From Detecting Cyber Attacks to Mitigating Risk Within a Hybrid Environment

Chiara Foglietta, *Member, IEEE,* Dario Masucci, Cosimo Palazzo, Riccardo Santini, *Student Member, IEEE,* Stefano Panzieri, Luis Rosa, Tiago Cruz, *Member, IEEE,* Leonid Lev *Senior Member, IEEE*

***Abstract*—Telecommunication networks based on commonplace technologies (such as Ethernet) often constitute a vulnerable attack vector against modern Critical Infrastructures (CIs), particularly for Supervisory Control and Data Acquisition (SCADA) systems, which rely on them for monitoring and controlling physical components. This article presents a unique platform that encompasses a range of capabilities, from cyber attack detection to mitigation strategies, through interdependency and risk evaluation. The platform is made of two main components: a cyber attack detection subsystem and a risk assessment framework. Both blocks are innovative from a research point of view and they have been developed and customized to fit the CIs' features, that are completely different from telecommunication networks. This platform has been tested on a hybrid environment testbed, made of virtual and real components, within the scope of the EU FP7 CockpitCI and EU H2020 ATENA projects. The case study corresponds to a medium voltage power grid controlled by a SCADA control center, where the platform has been validated with optimal results in terms of detection capabilities and time response.**

***Index Terms*—cyber attack detection risk assessment, decision support systems, cyber-physical systems, supervisory control and data acquisition (SCADA)**

I. Introduction

THE concept of Critical Infrastructure has been changing over the past years. This notion, which was mainly related to the public sector during the 1980s [1], was redefined as a matter of national security [2] during the 1990s, and particularly after 9/11. This comes as no surprise, as CIs are the key assets, systems or networks of our lives; their partial destruction would have a negative effect on security, economy and public health.

With time, the definition of CI has been extended to include other services. The concept of "lifeline system" was developed to evaluate large and geographically distributed networks, such as electric power, gas and liquid fuels, telecommunications, transportation, waste disposal and water supply. Thinking about CIs through the subset of lifelines helps clarify common features on essential support systems and provides insights into the challenges to improve the performance of large networks.

CIs are often interdependent, due to physical proximity and service interaction [3]. For instance, this is the case with old pipelines and cables lying in the city underground, nearby other important facilities such as electric transformers. Another example of infrastructure interdependency is related to telecommunication networks, which play a crucial role to support the management and operation of several modern CIs, as it is the case for SCADA systems. The first generations of SCADA systems relied on isolation (the air-gap principle), based on the use of proprietary and/or poorly documented technologies, to ensure security. However, as architectures evolved, these systems assimilated technologies from the Information and Communication Technologies (ICT) world, such as TCP/IP and Ethernet networking, encouraging the interconnection of Industrial Control Systems (ICS) with organizational ICT network infrastructures and even with the exterior (e.g., for remote maintenance) [4]. Once connected, SCADA ICS became increasingly vulnerable to a range of emerging threats, being targeted by a range of new actors as part of a cyber-warfare strategy. The well-publicized Stuxnet [5] worm is a prominent example of the latter case.

Nowadays, the number of known and unknown (zero-day) vulnerabilities are increasing, and therefore also the number of cyber attacks against lifeline systems [6]. Detecting such attacks is a basic step towards the proper management of physical infrastructures, but operators also need information about the effects of such incidents on physical systems. For the latter purpose, it is mandatory to create models that consider the existing interdependencies among several Critical Infrastructures, in order to assess the consequences of adverse events: this means that physical devices and delivered services must be considered together to produce a common result.

A. Contributions

This paper describes a unique framework for monitoring cyber attacks, evaluating trustworthiness of detected attacks, assessing the effects on the physical systems and suggesting possible countermeasures to operators for mitigating risks. The main contribution is the integration of heterogeneous capabilities in order to cope with cyber attacks and interdependencies, in a near real-time manner. Each capability (there is a layer for detection and another for risk assessment) has been carefully designed to work within and in connection with CIs.

Both the detection and the risk assessment layers are innovative from a research point of view and, in addition, they

C. Foglietta, D. Masucci, C. Palazzo, R. Santini and S. Panzieri are with the Department of Engineering, University of "Roma TRE", Rome 00146, Italy (email: chiara.foglietta@uniroma3.it, dario.masucci@uniroma3.it, cosimo.palazzo@uniroma3.it, riccardo.santini@uniroma3.it, stefano.panzieri@uniroma3.it)

L. Rosa and T. Cruz are with the Department of Informatics Engineering, University of Coimbra, Coimbra 3030-290, Portugal (email: lmrosa@dei.uc.pt, tjcruz@dei.uc.pt)

L. Lev is with Israel Electric Corporation, Haifa 31000, Israel (email: leonid.lev@iec.co.il)

Manuscript received June 07, 2017; This work was supported by the ATENA European H2020 project (H2020-DS-2015-1 Project 700581). (*Corresponding author: C. Foglietta*)

have been integrated in order to help operators in the decision making process of evaluating the consequences of cyber attacks on physical infrastructures. These innovations encompass several components, such as the Shadow Security Unit (SSU) and CISIApro. The former is a detection probe specifically developed to be deployed at the edge of the SCADA ICS cyber-physical domain, while the latter constitutes the main element of the risk assessment layer, being able to handle real-time data coming from heterogeneous sources and assess the consequences of faults and cyber-attack on interdependent CIs.

The validation of this process was realized within a hybrid environment, designated as the Hybrid Environment for Development and Validation (HEDVa), made of virtual equipment and real physical devices, with the purpose of testing its performance in a real environment without the possibility of generating real damage.

B. Organization

The paper is organized as follows: Section II reviews the literature on anomaly detection for cyber attacks in SCADA networks and on interdependency and risk evaluation; on Section III the cyber attack detection platform is described, with Section IV presenting the interdependency model with risk assessment; the case study is detailed in Section V, in terms of validation environment and results; and, finally, conclusions and future works are on Section VI.

