bed, its functionality is threefold:

1. To enter the actual devices of the IEC HTB;

2. To locally simulate a subset of SCADA devices such as SCADA Control Centre and RTU.

3. To conduct cyber attacks and analyze their consequences on the electrical grid.

The VPN connection between ENEA and IEC foreseen two different solutions:

1. Commercial Solution by checkpoint (see Figure 59);

2. Open source solution by Coimbra Virtual Machine (VM) (see Figure 60).

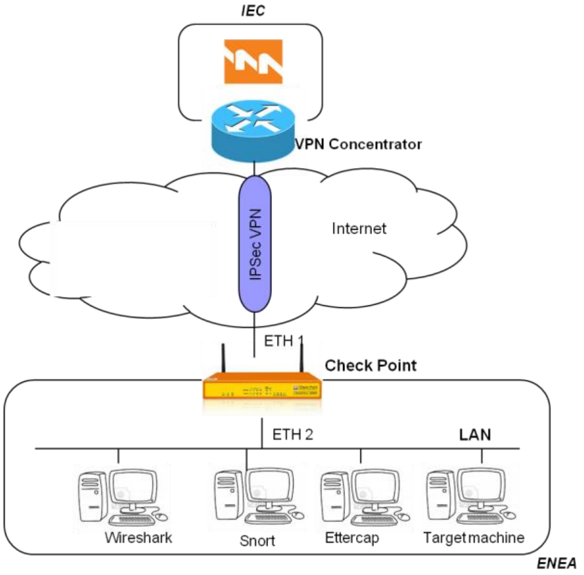


Figure 59: Commercial solution

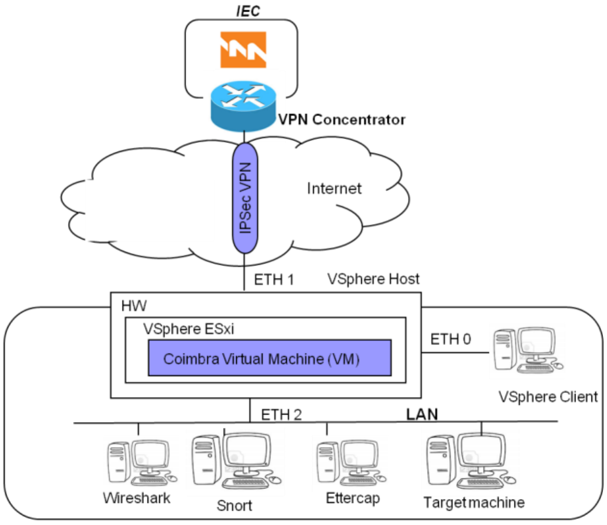


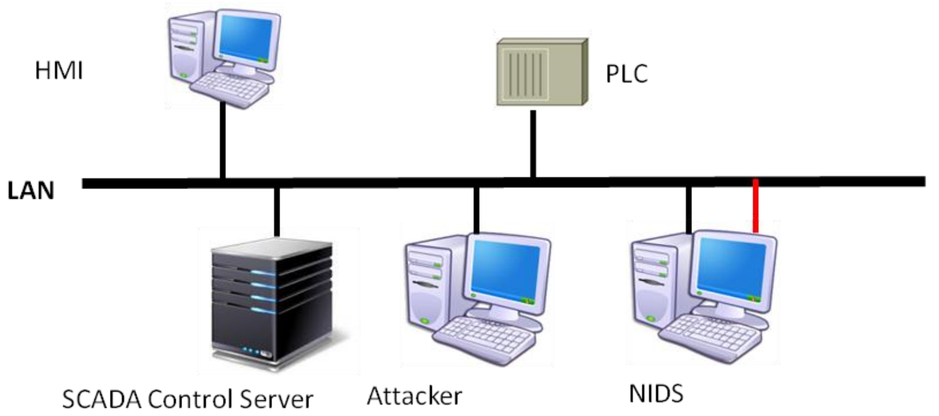
Figure 60: Open Source solution

The first represent the more immediate and simple solution because in order to establish a security VPN will be sufficient buy an hardware device named Check Point. In this device firewall, antivirus, antispam, VPN site-to-site and so on functionality are implemented.

Instead, the second solution is longer because in order to establish a security VPN there is the need to install a virtualization platform and not only.

9.1.1 Test Bed architecture

ENEA Test Bed [99] is based on a switched Local Area Network (LAN) which implements an Ethernet communication between SCADA devices, attacker and Network Intrusion Detection System (NIDS), as in Figure 61.



.

Figure 61: ENEA remote Test Bed Architecture

SCADA devices are the SCADA Human Machine Interface (HMI), the SCADA Control Server and the Modicon 340 PLC.

SCADA HMI is a user graphical interface to monitor electrical grid status (e.g. visualize alarms caused by faults of the electrical grid) and to control, in real time, field equipments, like the PLC/RTU, throughout the SCADA Control Server.

The aim of the SCADA Control Server is threefold:

i) To communicate directly with PLC/RTU;

ii) To collect data from them

iii) To transfer monitoring and control information to HMI.

PLC/RTU is a local processor on the field, which collects electrical grid status (e.g. voltage, current, power and breakers status) from a large amount of sensors and transmits control commands from SCADA control server to field actuators (e.g. electrical breakers). The communication protocol between SCADA control server and the PLC is Modbus over TCP/IP.

A dedicated machine is used to conduct network attacks, equipped with Kali Linux distribution that includes different ready to use attack tools, such as Ettercap. Ettercap has been used to conduct MITM ARP (Address Resolution Protocol) Poisoning attacks as described in the following section.

A Network Intrusion Detection System (NIDS) is used to detect the cyber attacks on the HTB. NIDS is an open source tool named Snort [100], that is supplied by Security Onion. Security Onion [101] is a Linux distribution for intrusion detection system, network security monitoring and log management. Snort is a tool, which analyzes the real time network traffic and it is composed by:

- One or more probes, for monitoring;
- A server that receives the information collected by the probes;
- A management workstation: an interface between IDS and administrator. Particularly, in our HTB, probes, server and management workstation are collapsed in a single machine.

9.1.2 MITM cyber attacks on SCADA by means of ARP poisoning

Remote Test Bed is used to implement actual MITM cyber attacks on SCADA devices, with the aim of compromising the status of its controlled electrical grid. That is performed in an incremental way, starting from compromising the communication between SCADA Control Server and a field device and between SCADA Control Server and SCADA HMI.

For IP communication between two devices over Ethernet, the logical IP address of the destination device must be mapped into its physical Media Access Control (MAC) address. Address Resolution Protocol (ARP) maps the IP logical address to MAC address and this mapping is cached in local ARP cache.

Two devices (e.g. SCADA Control Server and PLC) to communicate by means of LAN infrastructure, need to know the MAC address of their respective peer. Hence, they exchange ARP messages to map the logical IP address and the MAC address and build their ARP cache. Authentication and encryption features in the ARP protocol are missing and so the protocol is not secure. This causes the following consequences:

- Updating the cache even if no updating request has been sent;
- Overwriting the cache entries silently.

Hence, an attacker may operate on ARP cache to manipulate the communication between the two devices.

MITM cyber attack, by ARP Poisoning [99], consists in poisoning of ARP cache. Once ARP cache has been successfully poisoned, the traffic that devices exchange will go through the attacker. In this case the attacker is in the middle of the communication between the two devices and it can easily monitor all communication. The goal of this MITM cyber attack is to intercept, view and alter the content of the packets exchanged.

MITM cyber attack by ARP Poisoning have been conducted in two significant segments of SCADA system.

A MITM cyber attack by means ARP Poisoning, using Ettercap tool, conducted against SCADA Control Server and PLC is shown in Figure 62. The attacker captures all the traffic between the PLC and SCADA Control Server and then changes the value of the exchanged packet contents.

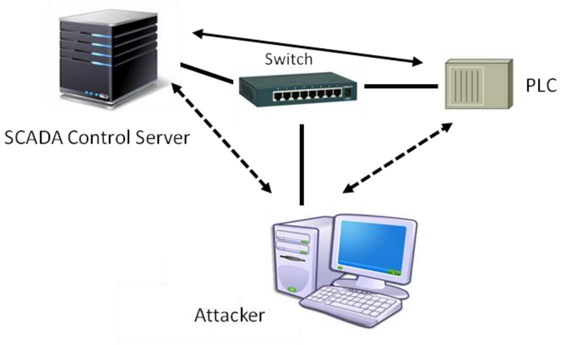


Figure 62: MITM cyber attack: Alter response contents

A MITM cyber attack by means of ARP Poisoning, using Ettercap tool, conducted against the communication among SCADA HMI and SCADA Control Server, is shown in Figure 63.