II. Related Work

Recent successful cyber attacks on SCADA networks demonstrate that CIs, such as power distribution grids, wastewater treatment units [7] [8] or even nuclear fuel processing plants [9] are jeopardized by digital malicious agents. SCADA networks - and ICS in general - frequently constitute vulnerable targets due to their design philosophy and their technologies, primarily oriented towards reliability. In fact, the ICT and the ICS domains are significantly different in terms of their fundamental operational and functional properties: while the latter privileges availability and reliability over confidentiality and data integrity, the former is exactly the opposite [10].

Consequently, cyber threat detection within ICS necessarily requires a domain-specific approach. Despite this conceptual difference, adopting solutions conceived for the ICT domain is frequently necessary, something that must be dealt in a suitable way, mainly for three reasons: 1) some components have to work uninterrupted, on a 24/7 basis [11]; 2) every software release must be carefully tested by equipment manufacturers before being released; and 3) security mechanisms must impose a minimal overhead on the protected ICS. In face of this situation, several ICS managers ultimately adopted improper systems life-cycle management practices, disregarding regular updates or patching [12], therefore increasing the probability of a successful attack. Also, the real-time nature of some SCADA systems discourages the use of conventional inline network security mechanisms such as firewalls with deep packet inspection capabilities or Network Intrusion Detection Systems (NIDS), as they constitute an unwanted point-of-failure that might also degrade latency.

There is a considerable amount of work regarding SCADA intrusion and anomaly detection, including Intrusion Detection Systems (IDS) for embedded platforms, such as [13], device-level anomaly detection [14] [15] and classification [16]. However, not all solutions are equally feasible: for instance, device-level anomaly detection requires adding hardware mechanisms for probing. Moreover, dealing with attempts to cause loss of process visibility via Man-in-the-Middle or process-level attacks (though reprogramming or by inducing device behavior deviations) is out of scope for these mechanisms.

Recognizing the need to improve upon existing security solutions for SCADA ICS, the proposed framework was set up with the objective of addressing both the problems of CI interdependency and security, in an integrated fashion, building upon the past experience of the MICIE [17] and CockpitCI [18] projects. The development and implementation of a Cyber-Security detection framework for SCADA ICS will be described and analyzed with details in Section III. This detection framework improves upon the state-of-the-art, by adding diversified and specific analysis mechanisms and detection probes. Among its innovative features, which will be next described into more detail, one of the most relevant has to do with the awareness about the physical processes behind the SCADA ICS, that must be considered as an important feature of the overall system.

However, detecting cyber attacks is not enough if we want to improve operators' situation awareness and their time response. The assessment of consequences is a fundamental step for providing meaningful information to CI operators.

Consequence assessment on heterogeneous infrastructures is a complex task usually related with CI simulators and with interdependency modeling techniques. In [19], the authors survey the principal methods for modeling and simulation of CIs. Their report reveals that most of the approaches for dealing with infrastructure interdependencies, cascading system failures and risk mitigation are complementary, rather than competing.

The majority of simulators employ the agent-based paradigm, in which a population of autonomous interacting agents coordinate their decisions to reach a higher-level global objective. [20] Each infrastructure is modeled as an agent, with interdependencies being modeled as edges between agents. This enables agents to exchange information: each agent receives inputs from other agents and sends its outputs to other agents (see Nieuwenhuijs *et al*. [21] for further details). In this paper, CISIApro (Critical Infrastructure Simulation by Interdependent Agents) simulator [22] employs the agent-based paradigm, where each agent has a high-level description of the internal dynamics of an infrastructure. The main goal of CISIApro is to study the propagation of faults/attacks and the resulting performance degradation [23]. CISIApro exploits a specific metrics similar to a risk assessment one: the operational level. In [24], the authors survey the most used approach for risk assessment. CISIApro architecture will be described and analyzed in details on Section IV.

This article also analyses how increasing situation aware-

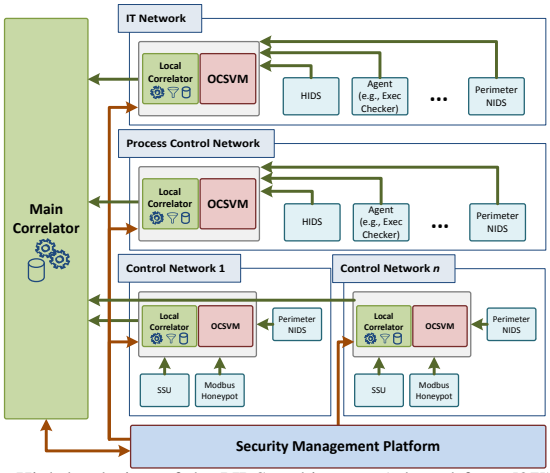

Fig. 1. High-level view of the PIDS architecture (adapted from [27])

ness improves the response capabilities and the decision process. As demonstrated in [25], including data on the propagation of an adverse event and their consequences on equipment and services improves the resilience and the response time of the considered infrastructures through suitable mitigation algorithms. The output of CISIApro can be included in an optimization algorithm for improving the decision process considering also events that are not within the infrastructure. The proposed architecture in this article is based on the previous works of data fusion frameworks [26].

III. Developing Cyber Attack Detection Capabilities

One of the principal goals of the architecture consisted of implementing capable cyber detection capabilities for SCADA ICS. Such mechanisms constitute a Dynamic Perimeter Intrusion Detection System (PIDS) [27], which is responsible for the continuous CI security auditing and monitoring.

The PIDS architecture (see Fig. 1) is organized along the three different zones of the CI, each one with its own security scope: the Control Network, SCADA Process Control Network, and the IT Network. Components and security policies are customized to suit the characteristics and requirements of each network domain, whose perimeter is monitored by a Network IDS (NIDS), configured according to its network traffic profile.