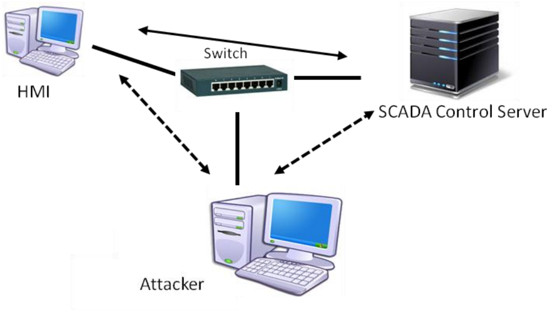


Figure 63: MITM cyber attack: Anomalous Data

The attacker intercepts status information of the electrical grid that the SCADA Control Server supplies to SCADA HMI. On the basis of the captured data, the attacker may send a false alarm to the SCADA operator, displaying it on SCADA HMI. On such a false alarm, SCADA operator may send inappropriate commands to the electrical grid that may alter the grid status: i.e. he may disconnect a large number of electrical customers or may damage critical electrical devices such as the AV/MV transformer.

10 Short discussion and conclusions

Modelling and prediction of Quality of Service indicators of the interdependent Systems of Systems (MV electrical grid, its SCADA and the corporate network) under cyber attacks is discussed within the document.

It is assumed that at the state of the art, no single modelling technique has the credible modelling power and the analytical tractability to adequately deal with the Quality of Service (QoS) of such Systems of Systems.

Different heterogeneous modelling approaches have been extensively investigated and discussed, models of SCADA, corporate network and electrical grid under different cyber attack cases have been built and SCADA and electrical grid QoS has been predicted. The final aim was to provide knowledge, approaches, partial algorithms or whole models and QoS indicators for the CockpitCI tool. The tool is composed by a cyber detection layer and a risk prediction layer, which ideally intend to support, even in near real time, SCADA and CERT operators.

Considering that the risk prediction layer of CockpitCI tool was to be based on CISIA tool, at this stage, we can wrap up that modelling results more useful for CockpitCI tool are as follows:

- Regarding modelling approaches, algorithms and models, it seems that the most worthwile ones are:

  i) The SIR model of epidemics: it may be used in cyber security to study how a malware infection spreads among different machines. SIR model represents a disease spread where individuals are susceptible to a disease, potentially contract the disease, recover and become immune to future infections after recovery. In order to compute the injection and spreading of malware within corporate network and SCADA, SIR models are implemented via the open source tool Netlogo.

  ii) Agent based simulation, with the support of the Intelligent RAO simulator intends to address high-level, inter-system behaviour simulation. As shown, the simulation model developed using the Intelligent RAO Simulator on the basis of modelling framework presented in D2.2, is capable to reproduce both parts of CockpitCI reference scenario - FISR and cyber attacks, including dynamic attacks, still in progress while executing FISR. The QoS gives the potential damage, giving an indispensible data for risk estimation. Further work is undergoing to adapt the model to Monte-Carlo simulations. In this case, many simulation runs should be executed, randomly sampling the values of state rankings for concerned elements, and this for static as well as dynamic attacks.

  iii) The other modelling approaches are not less relevant from a scientific point of view, such as the attack and defence trees, temporal network reliability analysis. Simply such approaches seem to be far away for the preselected solution of CockpitCI tool. Also models implemented by NS2 to represent DoS and MITM attacks and their consequences on quality of FISR service indicators and, in turn, on quality of power to grid customers, even if are considered very relevant in terms of knowledge data, do not seem to provide specific algorithms for the CockpitCI tool. In fact such models provide a very detailed level of representation of the ICT based world, that cannot be represented by CISIA approach.

- Regarding QoS indicators of SCADA and electrical grid, it seems that the most worthwhile ones are:

  i) SCADA QoS indicators

     a. *Loss of View (LoV),* if the SCADA Control Centre can't receive packets from the RTUs:

     b. *Loss of Control (LoC) per cent*, if the RTUs can't receive packets from the SCADA Control Centre:

     c. Time Response of SCADA in executing FISR procedure

     d. Packets routing

  ii) Electrical grid QoS indicators

     a. The duration of electrical interruptions for customer for year

     b. The number of long/short electrical interruptions for customer per year

     c. SAIDI - System Average Interruption Duration

     d. SAIFI - System Average Frequency Interruption

     e. CAIDI - Customer Average Interruption Duration

     f. Tn = Σ(KVA*Duration)/Installed KVA. Tn is indeed an equivalent time of complete loss of electricity for all the customers while executing FISR.

- Regarding cyber attack cases: all the attack cases investigated in the document seems to be adequate, with a special attention to MITM attacks and malware propagation.

The deliverable also focuses on the ideal integration of modelling and test beds. While modelling is in charge of predicting consequences of cyber attacks on SCADA and the electrical grid, the test bed is in charge to reproduce cyber attacks and their propagation more realistically than modelling. The last part of the document discusses the integration between test bed and modelling. Models hardly rely on the assumptions which characterize the actual world, including SCADA and ICT technology world. More over cyber security is a very complex and dynamic argument, far to be well understood and completely captured by modelling. Laboratory activities may help in better representing the actual world, understanding the value of model parameters, validating models and improve their adherence to the actual world. A test bed laboratory, with the heart at the Israel Electric Corporation and remote terminals distributed among the other partners of the project, has been realized. Particularly, at ENEA, a test bed laboratory is going to be realized where cyber attacks can be reproduced on a simple mock up of SCADA, and parameters analyzed.

11 References

1. S. Rinaldi, J. Peerenboom, and T. Kelly. Identify, understanding, and analyzing critical infrastructure interdependencies. IEEE Control System Magazine, Dec:11–25, 2001.
2. W. Kroeger. Critical infrastructures at risk: a need for a new conceptual approach and extended analytical tools. Reliability Engineering and System Safety, 93:1781–1787, 2008.
3. P. Pederson, D. Dudenhoeffer, S. Hartley, and M. Permann. Critical infrastructure interdependency modeling: A survey of US and international research. Technical report, Idaho National Laboratories, INL-EXT/06/11464, 2006.
4. GAO General Accounting Office. Critical infrastructure protection: Challenges and effort to secure control systems. Technical report, Report-04-354, March 2004.
5. McAfee. In the dark (Second annual critical infrastructure protection report). Technical report, CSIS - Center for Strategical International Studies, 2011.
6. NIST. Managing information security risk. Technical report, NIST Special Publication 800-39, March 2011.
7. E.J. Byres, M. Franz, and D. Miller. The use of attack trees in assessing vulnerabilities in SCADA systems. In International Infrastructure Survivability Workshop (IISW’04), 2004.
8. K. Stouffer, J. Falco, and K. Scarfone. Guide to Industrial Control System (ICS) security. Technical report, NIST Special Publication 800-82, June 2011.
9. CockpitCI deliverable 2.1- Overview of modelling techniques and tools for SCADA systems under cyber attacks
10. A. Avizienis, J. C. Laprie, Brian Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing, 1:11–33, January-March 2004.
11. F. Cohen, Managing Network Security - Attack and Defense Strategies. Network Security Magazine, July 1999.
12. Karin Sallhammar, Bjarne E. Helvik and Svein J. Knapskog. On Stochastic Modeling for Integrated Security and Dependability Evaluation. The Journal of Networks (ISSN 1796-2056), Vol. 1, No. 5, September/October 2006.
13. Q. Wu, S. Shiva, S. Roy, C. Ellis, and V. Datla. On Modeling and Simulation of Game Theory-based Defense Mechanisms against DoS and DDoS Attacks.SpringSim 2010.
14. J. Jormakka and J. V. E. Molsa. Modelling information warfare as a game. Journal of Information Warfare; Vol. 4(2), 2005.
15. P. Liu, W. Zang, and M. Yu. Incentive-based modeling and inference of attacker intent, objectives, and strategies. ACM Transactions on Information and System Security (TISSEC), 8(I):78-I 18, 2005.

16. C. Xiaolin, T. Xiaobin, Z. Yong, and X. Hongsheng. A markov game theory-based risk assessment model for network information systems. International conference on computer science and software engineering, 2008.
17. B. Schneier. Attack trees. Dr. Dobb Journal of Software Tools, 24(12):21-29, 1999.
18. Andrew P. Moore, Robert J. Ellison, and Richard C. Linger. Attack Modeling for Information Security and Survivability, 2001.
19. Bonnie Zhu, Anthony Joseph, Shankar Sastry, "*A Taxonomy of Cyber Attacks on SCADA Systems*", p380-388 IEEE Computer Society Washington DC USA 2011 ISBN 978-0-7695-4580-6.
20. Christofer Minich and Howard Ragunton. Surviving a cyber attack on your SCADA system.
21. W. Reisig, Petri Nets: an Introduction, Springer Verlag, 1985.
22. J.L. Peterson, Petri Net Theory and the Modeling of Systems, Prentice-Hall Englewood Cliffs, NJ, 1981.
23. Mattew H. Henry, Ryan M. Layer, Kevin Z. Snow, and David R. Zaret. Evaluating the risk of cyber attacks on SCADA systems via petri net analysis with application to hazardous liquid loading operations. IEEE, 2009.
24. Penet tool. http://powercyber.ece.iastate.edu/penetintro.html
25. G. Ciardo, J. Muppala, and K. Trivedi, User Manual for SPNP: Stochastic Petri Net Package.
26. Chee-Wooi Ten and Chen-Ching Liu. Vulnerability assessment of cyber security for SCADA systems. IEEE, 2008.
27. T. Tassier. SIR Model of Epidemics, Anual report, 2005.
28. E. Ciancamerla, M. Minichino, S. Palmieri - On prediction of QoS of SCADA accounting cyber attacks - Probabilistic Safety Assessment and Management Conference (PSAM11) and the Annual European Safety and Reliability Conference (ESREL 2012) - Helsinki, Finland - 25-29 June 2012
29. E. Ciancamerla, A. Di Pietro, C. Foglietta, M. Minichino, S. Palmieri, S. Panzieri- From Holistic Assessment to Impact Evaluation- in CRITIS, 7th International Conference in Critical Information Infrastructures Security, 2012
30. http://ccl.northwestern.edu/netlogo/.
31. Stouffer, Falco, Scarfone; "Guide to Industrial Control Systems (ICS) Security", NIST SP 800-82, 2011, http://csrc.nist.gov/publications/nistpubs/800-82/SP800-82-final.pdf
32. Weiss, Joseph, "Protecting Industrial Control Systems from Electronic Threats", Momentum Press, 2010, ISBN:1606501976 9781606501979.
33. Solum, Martin, "Quickdraw Retrospective, Part #1," Digital Bond, November 17, 2009, http://www.digitalbond.com/2009/11/17/quickdraw-retrospective-part-1/.
34. P. Chee-Wooi Ten, M. Govindarasu and Chen-Ching Liu. Cybersecurity for electric power control and automation systems. IEEE International Conference on System, Man and Cybernetics, 2007. ISIC, pages 29-34, 2007.

35. MICIE Project, Deliverable D2.2.1: Interdependency modelling framework, interdependency indicators and models, pp 68-70.
36. A. Artiba, V.V. Emelyanov, S.I. Iassinovski -Introduction to Intelligent Simulation: The RAO language. Kluwer Academic Publishers, 1998.
37. S. De Porcellinis, S. Panzieri, R. Setola, "Modelling critical infrastructure via a mixed holistic reductionistic approach," Int. Journal of Critical Infrastructures, Inderscience eds., vol. 5, n. 1/2, pag. 86-99
38. E. Ciancamerla, A. Di Pietro, C. Foglietta, M. Minichino, S. Palmieri, S. Panzieri- From Holistic Assessment to Impact Evaluation- in CRITIS, 7th International Conference in Critical Information Infrastructures Security, 2012.
39. E. Ciancamerla, M. Minichino, V. Rosato, G. Vicoli - SCADA systems within CI interdependency analysis: cyberattacks, resilience and quality of service - Workshop on Experimental Platforms for Interoperable Pub-lic Safety Communications - Joint Research Centre (JRC) – 10, 11 October 2011- Ispra – Italy
40. E. Ciancamerla, C. Foglietta, D. Lefevre, M. Minichino, L. Lev and Y. Shneck - Discrete event simulation of QoS of a SCADA system interconnecting a Power grid and a Telco network - 1st IFIP TC11 International Conference on Critical Information Infrastructure Protection 2010 World Computer Congress 2010 proceedings - Springer - Brisbane 2010 – ISSN 1868-4238
41. C. C. Zou, W. Gong, and D. Towsley. 2002. Code red worm propagation modeling and analysis. In Proceedings of the 9th ACM conference on Computer and communications security (CCS '02), Vijay Atluri (Ed.). ACM, New York, NY, USA, 138-147.
42. L. O'Murchu N. Falliere. W32.Stuxnet dossier, Symantec White Paper, February 2011.
43. W32.Duqu. The precursor to the next Stuxnet, Symantec White Paper, November 2011.
44. S. De Porcellinis, S. Panzieri, R. Setola, "Modelling critical infrastructure via a mixed holistic reductionistic approach," Int. Journal of Critical Infrastructures, Inderscience eds., vol. 5, n. 1/2, pag. 86-99, Inderscience, 2009.
45. S. De Porcellinis, S. Panzieri, R. Setola, G. Ulivi, "Simulation of Heterogeneous and Interdependent Critical Infrastructures," Int. Journal of Critical Infrastructures, vol. 4, n. 1/2, pag. 110-128, Inderscience Ent.. Ltd., UK, 2008
46. S. M. Rinaldi. Modeling and Simulating Critical Infrastructures and Their Interdependencies. In Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 2 - Volume 2 (HICSS '04), Vol. 2. IEEE Computer Society, Washington, DC, USA, 20054.1-, 2004.
47. M. Abrams and J. Weiss. Bellingham, washington, control system cyber security case study. http://csrc.nist.gov/groups/SMA/_sma/ics/documents/Maroochy-Water-Services-Case-Study_report.pdf, 2008.
48. R.E. Barlow and F. Proschan. Statistical Theory of Reliability and Life Testing. Holt, Rinehart and Winston, New York, 1975.

49. I. Bahar, E. Frohm, C. Gaona, G. Hachtel, E. Macii, A. Padro, and F. Somenzi. Algebraic decision diagrams and their applications. Formal Methods in System Design, 10(2-3):171_206, 1997.

50. M.O. Ball. Computational complexity of network reliability analysis: An overview. IEEE Trans on Reliability, R-35:230–239, 1986.

51. A. Bobbio, R. Terruggia, E. Ciancamerla, and M. Minichino. Reliability analysis of multi-source multi-sink critical interacting systems. In 3rd IFAC Workshop on Dependable Control of Discrete Systems (DCDS'11) June 15-17, 2011 Saarbrücken, Germany, pages 129–134.

52. A. Bobbio and R. Terruggia. Reliability and quality of service in weighted probabilistic networks using algebraic decision diagrams. In Proceedings IEEE Annual Reliability and Maintainability Symposium, pages 19–24, Fort Worth, TX, 2009.

53. J. Riordan. Introduction to combinatorial analysis. Dover Publications Inc.,2002.

54. A. Bobbio, R. Terruggia, E. Ciancamerla, and M. Minichino. Reliability analysis of multi-source multi-sink critical interacting systems. In 3rd IFAC Workshop on Dependable Control of Discrete Systems (DCDS'11) June 15-17, 2011 Saarbrücken, Germany, pages 129–134, 2011.

55. K.S. Brace, R.L. Rudell, and R.E. Bryant. Efficient implementation of a BDD package. In Proceedings 27-th ACM/IEEE Design Automation Conference, pages 40–45, 1990.

56. R.E. Bryant. Graph-based algorithms for Boolean function manipulation. IEEE Transactions on Computers, C-35:677–691, 1986.

57. Gary Hardy, Corinne Lucet, and Nikolaos Limnios. K-terminal network reliability measures with binary decision diagrams. IEEE Transactions on Reliability, 56(3):506–515, 2007.

58. A. Rauzy. New algorithms for fault trees analysis. Reliability Engineering & System Safety, 40(3):203 – 211, 1993.

59. R.E. Bryant. Graph-based algorithms for Boolean function manipulation. IEEE Transactions on Computers, C-35:677–691, 1986.

60. IEC 60870-5-101 Telecontrol equipment and systems - Part 5-101: Transmission protocols - Companion standard for basic telecontrol tasks

61. E. Ciancamerla, S. Di Blasi, C. Foglietta, D. Lefevre, M. Minichino, L. Lev, and Y. Shneck. Qos of a scada system versus qos of a power distribution grid. In Proceedings 10th International Probabilistic Safety Assessment & Management (PSAM) Conference PSAM 10, Seattle, WA, 2010.