PIDS detection agents are deployed within each CI domain, feeding the information stream from which the security status of the CI is inferred. These agents provide diversified detection capabilities, encompassing customized third party modules, as well as security components especially developed for this project. The latter include network probes (such as NIDS), IT [28] and SCADA [29] honeypots, host-level agents (such as Host IDS-HIDS) and device monitoring components (like the Shadow Security Unit - SSU), among others.

The information fed by the detection agents is processed by a hierarchically distributed multi-zone correlation architecture [27]. The initial processing stage is provided by the local zone correlators, which also perform event reduction, filtering and aggregation. By their turn, local correlators feed a main correlator that has a global view on the CI security status. This approach provides context separation and allows for improved efficiency and scalability for event processing. In this way, the main correlator can focus on inter-scope event correlation and root cause analysis, as well as alert prioritization.

Correlation is complemented by machine-learning capabilities, in the form of One-Class Support Vector Machine (OCSVM) [30] anomaly detection modules [27]. The OCSVM analysis components, which were especially developed for the PIDS [31] are fed with real-time network traces, being deployed in the three network zones - for this reason, each module requires different training sets. Such modules are capable of detecting potentially relevant security events (abnormal or deviant behaviors), which are sent to the main correlator.

The segmented architecture of the PIDS encourages the selection and deployment of the security agents which better suit the specific characteristics of each CI domain, also allowing to fine-tune the configurations for agents, local correlators and OCSVM modules accordingly with their context. Moreover, analysis components are fed with zone-specific topology information provided by network and asset management systems.

The operation of the PIDS is coordinated through a Security Management Platform (SMP) which has a dual role: it provides the mechanisms for managing the PIDS components (via an out-of-band interface or secure channel) as well as the monitoring of in-place security and vulnerabilities within the network. The SMP is also responsible to feed other architecture components (i.e., CISIApro for risk assessment) by retrieving relevant information from the detection layer.

The event messages which flow between the PIDS components are encoded using the Intrusion Detection Message Exchange Format (IDMEF - RFC4765) [32], a platform-neutral and extensible format. Event passing is supported using Advanced Message Queuing Protocol (AMQP) [33] message queues, providing secure and reliable transport, while offering high-availability capabilities.

A. Innovative Detection Agents

The PIDS encompasses several kinds of detection agents, including existing components (such as the Snort NIDS [34] or the OSSEC HIDS [35], which are customized and integrated using coupling adaptors) as well as components specifically developed for this architecture, like the SCADA Honeypot [29] [36], the Shadow Security Unit (SSU) [37], Host Output Traffic Control, or the Exec, Vulnerability and Behaviour checker agents [27]. Among these agents, the first two constitute innovative concepts for domain-specific components that will be next discussed in more detail, from a research point of view.

1) SCADA Honeypot: The SCADA honeypot [29] [36] is a security agent designed to operate in the control network of a SCADA ICS system, binding to the unused IP addresses, coexisting with devices that populate this scope (such as PLCs (Programmable Logic Controller) and sensors/actuators). It emulates the operation and service footprint of a commercial Modbus PLC, luring attackers and reporting any interactions. It is a cost-effective proposal, being designed to run on a SBC (Single Board Computer) or within a virtual machine.

The SCADA honeypot is based on a hybrid architecture (see Fig. 2), in the sense that it runs both simulated and com-

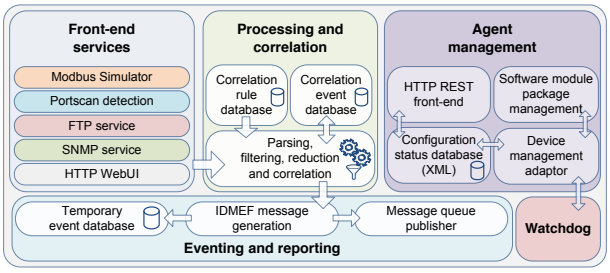

Fig. 2. The SCADA Honeypot architecture

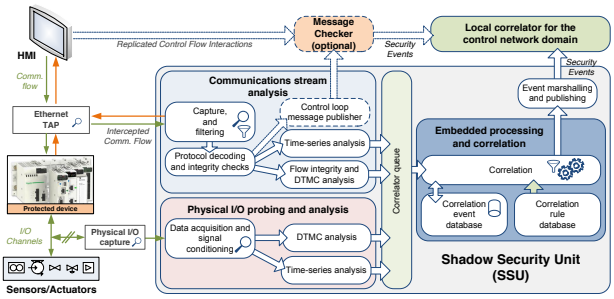

Fig. 3. The Shadow Security Unit operation workflow (detection mode)

plete implementations of services commonly available on PLC devices. Its components include a fully functional Modbus TCP emulator, providing an entire device instance, including variables (registers) and functions, enabling an attacker to interact with it. The *Front-end services* module also includes HTTP WebUI (web management interface), File Transfer Protocol (FTP) and Simple Network Management Protocol (SNMP) components, corresponding to services commonly found in various Modbus PLCs. These are further complemented by a Portscan service module, capable of capturing network interactions to detect anomalous activity, by listening to the TCP/IP ports not in use by other service modules.

Other components include a *Processing and correlation* module, which parses and processes alerts from the front-end services, also performing filtering and event aggregation. Security events are IDMEF-encoded and published on a local correlator queue by the *Eventing and reporting* module. Finally, an *Agent management* module provides management services via an out-of-band interface, working together with a *Watchdog* component to improve platform resilience.

2) Shadow Security Unit (SSU): As a result of several vulnerabilities, ranging from firmware bugs to communications protocol weaknesses, various PLC and RTU devices are known to be exposed to attacks such as flooding, protocol tampering or buffer overflow exploits. While different approaches have been proposed to deal with this situation, by using encrypted communications or mutual authentication mechanisms, their deployment is often unfeasible, due to latency overhead issues [11], reliability concerns or the need for modification of established protocols and architectures.