62. E. Ciancamerla, M. Minichino, S. Palmieri. On prediction of QoS of SCADA accounting cyber attacks. Probabilistic Safety Assessment and Management Conference (PSAM11) and Annual European Safety and Reliability Conference (ESREL 2012)

63. ADversary VIew Security Evaluation (ADVISE). PERFORM Performability Engineering Research Group at the University of Illinois at Urbana-Champaign. https://www.mobius.illinois.edu/advise-alpha/index.php/Main_Page.

64. E. Ciancamerla, M. Minichino, S. Palmieri - On prediction of QoS of SCADA accounting cyber attacks - Probabilistic Safety Assessment and Management

Conference (PSAM11) and the Annual European Safety and Reliability Conference (ESREL 2012) - Helsinki, Finland - 25-29 June 2012

65. http://www.micie.eu

66. Ns2 - The Network Simulator. http://www.isi.edu/nsnam/ns/.

67. CockpitCI project, Deliverable D2.2- Reference Scenario – SCADA system of Power grid and corporate network under cyber attacks

68. W. Kroeger. Critical infrastructures at risk: a need for a new conceptual approach and extended analytical tools. Reliability Engineering and System Safety, 93:1781-1787, 2008.

69. W. Shaw. SCADA system vulnerabilities to cyber attack. http://www.electricenergyonline.com/?page=show_article&mag=23&article=181, 2012.

70. P. Chee-Wooi Ten, M. Govindarasu, and Chen-Ching Liu. Cybersecurity for electric power control and automation systems. In IEEE International Conference on Systems, Man and Cybernetics, 2007. ISIC., pages 29_34, 2007.

71. IEC-10125. Fault Tree Analysis. IEC-Standard-No. 10125, 1990.

72. J. Byres, J. Carter, and A. Elramly and D. Ho_man. Worlds in collision-ethernet and the factory floor. In ISA 2002 Emerging Technologies Conference, Instrumentation, Systems and Automation Society, Chicago, 2003.

73. P. Chee-Wooi Ten, Chen-Ching Liu, and M. Govindarasu. Vulnerability assessment of cybersecurity for scada systems using attack trees. In Proceedings IEEE Power Engineering Society General Meeting, pages 1_8, 2007.

74. B. Kordy and M. Pouly and P. Schweitzer. Computational aspects of attack & defense trees. In Security and Intelligent Information Systems, volume 7053 of Lecture Note, in Computer Science, pages 103_116. Springer Berlin Heidelberg, 2012.

75. S. Bistarelli, M. Dall'Aglio, and P. Peretti. Strategic games on defense trees. In Theo Dimitrakos, Fabio Martinelli, Peter Y.A. Ryan, and Steve Schneider, editors, Formal Aspects in Security and Trust, volume 4691 of Lecture Notes in Computer Science, pages 1-15. Springer Berlin Heidelberg, 2007.

76. R. O'Harrow. Search engine exposes industrial-sized dangers. http://www.theage.com.au/digital-life/consumer-security/search-engine-exposesindus trialsized- dangers-20120604-1zrnw.html, 2012.

77. R. Fricks and K. Trivedi. Importance analysis with markov chains. In Proceedings IEEE Annual Reliability and Maintainability Symposium, pages 89_95, 2003.

78. C.W. Ten, G. Manimaran, C.C. Liu (2010): Cybersecurity for Critical Infrastructures: Attack and Defense Modeling. IEEE Transactions on Systems, Man, and Cybernetics, Part A 40(4): 853-865 (2010)

79. H. Truong, R. Samborski, T. Fahringer, (2006) “Towards a Framework for Monitoring and Analyzing QoS Metrics of Grid Services,” in 2nd IEEE International Conference on e-Science and Grid Computing(e-Science 2006), Decemeber 2006.

80. Noel, S., Jajodia, S., Wang, L. and Singhal, A. (2010) "Measuring Security Risk of Networks Using Attack Graphs," International Journal of Next-Generation Computing, Vol. 1, No. 1, July 2010.
81. B. Sabata, S. Chatterjee, M. Davis, J. J. Sydir, and T. F. Lawrence, “Taxonomy of qos specifications,” in WORDS ’97: Proceedings of the 3rd Workshop on Object-Oriented Real-Time Dependable Systems - (WORDS ’97), page 100.
82. Rogers R., Carey M., Criscuolo P. and Petruzzi M. (2008) Nessus network auditing 2nd edition, Syngress Publishing, Inc. Elsevier, Burlington.
83. NIST (2012) ‘NVD: Common Vulnerability Scoring System (CVSS)’ [Online] Available at NIST, http://nvd.nist.gov/cvss.cfm?version=2 (Accessed: 2 July 2012)
84. Ouedraogo M., Khadraoui D., Mouratidis H. and Dubois E., “Appraisal and reporting of security assurance at operational systems level,”Journal of Systems and Software 85(1), January 2012.
85. Ouedraogo M., Mouratidis H., Khadraoui D. And Dubois E. (2009) Security Assurance metrics and Aggregation Techniques for IT systems, In Proceedings of ICIMP 2009, P.98-102, IEEE Computer Society.
86. MITRE (2012): Common Vulnerabilities and Exposures. Available at: http://cve.mitre.org/ [Accessed 15 December 2012]
87. Shaw T. William (2013) Scada systems vulnerabilities to cyber attack, available at: http://www.electricenergyonline.com/?page=show_article&article=181
88. Z.W. Birnbaum. On the importance of different components in a multicomponent systems. In Ed. P.R. Krishnaiah, editor, Multivariate Analysis -II, pages 581-592, New York, 1969, Academic Press.
89. A. Roy, Dong Seong Kim, and S. Trivedi. Act : Towards unifying the constructs of attack and defense trees. Security and Communication Networks, 3:1-15, 2011.
90. S.A. Zonouz, H. Khurana, W.H. Sanders, and T.M. Yardley. RRE: A game-theoretic intrusion response and recovery engine. In Dependable Systems Networks, 2009. DSN '09. IEEE/IFIP International Conference on, pages 439-448, 29 July 2009.
91. National Cyber Security Division of the US Department of Homeland Security (DHS). Cyber security evaluation tool. http://www.uscert.gov/control_systems/satool.html
92. http://www.isograph.com
93. Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", IETF RFC 4301. December 2005.
94. SSL VPN
95. Hamzeh, K., et al., “Point-to-Point Tunneling Protocol (PPTP)”, IETF RFC 2637, 1999.
96. OpenVPN, available at: http://openvpn.net
97. Aboba, B., and Dixon, W., "IPsec-Network Address Translation (NAT) Compatibility Requirements", IETF RFC 3715, March 2004.
98. Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, E. Lear, “Address Allocation for Private Internet's”, IETF RFC1918 February 1996.

99. E.Ciancamerla, B.Fresilli, M.Minichino, T.Patriarca, S. Iassinovski. ''An electrical grid and its SCADA system under cyber attack'', International Carnahan Conference on Security Technology (ICCST),13-17 October 2014, Rome.

100. www.snort.org

101. securityonion.net

102. Chris Simmons, Charles Ellis, Sajjan Shiva, Dipankar Dasgupta, Qishi Wu, "*AVOIDIT: A Cyber Attack Taxonomy*", Department of Computer Science, University of Memphis, Memphis, TN, USA, August 2009.

12 Appendix 1 - Cyber attack taxonomy

Objective

To provide an input on how we can model cyber-attacks according to a defined taxonomy. This input could also be a basis Modelling Tool based on epidemic spreading (SIR simulation).

12.1 Taxonomy description

A number of different taxonomies have been established during the last decade (Lough, Howard, Hansman, AVOIDIT) to formally describe a cyber attack. The following taxonomy is based on AVOIDIT (Attack Vector, Operational Impact, Defense, Information Impact, and Target) [102] and has been completed by adding supplementary categories in order to be able to precisely define cyber attacks targeting both SCADA and IT networks. The proposed taxonomy is also based on a paper focusing on the classification of cyber attacks on SCADA systems. Figure 64 provides an overview of the taxonomy.

The taxonomy classifies cyber attacks from 5 points of view: **"*Attack Vector*", "*Operational Impact*", "*Defence*", "*Informational Impact*" and "*Attack Target*"**. As mentioned by the authors of the AVOIDIT [102] taxonomy, the logical requirements of the classification are the following:

1. ***Mutually exclusive****: each attack can only be classified into one category, which prevents overlapping.* Even if a complex attack has to be classified in more than one type of cyber attack according to the level of the exploit or depth of the attack.

ID	Parent	Name	Attack Vector	Operational Impact	Defence	Informational Impact	Target
001	-	Industrial spying 1	Social Engineering	User Compromise	Awareness	Disclosure	User
002	001	Industrial spying 2 …	Social Engineering	Misuse of resources	Awareness	Disclosure	Local

2. ***Comprehensible***: Clear and concise information; able to be understood by experts as well as those who are less familiar.
3. ***Complete/Exhaustive***: available categories are exhaustive within each classification, it is assumed to be complete.
4. ***Unambiguous***: involves clearly defined classes, with no doubt of which class the attack belongs to.
5. ***Repeatable***: the classification of attack should be repeatable.
6. ***Terms well defined***: categories should be well defined, and those terms should consist of established terminology that is compliant within the security community.
7. ***Useful***: the ability to be used to gain insight into a particular field of study, particularly by those having great interest within the field of study.

The AVOIDIT taxonomy paper, mentioned above, gives a precise description of the chosen categories which can be summarised and completed as follows:

Attack Vector: An attack vector is defined as a **path** by which an attacker can gain access to a host. The majority of attacks can be described according to the following attack vector:

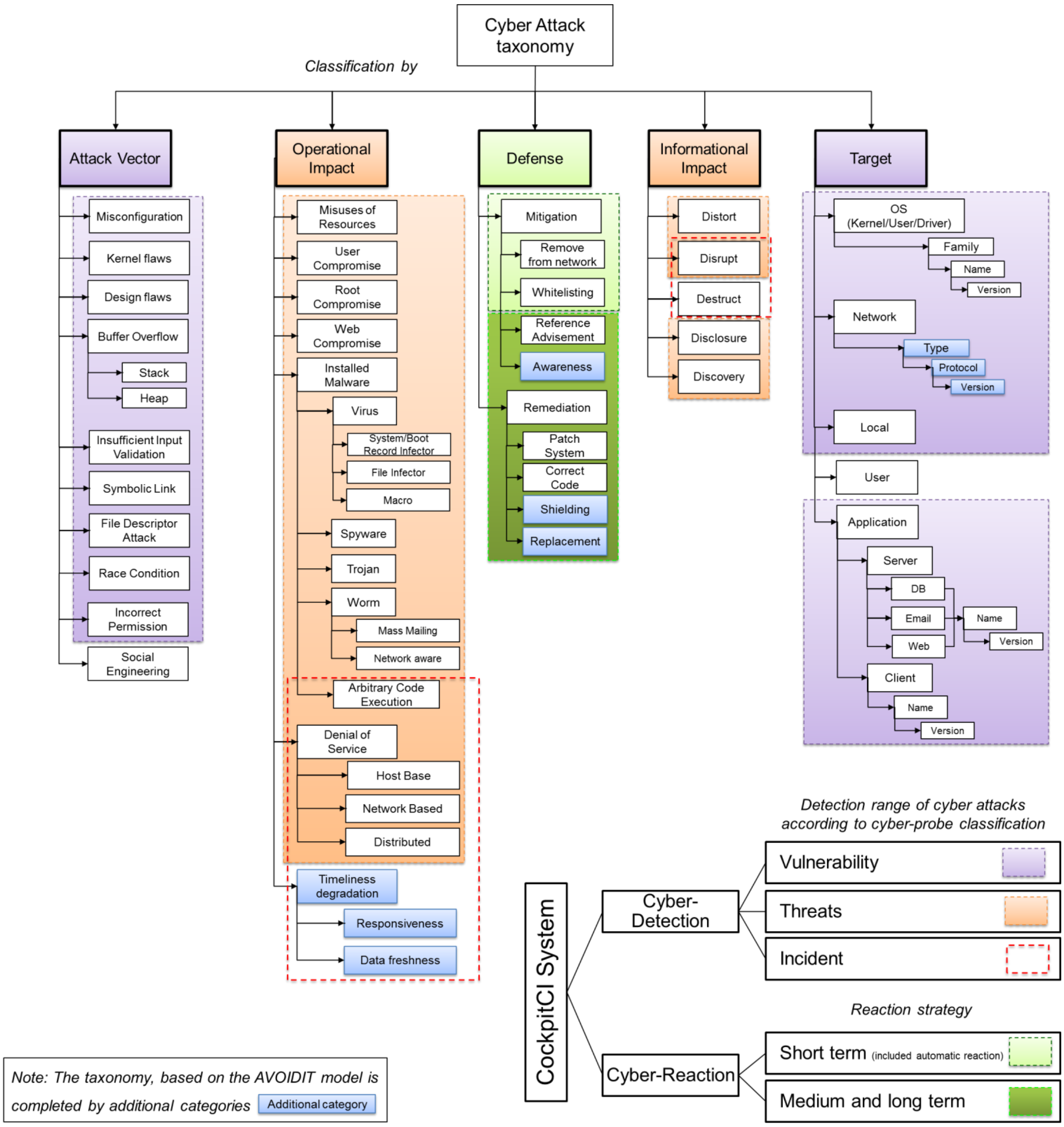


Figure 64: Cyber-attacks taxonomy for CockpitCI system

1. ***Misconfiguration***: use of a configuration flaw within a particular application to gain access to a network or personal computer.
2. ***Kernel Flaws***: use of a kernel flaw within an operating system to gain certain privileges to exploit vulnerabilities within the operating system. (e.g. Vulnerability on kernel of Wind River System VxWorks which is used in hundreds of devices: http://www.kb.cert.org/vuls/id/362332)
3. ***Design Flaws***: use of a design flaw within a system or device to retrieve sensitive information (e.g. password theft using the firewire or thunderbolt connectivity flaw even if the computer is locked and encrypted: http://erratasec.blogspot.com/2011/02/thunderbolt-introducing-new-way-to-hack.html#.UZ-iOypXvJE)
4. ***Buffer Overflow:*** Buffer overflow is caused when a piece of code does not adequately check for appropriate input length and the input value is not the size the program expects. An attack can exploit a buffer overflow vulnerability leading to a possible exploitation of arbitrary code execution.
5. ***Insufficient Input Validation:*** A program fails to validate the input sent from a user. An attacker can exploit the insufficient input validation vulnerability and inject arbitrary code (e.g. SQL injection).
6. ***Symbolic Links:*** A file that points to another file. An attacker can exploit a symbolic link vulnerability to point to a target file for which an operating system process has write permissions.
7. ***File Descriptor:*** A file that uses numbers from a system to keep track of files, as opposed to file names. Exploitation of the file descriptor vulnerability allows an attacker the possibility of gaining elevated privileges to program related files.
8. ***Race Condition***: Occurs when a program attempts to run a process and the object changes concurrently between repeated references allowing an attacker to gain elevated privileges while a program or process is in privilege mode.
9. ***Incorrect File/Directory Permission:*** An incorrect permission associated to a file or directory consists of not assigning users and processes appropriately.
10. ***Social Engineering:*** The process of using social interactions to acquire information about a victim or computer system, which, in normal circumstances, is not available.(e.g. phishing is a social engineering method to penetrate systems, even those protected by technical systems like IDS: http://www.social-engineer.org/framework/Real_World_Social_Engineering_Examples:_Phishing)

Operational Impact: An operational impact is defined here as an evaluated consequence of an attack at operational level (IT and SCADA level). Classification by Operational Impact involves the ability for an attack to culminate and provide high level information known by security experts, as well those less familiar with cyber attacks.

1. ***Misuse of Resources:*** An unauthorised use of IT/SCADA resources or IT/SCADA functions (usable with specific privileges).
2. ***User Compromise***: Gaining unauthorised use of user privileges on a host.
3. ***Root Compromise***: Gaining or elevating privileges to unauthorised privileges of an administrator on a particular host/ system.

4. ***Web Compromise***: A website or web application using vulnerabilities to further an attack (cross site scripting or SQL injection).
5. ***Installed Malware***: An attack can be launched via user installed malware, whether by intentional installation or drive by installation. Installed malware can allow an adversary to gain full control of the compromised system leading to the exposure of sensitive information or remote control of the host.
    a. ***Virus:*** A piece of code that will attach itself through some form of infected files, which will self-replicate upon execution of a program. (boot record infectors, file infectors, and macros).
    b. ***Spyware:*** collecting information from a computing system without the owner's consent.
    c. ***Trojan***: A benign program that allows unauthorised backdoor access to a compromised system.
    d. ***Worms***: A self-replicating computer program that spreads throughout a network. Worms include mass mailing and network aware worms.
    e. **Arbitrary Code Execution**: Involves a malicious entity that gains control through injecting its own code in order to perform any operation on the targeted application.
6. ***Denial of Service***: Denial of Service (DoS) is an attack which denies a victim access to a particular resource or service i.e.:
    a. ***Host Based:*** A Host based DoS aims at attacking a specific computer target within the configuration, operating system, or software of a host. These types of attacks usually involve resource hogs, aimed at consuming up all resources on a computer; or crashers, which attempt to crash the host system.
    b. ***Network Based:*** A Network based DoS targets a complete network of computers to prevent the network from providing normal service. Network based DoS usually occurs in the form of flooding with packets, where the network's connectivity and bandwidth are the target.
    c. ***Distributed:*** A Distributed Denial of Service (DDoS) is becoming increasingly more popular as an attacker's choice of DoS. A distributed denial of service uses multiple attack vectors to obtain its goal.
7. ***Timeliness degradation***: This attack aims to stop a system responding on time to commands. This type of attack will degrade the QoS provided by a system by targeting either the entire system, specific system functionality or system resources and can seriously impact the QoS of a Critical Infrastructure if it targets industrial components such as a PLC controller. The timeliness aspect includes both the responsiveness of a system (real-time response) and the freshness of data (for an industrial system, the data is only valid in a designed time period).