This situation has prompted the development of a security solution for PLC/RTU devices, capable of providing ongoing monitoring capabilities with minimal overhead: the Shadow Security Unit (SSU) [37] (see Fig. 3). Working in parallel with RTUs/PLCs and requiring few modifications on existing setups, the SSU is able to transparently monitor both the SCADA command flow and the state of the physical process interfaces. It should be noticed that the SSU is deployed out of the critical control path, and minimizes any potential impact on the monitored device (for instance, due to a malfunction).

The fundamental SSU operation principle follows a two-stage model comprising a training and a detection phase. The training phase is started once the SSU is first deployed, with the purpose of establishing the SCADA protocol and I/O models for the normal PLC/RTU operation state. Afterwards, the SSU will go into detection mode, using the models built during training to perform runtime analysis of acquired data.

The SSU embeds several techniques allowing it to detect threats such as layer 2 network attacks, protocol tampering or process disruption attempts – moreover, it can also provide information on the operational/health status of the monitored device. The SSU is also able to replicate copies of communication control flows, for consumption by an optional Message Checker system (see Fig. 3). This system creates a closed loop between the SCADA Human-Machine Interface (HMI) and the PLCs, for transaction integrity validation and tampering checks, at the communication endpoints.

Fig. 3 depicts the operation of the SSU prototype, designed for Modbus/TCP PLCs. The *Communications stream analysis* stage captures and processes the PLC network interactions, acquired via an integrated Ethernet TAP, which is placed between the monitored device and the upstream communications link. Modbus protocol interactions are modeled using first order Discrete Time Markov Chains (DTMC) to create probabilistic automata models (built during training), used for behavior analysis. This stage also performs network flow anomaly detection, via time-series analysis.

The *Physical I/O probing* stage assesses the state of the physical inputs and outputs of the PLC. It works together with a data acquisition hardware component consisting of several differential voltage probes (one for each I/O channel) coupled to an 8 channel, 10-bit Analog to Digital Converter. A software module processes and timestamps the captured data, for time series (on analog I/O) and DTMC analysis (for discrete I/O).

The outputs of the communications and I/O probing stages are fed to a *Local processing and correlation* stage. Its role is manifold, allowing to relate anomalous SCADA protocol interactions (such as low-probability opcode sequences) with physical I/O behavior deviations, within a time window, or to validate authorized device source addresses (using Access Control Lists - ACLs).

Finally, eventing, management and watchdog components (not depicted in the figure) are similar to the ones from the Modbus Honeypot architecture.

B. PIDS Detection Capabilities

By encompassing both a diverse number of heterogeneous probes and analysis techniques, the design of the cyber-detection layer was conceived in order to provide comprehensive detection capabilities. Such capabilities provide the means to cope with several attack scenarios, namely:

1) ***Layer 2/3 communications network attacks****:* these include protocol-level scanning/scouting attempts, attack on packet/frame integrity or denial-of-service attacks. Such scenarios are covered by an array of network probes such as

the signature-based and anomaly detection NIDS, as well as analytic techniques supported by information provided by agents for counter and event extraction from the network management layer. Scanning attempts can also be detected with the help of honeypot probes, which play in important role in providing profiling information about the attacker strategy and skill level.

*2) **SCADA Protocol/Service-level attacks**:* this class encompasses Man-in-the-Middle (MITM)-based scouting attempts, function code scan attacks, abuse of protocol specifications, or vulnerability exploitation. These are covered by an array of network probes such as the signature-based NIDS and also the SSU. Honeypot probes are also effective to detect scouting attempts for this attack category.

*3) **Process Level/Semantic attacks**:* this category includes advanced MITM scenarios with process-specific manipulation, direct process manipulation, interception/fuzzing or device reprogramming. The SSU was designed specifically to deal with these scenarios, being deployed in the cyber-physical edge of the infrastructure to gather information about the network-side interactions and physical I/O state changes.

*4) **Host and Server-level attacks**:* these threats are monitored by HIDS probes, as well as other components such as the software checker or behavior checker. These provide OS-level behavior and configuration change detection, including out-of-band binary analysis capabilities which allow the PIDS to monitor with the host-level context of the infrastructure.

This list is not exhaustive, being organized by categories in order to provide a high-level overview of the PIDS detection capabilities regarding each type of threat. More details about how the PIDS operates in the presence of specific attacks are available in [27].

IV. Mitigating Risk using Interdependency Modeling

CI operators need additional information on the consequences of cyber attacks on devices and on service performances, to properly react to malicious events. The first important step is the detection of a cyber attack and later on, the mitigation of its consequences on the system. Notice that, the interdependencies are evident during the domino effects, but they can be also exploited during reaction and mitigation strategies. To this aim, the following section describes an agent-based interdependency simulator called CISIApro [22], developed for research purposes and validated in several European projects ([17], [38]) for understanding the cascading effects of faults and/or threats, such as cyber attacks, in ICS and in CIs. Due to the fact that the evolution of the system states is usually governed by complex dynamics (often non-linear), CISIApro simulator must consider the underlined mathematical model during the interdependency analysis. In this section, we introduce the mathematical concepts implemented in CISIApro to consider infrastructures interactions and interdependencies. Thanks to graph theory and to Multilayer Networks modeling, our agent-based simulator is able to capture the system behavior both in anomaly situation (such as single fault, natural disaster or cyber attack) and in normal state where each component is able to produce and delivery outputs following the system model.

A. Multilayer Networks

We can formally define a graph (i.e., a single-layer network) as a tuple $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E} = \mathcal{V} \times \mathcal{V}$ is the set of edges that connect pairs of nodes. Two nodes are called adjacent, if there exists an edge between them.

In order to model Critical Infrastructure dynamics, we need to enrich the graph description of the system, considering the structure represented by the *layers* of the network, in addition to classical nodes and edges entities. Using the formalism of the multilayer networks [39], a complex system with d different types of layers is usually indicated as $\mathbf{L} = \{\mathcal{L}_a\}_{a=1}^{d}$, where other variables can be used to indicate whether a node is present in the considered layer.

We firstly construct a set $\mathcal{V} \times \mathcal{L}_1 \times \cdots \times \mathcal{L}_d$ and then define a subset $\mathcal{V}_M \subseteq \mathcal{V} \times \mathcal{L}_1 \times \cdots \times \mathcal{L}_d$ containing only the corresponding node-layer combinations. Let u and α_i be the considered node and layer respectively, then $(u, \boldsymbol{\alpha}) \equiv (u, \alpha_1, \ldots, \alpha_d)$ represents the set containing the topological connection (u, α_i) between node u on the layer α_i.

We can now introduce the edge set $\mathcal{E}_M \subseteq \mathcal{V}_M \times \mathcal{V}_M$, defined as the set of all possible combinations of node-layers. It should be noticed that different connection can be considered through this set, such as self node connection in different layers as well as multiple layer linking.

Finally, we can define a *multilayer network* as a quadruplet $M = (\mathcal{V}_M, \mathcal{E}_M, \mathcal{V}, \mathbf{L})$. Notice that a single-layer network is a special case of a multilayer network, where $d = 0$ and $\mathcal{V}_M = \mathcal{V}$ becomes redundant. In addition, given a subset $D \subseteq \mathbf{L}$ of the layers of the *multilayer network* M, a special set of nodes that can be reached by any edge starting from a generic node v from any of the layers in D, is called *neighborhood* and is formally defined as $\Gamma(v, D)$.

In what follows, CISIApro multilayer network structure is discussed and the interdependency risk evaluation modeling introduced.

B. CISIApro Structure

CISIApro is an agent-based simulator, where complex behaviors are modeled through the interaction of simple agents. Each agent has the same structure and is able to interacts with the environment and with other agents.

In general, the first two elements in a multilayer network M yield a graph $\mathcal{G}_M(\mathcal{V}_M, \mathcal{E}_M)$, so we can interpret a multilayer network as a graph whose nodes and edges are labeled in a certain way. We can easily say that a multilayer network M is directed if all the underlying graph $\mathcal{G}_M$ are directed. Mathematically, the $\mathcal{E}_M$ is an ordered set of edges, and therefore $((u, \boldsymbol{\alpha}), (v, \boldsymbol{\beta})) \neq ((v, \boldsymbol{\beta}), (u, \boldsymbol{\alpha}))$.

CISIApro structure is a directed multilayer network where each agent is a node, which appears in at least one layer but can also be included in all the layers. CISIApro also employs the usual convention of disallowing self-edges in the

multilayer network by preventing self-edges in the underlying graph, i.e., $((u,\boldsymbol{\alpha}),(u,\boldsymbol{\alpha})) \notin \mathcal{E}_M$.

CISIApro structure associates each agent with the set of nodes represented by the same entity in different layers. Therefore, within CISIApro structure, the coupling edges, denoted by $\mathcal{E}_C = \{((u,\boldsymbol{\alpha}),(v,\boldsymbol{\beta})) \in E_M | u = v, \forall u,v \in \mathcal{E}_M, \forall \boldsymbol{\alpha},\boldsymbol{\beta} \in \mathbf{L}\}$, are always present.

CISIApro structure defines a layer through a propagation or diffusion model among the nodes in the considered layer. A possible representation of a multilayer graph is depicted in Fig. 4. The multilayer network is composed of three layers, where the black dotted lines are the coupling edges, the red dotted lines are inter-layer edges $\mathcal{E}_{inter} = \{((u,\boldsymbol{\alpha}),(v,\boldsymbol{\beta})) \in E_M | u \neq v, \boldsymbol{\alpha} \neq \boldsymbol{\beta}\}$ and the other lines are the intra-layer edges $\mathcal{E}_{intra} = \{((u,\boldsymbol{\alpha}),(v,\boldsymbol{\beta})) \in E_M | u \neq v, \boldsymbol{\alpha} = \boldsymbol{\beta}\}$

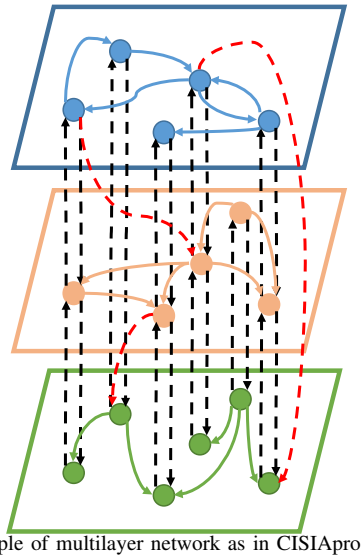

Fig. 4. An example of multilayer network as in CISIApro simulator

C. Discrete Dynamic Model

Each node $(u,\boldsymbol{\alpha})$ that appears in at least one layer of M has associated a status vector $x_u(t)$, which describes the evolution of the u component at time t. The status vector of each component is modeled as being governed by a non-linear discrete dynamical equation, where the status update for each component u is modulated using its internal state $x_i(t)$ and the data received from the neighbors.