Defence: Classification by defence highlights several strategies a defender can employ to remain vigilant in defending against pre and post attacks.

1. ***Mitigation***: A form of defence used prior to vulnerability exploitation or during an attack, to mitigate damage an attack has caused, or has the potential to cause. Mitigation involves reducing the severity of the attack.

   a. ***Remove from Network***: The ability of an administrator to remove infected hosts, thus preventing further damage.
   b. ***Whitelisting***: A list of permissible connections that are known to the defender.
   c. ***Reference Advisement***: Notes provided by the defender to mitigate an attack, or a vulnerability/vendor database reference number used to alleviate a vulnerability or attack.
   d. Awareness

2. ***Remediation***: Defence used, in the presence or prior to vulnerability exploitation, to prevent an attack.
   a. ***Patch System***: Applying patches which have been released due to software vulnerabilities.
   b. ***Correct Code***: Steps within an organisation to release a code patch to a specific application that will eliminate the potential for an attacker to exploit.
   c. ***Shielding***: Steps within an organisation to avoid unnecessary physical or logical access to system resources.
   d. ***Replacement***: Steps within an organisation to remove the out-dated system or breakdown system and to replace it with a more secure system.

***Informational Impact*:** An informational impact is defined here as an evaluated consequence of an attack on the reliability of information used (confidentiality, integrity and availability of information) at operational level (IT and SCADA level). An attack on a targeted system has the potential to impact sensitive information in various ways. A committed resource must be able defend information warfare strategies in an effort to protect themselves against theft, disruption, distortion, denial of service, or destruction of sensitive information assets.

1. ***Distort***: A distortion of information, usually when an attack has caused the modification of a file.
2. ***Disrupt***: A disruption to services, usually from a Denial of Service attack, involving unavailability of information access.
3. ***Destruct***: A destruction of information, usually when an attack has caused a deletion of files or a removal of access.
4. ***Disclosure***: A disclosure of information, usually providing an attacker with access to information that they would not normally have access to.
5. ***Discovery***: To discover previously unknown information. For example, when a scanning tool probes for information, the information discovered can be used to launch an attack on a particular target.

***Target*:** Generally an attack targets a specific type of host. The classification assigned to the target is able to improve the defence of a whole set of systems by adapting mitigation and remediation actions to a specific range of systems.

1. ***Operating System (Kernel/ User/ Driver):*** Responsible for the coordination of activities and distributing the resources of a computer. An attack can be designed to target vulnerabilities within a particular operating system which can be defined by its

family (Microsoft Windows), its name (e.g. MSWIN7), and its version (MS Windows 7 64-bit SP1).

2. ***Network:*** To target a particular network or gain access through vulnerability within a network or one of the network protocols. The network target can be specified by its Area (Corporate network, IT operational network, SCADA network etc…) its Type (wired, wireless, radio waves etc…), its Protocol (ModBus, Ethernet [802.3], internet [IPV4], SONET [OC1] etc…) and its Version.
3. ***Local:*** An attack targeting a user's local computer.
4. ***User:*** An attack against a user is an attack to retrieve a user's personal information.
5. ***Application***: An attack towards specific software. An application can be either client or server. A client application is software that helps a user perform common tasks. A server application is software designed to serve as a host to multiple concurrent users.

13 Glossary

The purpose of the glossary is to identify and define an unambiguous vocabulary of terms which will be used to formulate the requirements.

access

ability and means to communicate with or otherwise interact with a system in order to use system resources. Access may involve physical access (authorization to be allowed physically in an area, possession of a physical key lock, PIN code, or access card or biometric attributes that allow access) or logical access (authorization to log in to a system and application, through a combination of logical and physical means)

access control

protection of system resources against unauthorized access; a process by which use of system resources is regulated according to a security policy and is permitted by only authorized entities (users, programs, processes, or other systems) according to that policy .

access control list

a list of permissions attached to an object. An access control list (ACL) specifies which users or system processes are granted access to objects, as well as what operations are allowed on given objects. Each entry in a typical ACL specifies a subject and an operation. For instance, if a file has an ACL that contains (Alice, delete), this would give Alice permission to delete the file.

asset

physical or logical object owned by or under the custodial duties of an organization, having either a perceived or actual value to the organization. In the case of industrial automation and control systems the physical assets that have the largest directly measurable value may be the equipment under control.

There are many types of assets, including: (a) information; (b) software, such as a computer program; (c) physical, such as computer; (d) services; (e) people, and their qualifications, skills, and experience; and (f) intangibles, such as reputation and image.

attack

assault on a system that derives from an intelligent threat, i.e., an intelligent act that is a deliberate attempt (especially in the sense of a method or technique) to evade security services and violate the security policy of a system. There are different commonly recognized classes of attack:

- An "active attack" attempts to alter system resources or affect their operation.
- A "passive attack" attempts to learn or make use of information from the system but does not affect system resources.

- An "inside attack" is an attack initiated by an entity inside the security perimeter (an "insider"), i.e., an entity that is authorized to access system resources but uses them in a way not approved by those who granted the authorization.

- An "outside attack" is initiated from outside the perimeter, by an unauthorized or illegitimate user of the system (including an insider attacking from outside the security perimeter). Potential outside attackers range from amateur pranksters to organized criminals, international terrorists, and hostile governments.

attack potential

perceived potential for success of an attack, should an attack be launched, expressed in terms of an attacker's expertise, resources and motivation

attack vector

path or means by which an attacker can gain access to a computer or network server in order to deliver a malicious outcome

authenticate

verify the identity of a user, user device, or other entity, or the integrity of data stored, transmitted, or otherwise exposed to unauthorized modification in an information system, or to establish the validity of a transmission.

authentication

security measure designed to establish the validity of a transmission, message, or originator, or a means of verifying an individual's authorization to receive specific categories of information.

authentication factor:

Piece of information and/or process used to authenticate or verify the identity of an entity. Authentication factors are divided into four categories: 1) something an entity has (e.g., device signature, passport, hardware device containing a credential, private key); 2) something an entity knows (e.g., password, PIN); 3) something an entity is (e.g., biometric characteristic); or 4) something an entity typically does (e.g., behavior pattern).

authorization

right or a permission that is granted to a system entity to access a system resource. Authorization uses identity attribute, for the subject, and control attributes, for the resource, to decide on a permission. Typically, authorization decisions are based on a policy. An authorization result may be intended for immediate use in accessing a resource, or it may be encoded as the value of an identity attribute and registered with the dentist of the subject for future use.

availability

probability that an asset, under the combined influence of its reliability, maintainability, and security, will be able to fulfill its required function over a stated period of time, or at a given point in time.

border

edge or boundary of a physical or logical security zone.

boundary

software, hardware, or other physical barrier that limits access to a system or part of a system.

communication path

logical connection between a source and one or more destinations, which could be devices, physical processes, data items, commands, or programmatic interfaces. The communication path is not limited to wired or wireless networks, but includes other means of communication such as memory, procedure calls, state of physical plant, portable media, and human interactions.

communication security

(1) measures that implement and assure security services in a communication system, particularly those that provide data confidentiality and data integrity and that authenticate communicating entities.