Formally, the discrete-time nonlinear dynamics of the status vectors at time k are specified as follows:

$$\begin{aligned} x_u(t+1) &= g_u\left(x_u(t), y_{\Gamma^+(u,\mathbf{L})}(t), z_u(t)\right) \\ y_u(t) &= h_u(x_u(t), z_u(t)) \end{aligned} \tag{1}$$

where g_u and h_u are nonlinear functions, $z_u(t)$ represents the external input for the node u and $y_{\Gamma^+(u,\mathbf{L})}(t)$ the received data from the incoming neighborhood. The incoming neighborhood of the node u is defined as:

$$\Gamma^+(u,\mathbf{L}) = \{v \in \mathcal{V}_M | ((v,\boldsymbol{\beta}),(u,\boldsymbol{\alpha})) \in \mathcal{E}_M, \boldsymbol{\alpha},\boldsymbol{\beta} \in \mathbf{L}\} \tag{2}$$

Without loss of generality, we can stack both the status vectors x_{ui} in a *state vector* $\mathbf{x}(t) = [x_1(t) \dots x_u(t)]^T, \forall u \in \mathcal{V}_M$, and the inputs into an *input vector* $\mathbf{z}(t) = [z_1(t) \dots z_u(t)]^T, \forall u \in \mathcal{V}$. Hence, the resulting dynamical system can be rewritten as:

$$\begin{aligned} \mathbf{x}(t+1) &= \mathbf{g}\left(x(t), \mathbf{y}_{\Gamma^+(\cdot,\mathbf{L})}(t), \mathbf{z}(t)\right) \\ \mathbf{y}(t) &= \mathbf{h}(\mathbf{x}(t), \mathbf{z}(t)) \end{aligned} \tag{3}$$

where $\mathbf{g}, \mathbf{h}$ represent the column vectors of $g_u, \forall u \in \mathcal{V}$ and $h_u, \forall u \in \mathcal{V}$, respectively.

As pointed out in [39], the dynamical model defined in (3) is general enough to include all the classical approaches already defined in Multilayer Networks literature, such as percolation cascades or Susceptible-Infected-Recovered (SIR) models. In this context, agent-based algorithms or interdependency modeling, can then be used to represents complex interactions such as the effects of a Man-In-The-Middle attack in the Critical Infrastructure world.

D. CISIApro Implementation Details

As already pointed out in the previous sections, CISIApro is an agent-based simulator, where each agent has the same structure. In particular, each agent receives resources and failures from the upstream agents and spreads it to the downstream ones, following the dynamic model defined in Eq. 3.

The layers are obtained from the propagation of a resource or a fault. A resource is a service or a data produced and/or consumed by the agent, represented in CISIApro as an entity. The entity produces or receives also failures (in general, malfunctions) representing a physical failure or a possible cyber attack. The malfunctions are spread among the agents following different propagation models that take into account the class of the interdependencies (i.e., layers) and the reliability of the information. The considered layers are physical, logical, geographical and cyber.

The ability to produce resources is summarized by the concept of operational level, depending on the availability of received resources, on the propagation of faults, and on the functionality of the entity itself.

The operational level of each agent can be considered as a risk metric. Usually risk is a numeric value, from the impact severity, the likelihood of occurrence or threat, and the vulnerability analysis. In CISIApro applications, the likelihood of occurrence is usually considered more connected to the concept of trustworthiness of the information. For each entity, the user can add also a vulnerability variable, but in the following case study we suppose that the vulnerability depends only on the distance from the source and on the persistence of the attack itself. The operational level of each agent is associated to a risk level: the risk is the amount of harm due to specific events, such as a cyber attack, and can be evaluated as

$$Risk = 1 \; - \; OperationalLevel \tag{4}$$

where 1 is the maximum values of the operational level. A higher value of operational level means a lower risk. Therefore, the operational level represents a dynamic risk assessment considering the cascading effects of adverse events, i.e., natural disasters, failures or cyber attacks. This value is normalized for each infrastructure considering the quality of service towards

customers and other infrastructures. For a complete analysis of the implementation of CISIApro structure, we refer the reader to [40].

E. Mixed Holistic Reductionist Approach

The main approach used in CISIApro to model the interactions among infrastructures, considering all the underlying interdependencies, is the Mixed Holistic Reductionist (MHR) modeling technique [41].

This approach allows us to choose the right level between decomposition and abstraction of a complex system-of-systems to obtain meaningful information.

The MHR approach, proposed by [41], was created to exploit the advantages of holistic and reductionist methods. On one hand, in holistic modeling, infrastructures are seen as singular entities with defined boundaries and functional properties while, on the other hand, reductionist modeling emphasizes the need to fully understand the roles and behavior of individual components to truly understand the infrastructure as a whole. Different levels of analysis require one or both of the two points of view and their boundaries are lost in event of complex scenarios. With the MHR model, relationships among infrastructures could be seen at different levels through either a top-down or a bottom-up approach. A key element of operators is the Quality of Services towards customers. This analysis strengthens the addition of another layer, called service, describing functional relationships among components and infrastructures at different levels of granularity. In MHR, services to customers and to other interconnected infrastructures are explicitly considered as a middle layer between holistic and reductionist agents.

Notice that, CISIApro is the upgraded version of CISIA simulator, where the agent-based modeling has been maintained, but different functionalities related to the design phase has been added. In CISIApro, a Graphical User Interface is provided to quickly create entities and connect them easily (see [40]), considering the exchanged resources among agents.

The main advantage of this simulator is the great flexibility of the modeling approach: CISIApro is able to consider several interdependent infrastructures, electrical grid, gas networks, water distribution, telecommunications and rail road systems, as in [25], [42]. The operational mode can be both centralized and distributed. In the latter case, a CISIApro simulator must be located in each control center and subsequently synchronized using aggregated data, such as the operational level of services.

Inputs data of CISIApro are usually provided by the SCADA control center, and describe the state of the physical plant or, thanks to the PIDS (see Section III), the effects of the cyber attacks on the overall system.

The output of CISIApro can be used to help operators in decision making process: the results show how faults, cyber attacks and natural disasters affect the modeled infrastructures in terms of equipment and performances. In addition, thanks to specific software adapters, the output of CISIApro can be exposed as a service to feed other processes, such as Decision Support Systems (DSS) or other Risk-assessment tools. For an example of this architecture, please refer to [25], [42].