(2) state that is reached by applying security services, in particular, state of data confidentiality, integrity, and successfully authenticated communications entities

compromise

unauthorized disclosure, modification, substitution, or use of information (including plaintext cryptographic keys and other critical security parameters).

confidentiality

assurance that information is not disclosed to unauthorized individuals, processes, or devices

control centre

central location used to operate a set of assets. Infrastructure industries typically use one or more control centers to supervise or coordinate their operations. If there are multiple control centers (for example, a backup center at a separate site), they are typically connected together via a wide area network. The control center contains the SCADA host computers and associated operator display devices plus ancillary information systems such as a historian.

control equipment

class that includes distributed control systems, programmable logic controllers, SCADA systems, associated operator interface consoles, and field sensing and control devices used to manage and control the process. The term also includes field bus networks where control logic and algorithms are executed on intelligent electronic devices that coordinate actions with each other, as well as systems used to monitor the process and the systems used to maintain the process.

control network

time-critical network that is typically connected to equipment that controls physical processes. The control network can be subdivided into zones, and there can be multiple separate control networks within one company or site.

countermeasure

action, device, procedure, or technique that reduces a threat, a vulnerability, or an attack by eliminating or preventing it, by minimizing the harm it can cause, or by discovering and reporting it so that corrective action can be taken.

cryptographic algorithm

algorithm based upon the science of cryptography, including encryption algorithms, cryptographic hash algorithms, digital signature algorithms, and key agreement algorithms.

cryptographic key

input parameter that varies the transformation performed by a cryptographic algorithm . Usually shortened to just "key."

cyber attack(s)

Type of attacks where services or applications in the Cyberspace are used or are the target of attack, or where Cyberspace is the source, tool, target, or place of an attack.

data confidentiality

property that information is not made available or disclosed to any unauthorized system entity, including unauthorized individuals, entities, or processes-

data integrity

property that data has not been changed, destroyed, or lost in an unauthorized or accidental manner. This term deals with constancy of and confidence in data values, not with the information that the values represent or the trustworthiness of the source of the values.

decryption

process of changing cipher text into plaintext using a cryptographic algorithm and key .

defense in depth

provision of multiple security protections, especially in layers, with the intent to delay if not prevent an attack. Defense in depth implies layers of security and detection, even on single systems, and provides the following features: a) attackers are faced with breaking through or bypassing each layer without being detected; b) flaw in one layer can be mitigated by capabilities in other layers; c) system security becomes a set of layers within the overall network security.

demilitarized zone

perimeter network segment that is logically between internal and external networks. The purpose of a demilitarized zone is to enforce the internal network's policy for external information exchange and to provide external, untrusted sources with restricted access to releasable information while shielding the internal network from outside attacks.

denial of service

prevention or interruption of authorized access to a system resource or the delaying of system operations and functions. In the context of industrial control systems, denial of service can refer to loss of process function, not just loss of data communications.

digital signature

result of a cryptographic transformation of data which, when properly implemented, provides the services of origin authentication, data integrity, and signer non-repudiation.

distributed control system

type of control system in which the system elements are dispersed but operated in a coupled manner. Distributed control systems may have shorter coupling time constants than those typically found in SCADA systems.

domain

environment or context that is defined by a security policy, security model, or security architecture to include a set of system resources and the set of system entities that have the right to access the resources.

electronic security

actions required to preclude unauthorized use of, denial of service to, modifications to, disclosure of, loss of revenue from, or destruction of critical systems or informational assets. Electronic security includes the concepts of identification, authentication, accountability, authorization, availability, and privacy.

encryption

cryptographic transformation of plaintext into cipher text that conceals the data's original meaning to prevent it from being known or used. If the transformation is reversible, the corresponding reversal process is called "decryption," which is a transformation that restores encrypted data to its original state.

corporate

business entity that produces or transports products or operates and maintains infrastructure services.

corporate system

collection of information technology elements (i.e., hardware, software and services) installed with the intent to facilitate an organization's business process or processes (administrative or project).

equipment under control

equipment, machinery, apparatus or plant used for manufacturing, process, transportation, medical or other activities.

field I/O network

communications link (wired or wireless) that connects sensors and actuators to the control equipment.

firewall

inter network connection device that restricts data communication traffic between two connected networks. A firewall may be either an application installed on a general purpose computer or a dedicated platform (appliance) that forwards or rejects/drops packets on a network. Typically firewalls are used to define zone borders. Firewalls generally have rules restricting which ports are open.

gateway

relay mechanism that attaches to two (or more) computer networks that have similar functions but dissimilar implementations and that enables host computers on one network to communicate with hosts on the other. Also described as an intermediate system that is the translation interface between two computer networks.

geographic site

subset of an corporate’s physical, geographic, or logical group of assets. A geographic site may contain areas, manufacturing lines, process cells, process units, control centers, and vehicles and may be connected to other sites by a wide area network.

guard

gateway that is interposed between two networks (or computers or other information systems) operating at different security levels (one network is usually more secure than the other) and is trusted to mediate all information transfers between the two networks, either to ensure that no sensitive information from the more secure network is disclosed to the less secure network, or to protect the integrity of data on the more secure network .

host

computer that is attached to a communication sub network or inter network and can use services provided by the network to exchange data with other attached systems.

industrial control systems

humans, hardware and software that can affect or influence the safe, secure, and reliable operation of an industrial process. These systems include, but are not limited to: a) industrial control systems, including distributed control systems (DCSs), programmable logic controllers (PLCs), remote terminal units (RTUs), intelligent electronic devices, supervisory control and data acquisition (SCADA), networked electronic sensing and control, and monitoring and diagnostic systems. (In this context, process control systems include basic process control system and safety instrumented system (SIS) functions, whether they are physically separate or integrated.) b) associated information systems such as advanced or multivariable control, online optimizers, dedicated equipment monitors, graphical interfaces, process historians, manufacturing execution systems, and plant information management systems. c) associated internal, human, network, or machine interfaces used to provide control, safety, and manufacturing operations functionality to continuous, batch, discrete, and other processes.

initial risk

risk before controls or countermeasures have been applied.

Insider

trusted person, employee, contractor, or supplier who has information that is not generally known to the public.

integrity

quality of a system reflecting the logical correctness and reliability of the operating system, the logical completeness of the hardware and software implementing the protection mechanisms, and the consistency of the data structures and occurrence of the stored data.

interception

capture and disclosure of message contents or use of traffic analysis to compromise the confidentiality of a communication system based on message destination or origin, frequency or length of transmission, and other communication attributes.

interface

logical entry or exit point that provides access to the module for logical information flows.

intrusion

unauthorized act of compromising a system.

intrusion detection

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.

IP address

address of a computer or device that is assigned for identification and communication using the Internet Protocol and other protocols.

key management

process of handling and controlling cryptographic keys and related material (such as initialization values) during their life cycle in a cryptographic system, including ordering, generating, distributing, storing, loading, escrowing, archiving, auditing, and destroying the keys and related material.

local area network

communications network designed to connect computers and other intelligent devices in a limited geographic area (typically less than 10 kilometers).

malicious code

programs or code written for the purpose of gathering information about systems or users, destroying system data, providing a foothold for further intrusion into a system, falsifying system data and reports, or providing time-consuming irritation to system operations and maintenance personnel. Malicious code attacks can take the form of viruses, worms, Trojan Horses, or other automated exploits. Malicious code is also often referred to as malware.

non repudiation

security service that provides protection against false denial of involvement in a communication.

OPC

set of specifications for the exchange of information in a process control environment. The abbreviation OPC originally came from OLE for Process Control, where OLE was short for Object Linking and Embedding.

outsider

person or group not trusted with inside access, who may or may not be known to the targeted organization.

penetration

successful unauthorized access to a protected system resource.

privilege

authorization or set of authorizations to perform specific functions, especially in the context of a computer operating system. Examples of functions that are controlled through the use of privilege include acknowledging alarms, changing set points, modifying control algorithms.

process

series of operations performed in the making, treatment or transportation of a product or material.

protocol

set of rules (i.e., formats and procedures) to implement and control some type of association (e.g., communication) between systems.

reference model

structure that allows the modules and interfaces of a system to be described in a consistent manner.

reliability

ability of a system to perform a required function under stated conditions for a specified period of time.

remote access

use of systems that are inside the perimeter of the security zone being addressed from a different geographical location with the same rights as when physically present at the location.

remote client

asset outside the control network that is temporarily or permanently connected to a host inside the control network via a communication link in order to directly or indirectly access parts of the control equipment on the control network.

repudiation

denial by one of the entities involved in a communication of having participated in all or part of the communication.

residual risk

the remaining risk after the security controls or countermeasures have been applied.

risk

expectation of loss expressed as the probability that a particular threat will exploit a particular vulnerability with a particular consequence.

risk assessment

process that systematically identifies potential vulnerabilities to valuable system resources and threats to those resources, quantifies loss exposures and consequences based on probability of occurrence, and (optionally) recommends how to allocate resources to countermeasures to minimize total exposure. Types of resources include physical, logical and human. Risk assessments are often combined with vulnerability assessments to identify vulnerabilities and quantify the associated risk. They are carried out initially and periodically to reflect changes in the organization's risk tolerance, vulnerabilities, procedures, personnel and technological changes.

risk management

process of identifying and applying countermeasures commensurate with the value of the assets protected based on a risk assessment .

risk mitigation controls

combination of countermeasures and business continuity plans.

role-based access control

form of identity-based access control where the system entities that are identified and controlled are functional positions in an organization or process.

router

gateway between two networks at OSI layer 3 and that relays and directs data packets through that inter-network. The most common form of router passes Internet Protocol (IP) packets.

safety

freedom from unacceptable risk .

safety network

network that connects safety-instrumented systems for the communication of safety related information.

secret

condition of information being protected from being known by any system entities except those intended to know it .

security

- measures taken to protect a system.
- condition of a system that results from the establishment and maintenance of measures to protect the system.
- condition of system resources being free from unauthorized access and from unauthorized or accidental change, destruction, or loss.
- capability of a computer-based system to provide adequate confidence that unauthorized persons and systems can neither modify the software and its data nor gain access to the system functions, and yet to ensure that this is not denied to authorized persons and systems.
- prevention of illegal or unwanted penetration of or interference with the proper and intended operation of an industrial automation and control system.

security architecture

plan and set of principles that describe the security services that a system is required to provide to meet the needs of its users, the system elements required to implement the services, and the performance levels required in the elements to deal with the threat environment. Security architecture would be an architecture to protect the control network from intentional or unintentional security events.

security audit

independent review and examination of a system's records and activities to determine the adequacy of system controls, ensure compliance with established security policy and procedures, detect breaches in security services, and recommend any changes that are indicated for countermeasures.

security components

assets such as firewalls, authentication modules, or encryption software used to improve the security performance of an industrial automation and control system.

security event

occurrence in a system that is relevant to the security of the system.

security function

function of a zone or conduit to prevent unauthorized electronic intervention that can impact or influence the normal functioning of devices and systems within the zone or conduit.

security incident

adverse event in a system or network or the threat of the occurrence of such an event.

security intrusion

security event, or a combination of multiple security events, that constitutes a security incident in which an intruder gains, or attempts to gain, access to a system (or system resource) without having authorization to do so.

security level

level corresponding to the required effectiveness of countermeasures and inherent security properties of devices and systems for a zone or conduit based on assessment of risk for the zone or conduit.

security objective

aspect of security which to achieve is the purpose and objective of using certain mitigation measures, such as confidentiality, integrity, availability, user authenticity, access authorization, accountability.

security perimeter

boundary (logical or physical) of the domain in which a security policy or security architecture applies, i.e., the boundary of the space in which security services protect system resources.

security performance

program’s compliance, completeness of measures to provide specific threat protection, post compromise analysis, review of changing business requirements, new threat and vulnerability information, and periodic audit of control systems to ensure security measures remain effective and appropriate. Tests, audits, tools, measures, or other methods are required to evaluate security practice performance.

security policy

set of rules that specify or regulate how a system or organization provides security services to protect its assets.

security procedures

definitions of exactly how practices are implemented and executed. Security procedures are implemented through personnel training and actions using currently available and installed technology.

security program

a combination of all aspects of managing security, ranging from the definition and communication of policies through implementation of best industry practices and ongoing operation and auditing.

security services

mechanisms used to provide confidentiality, data integrity, authentication, or no repudiation of information.

security violation

act or event that disobeys or otherwise breaches security policy through an intrusion or the actions of a well-meaning insider.

security zone

grouping of logical or physical assets that share common security requirements. A zone has a clear border with other zones. The security policy of a zone is typically enforced by a combination of mechanisms both at the zone edge and within the zone. Zones can be hierarchical in the sense that they can be comprised of a collection of subzones.

server

device or application that provides information or services to client applications and devices.

supervisory control and data acquisition (SCADA) system

type of loosely coupled distributed monitoring and control system commonly associated with electric power transmission and distribution systems, oil and gas pipelines, and water and sewage systems.

system software

special software designed for a specific computer system or family of computer systems to facilitate the operation and maintenance of the computer system and associated programs and data.

threat

potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm.

threat action

assault on system security.

traffic analysis

inference of information from observable characteristics of data flow(s), even when the data are encrypted or otherwise not directly available, including the identities and locations of source(s) and destination(s) and the presence, amount, frequency, and duration of occurrence.

use case

technique for capturing potential functional requirements that employs the use of one or more scenarios that convey how the system should interact with the end user or another system to achieve a specific goal. Typically use cases treat the system as a black box, and the interactions with the system, including system responses, are as perceived from outside of the system. Use cases are popular because they simplify the description of requirements, and avoid the problem of making assumptions about how this functionality will be accomplished.

user

person, organization entity, or automated process that accesses a system, whether authorized to do so or not.

vulnerability

flaw or weakness in a system's design, implementation, or operation and management that could be exploited to violate the system's integrity or security policy.

wide area network

communications network designed to connect computers, networks and other devices over a large distance, such as across the country or world.

wiretapping

attack that intercepts and accesses data and other information contained in a flow in a communication system. Although the term originally referred to making a mechanical connection to an electrical conductor that links two nodes, it is now used to refer to reading information from any sort of medium used for a link or even directly from a node, such as a gateway or sub network switch. "Active wiretapping" attempts to alter the data or otherwise affect the flow; "passive wiretapping" only attempts to observe the flow and gain knowledge of information it contains.

14 Acronym and symbols

Acronym	Explanation
AC	Access Complexity
ADVISE	ADversary VIew Security Evaluation
AEG	Attack Execution Graph
AIOS	Attacker Intent, Objectives and Strategies
ARP	Address Resolution Protocol
AT	Attack Tree
BDD	Binary Decision Diagram
CAIDI	Customer Average Interruption Duration
CBR	Constant Bit Rate
CCI	Communication Critical Infrastructure
CERT	Computer Emergency Response Team
CI	Critical Infrastructure
CIA	Confidentiality, Integrity, and Availability
CISIA	Critical Infrastructure Simulation by Interdependent Agents
CPM	Cyber Propagation Module
CSET	Cyber Security Evaluation Tool
CVSS	Common Vulnerability Scoring System
DDoS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol
DMZ	Demilitarized Zone
DNRA	Delay Network Reliability Analyzer
DNS	Domain Name System
DoS	Denial of Service
ECI	Electrical Critical Infrastructure
FISR	Fault Isolation and System Restoration
FTA	Fault Tree Analysis
FTP	File Transfer Protocol
FW	Firewall
GSPN	Generalized Stochastic Petri Net
GUI	Graphical User Interface
HMI	Human Machine Interface
HTB	Hybrid Test Bed
HV	High Voltage

ICCP	Inter-Control Center Communications Protocol
ICMP	Internet Control Message Protocol
ICS	Industrial Controls
ICT	Information and Communication Technology
IDS	Intrusion Detection System
IEC	Israel Electric Corporation
IED	Intelligent Electronic Device
IMT	Incident Management Team
IP	Internet Protocol
IPSec	Internet Protocol Security
IRP	Integrated Risk Prediction
ISP	Internet Service Provider
IT	Information Technology
LAN	Local Area Network
LoC	Loss of Control
LoV	Loss of View
LV	Low Voltage
MDLC	Motorola Data Link Communication
MHR	Mixed Holistic Reductionist
MITM	Man In The Middle
MTBDD	Multi Terminal Binary Decision Diagram
MTU	Master Terminal Unit
MV	Medium Voltage
N.O.	Normally Open
NARUC	National Association of Regulatory Utility Commissioners
NCC	National Control Centre
NIDS	Network Intrusion Detection System
NIST	National Institute of Standards and Technology
NMS	Network Management System
NRA	Network Reliability Analyzer
NVD	National Vulnerability Database
OL	Operative Level
OPC	OLE for Process Control
OS	Operation System
OSI	Open Systems Interconnect
PERFORM	Performability Engineering Research Group

PLC	Programmable Logic Controller
PN	Petri Net
PoP	Point of Presence
QoS	Quality of Service
R	Requirements
RF	Radio Frequency
RTT	Round Trip Time
RTU	Remote Terminal Unit
SAIDI	System Average Interruption Duration
SAIFI	System Average Frequency Interruption
SCADA	Supervisory Control And Data Acquisition
SCC	SCADA Control Centre
SNMP	Simple Network Management Protocol
STM	Synchronous Transport Module
STP	Spanning Tree Protocol
TB	Test Bed
TCP	Transport Control Protocol
UDP	User Datagram Protocol
VPN	Virtual Private Network