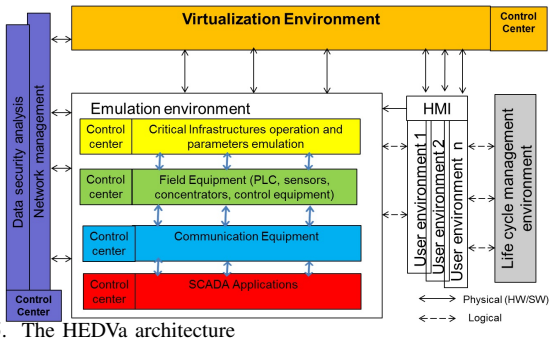

Fig. 5. The HEDVa architecture

V. Simulation and Results

This section describes the validation scenario instantiated within the Hybrid Environment for Development and Validation (HEDVa) testbed.This scenario is used to demonstrate how CISIApro is able to collect information from the SCADA control systems and the CI PIDS to evaluate risk on the single components and on the main service of the case study, suggesting better reaction strategies to operators. For this purpose, a Man-In-The-Middle cyber attack was implemented on the HEDVa, also providing the means to showcase PIDS detection capabilities.

A. Hybrid Environment for Development and Validation (HEDVa)

The HEDVa (see Fig. 5). was designed by the IEC (Israel Electric Corporation) for development and validation of Industrial Control Systems, Internet of Things and data security research projects, constituting a distributed environment with multi-tenant capabilities that allows the simultaneous coexistence of different lab environments. It provides the ability to emulate operation scenarios based on real SCADA and Network Management Systems (NMS), encompassing both process emulation and integration of physical components (hence its hybrid nature).

Available HEDVa resources include several categories, such as: virtual machines (VM), virtual or physical network routers and switches, virtualized networks (accommodating different topologies for each environment), storage, PLCs or RTUs, SCADA HMI applications, CI emulators, life-cycle management components and tools for requirement and validation management. All these resources are aggregated into a "Development Pool", providing the building blocks for the implementation of CI labs within the HEDVa.

Each lab tenant, which is responsible for the implementation and maintenance of a specific environment, is able to allocate the resources needed to create its own use case scenario. For instance, the creation of a simple SCADA lab can be undertaken in three steps: planning and definition of the lab characteristics, including network topologies, physical asset list (such as PLCs) and required virtual machines (such as HMIs, Master Stations or Historian DBs, also including the instances needed to support experimental measurements); provisioning of physical/virtual assets and network topology overlays; and the implementation of the use case model within the lab environment instance.

In the scope of this framework, the HEDVa hosts the *smart validation environment*, an evolution of the scenario that was originally devised for the MICIE project. This concept integrates user requirements, use case modeling, advanced emulation of CIs and availability of historical data within a validation scenario that was instantiated on the HEDVa, using a mix of physical devices (such as PLCs), virtual machines (hosting HMIs or historian DB hosts) and network topologies. It provides the means to support the development and validation of mechanisms and models for cyber-attack detection and mitigation.

B. The Case Study

As a case study, two interconnected infrastructures are here considered: a medium voltage power grid and the SCADA telecommunication network.

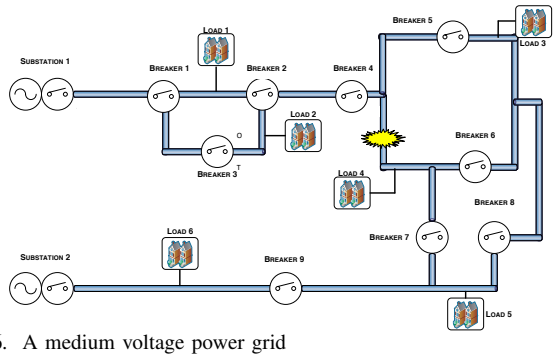

Fig. 6. A medium voltage power grid

The power grid is made of two lines fed by two substations to transform current from High Voltage grid to Medium Voltage network, see Fig. 6. The two lines are usually disconnected thanks to two circuit breakers (i.e., 7 and 8) that are normally open. The lines are radials, but their topology can change opening or closing several circuit breakers.

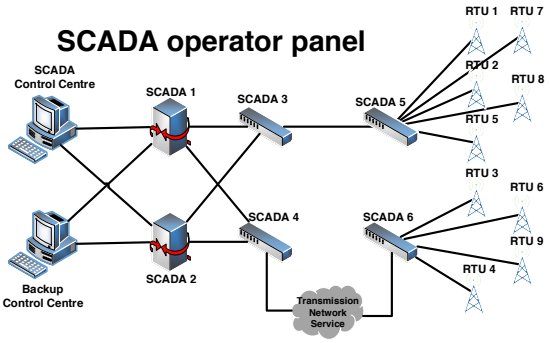

Fig. 7. The SCADA control network

The circuit breakers, except the two inside the substations, are telecontrolled from the SCADA control center by means of the telecommunication network, Fig. 7. Each telecontrolled circuit breaker has an associated PLC which is able to transmit data to the control center about actual state and alarms, and receive data from the control center about opening or closing circuit breakers.

In case of a permanent failure on the power grid, the operator executes a Fault Isolation and System Restoration (FISR) procedure, opening and closing the circuit breakers. This procedure, firstly, detects where the fault happened and, secondly, restores the current to customers as soon as the damage is repaired. If a cyber fault happens in the SCADA telecommunication network, the FISR procedure fails with unpredictable consequences.

C. Man-in-the-Middle with ARP Poisoning as Example of Cyber Attack

A Man-in-the-Middle (MITM) [43] [44] attack corresponds to a situation where a third-party becomes involved in the middle of the communication stream, while remaining unnoticed. For instance, an attacker may fool an HMI by directly interacting with it and providing normal process data (obtained from a network traffic capture and later used for a reply attack) while attacking the PLC in the background. MITM attacks can be implemented using several techniques, ranging from ARP poisoning [45] to routing redirection [46]; Fig. 8 illustrates the first scenario.

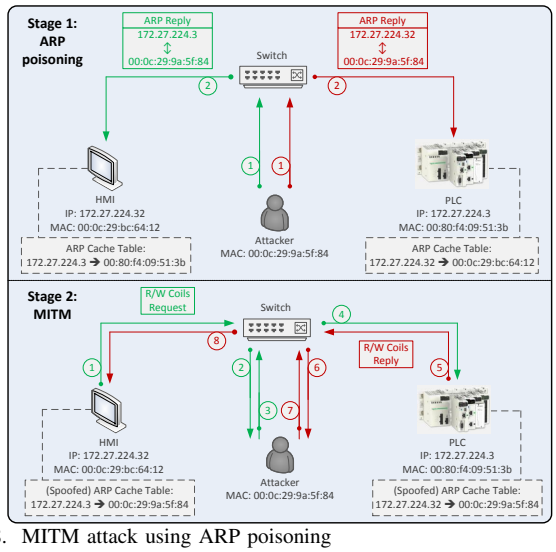

Fig. 8. MITM attack using ARP poisoning

In the first stage of the ARP poisoning MITM, the attacker generates a series of unrequested ARP replies for both the HMI and the PLC (top half in Fig. 8), poisoning the local ARP caches in such a way that the MAC address of the attacker system becomes associated with the IP of the HMI for the PLC and the IP of the PLC for the HMI, respectively. Further interaction attempts from the HMI to the PLC will be redirected to the attacker system, and vice versa. In a second attack stage, connections are intercepted in realtime using a packet manipulation tool (such as SCAPY [47]) to perform session hijacking on the TCP connection. Afterwards, the attacker may provide a fake device for the HMI to interact with, using a Modbus simulator programmed with information obtained from a previous survey or by replaying previously recorded protocol interactions, corresponding to a normal operation scenario.

1) Implementing a MITM on the HEDVa testbed: The use case that was implemented on the HEDVa consists of a hybrid energy grid testbed, in the sense that real PLCs

emulate breakers and substations, with a simulator calculating the voltage and current values for segments, accordingly with a mathematical model of the physical grid behavior. This approach combines the best of two worlds: the safety of using a simulated model, together with the benefits of using physical equipment to emulate control functions (hence the "hybrid" designation). The simulation model is also able to react accordingly with established operator reconfiguration FISR procedures. Figure 9 depicts the simplified logical network structure deployed on the HEDVa.

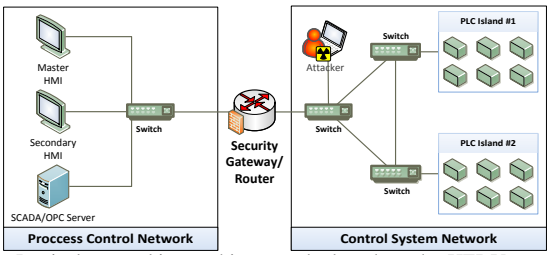

Fig. 9. Logical networking architecture deployed on the HEDVa

The testbed includes two HMI systems that interact with a set of PLCs - one for simulated grid supervision and control, providing a detailed view about the status of the breakers and substation feeders (the main HMI), and another one that mirrors basic functionality of the latter, for attack monitoring purposes. From a network point of view, each HMI continuously polls each PLC via Modbus/TCP. The network path between the control network (where the PLCs are) and the process control network (where the HMIs are) also includes several switches and a router/security gateway. The testbed was also equipped with a full PIDS deployment (not shown in the figure), including SSU units paired with the PLCs.

For validation purposes, an ARP MITM attack was prepared and executed on the HEDVa, with the purpose of fooling the main HMI and make it loose process visibility, making the attack go unnoticed. The attacker, which had access to the Control Network, intercepts and hijacks HMI-PLC communications, reproducing the normal operation of the simulated grid. Afterwards, normal state data was fed back to the main HMI while directly manipulating PLCs, without any visible supervisory feedback on the main HMI. The attack was accomplished and tested with the PIDS disabled, in order to establish the vulnerability level of the HEDVa testbed.

The first step, consisting on the redirection of the normal traffic from the main HMI was accomplished by sending ARP spoofed messages to impersonate the IP–MAC address associations. The attacker sent unicast unsolicited ARP-Reply messages to the PLCs and the router interface on the Control Network (the HMIs are on a different LAN segment), informing that the HMI and PLC IPs are respectively associated with the attacker MAC address. In order to keep the ARP cache of the end points spoofed, this needed to be done continuously during all the traffic interception. This enables the Modbus stream to be redirected through the attacker, where all traces are recorded. Despite the fact that the information stream was not modified or manipulated, this step provided the means for the attacker to eavesdrop all the communications and scout for information about the controlled processes.

In the second part of the attack, the intruder blocked all the requests from the main HMI to the PLCs while redirecting the HMI interaction to a scripted pseudo-simulator crafted with SCAPY, operating inline and in real-time. In order to create a successful attack, the attacker had to properly handle all the TCP operations, including connection establishment and termination, not forgetting acknowledgment and keep-alive related messages. For this purpose, it had to calculate and craft TCP header fields like the sequence and acknowledgment numbers. In addition, the attacker also needed to handle Modbus read and write operations, forging the replies using the same transaction field number from the requests and using the SCADA register values corresponding to a normal state.

Since the grid state is the result of the coordinated operation of all PLCs, the attacker scripts are able to respond to HMI operations and react accordingly, by simulating the correct reaction from the PLCs – as such, when the HMI user operates a specific breaker (managed by a PLC), the entire view is updated accordingly, including the energy values of affected segments. This is a delicate procedure, requiring all steps to be performed on real time, because otherwise the operator or even the system itself may trigger an alarm if a loss of a TCP connection, a malformed Modbus packet or even an inconsistent behavior in the SCADA scenario are detected.

2) ARP MITM detection by the PIDS: The HEDVa hosts a complete deployment of the PIDS, which was activated once the execution of the use case attacks on the existing infrastructure was proven to be feasible. During platform validation, several PIDS components demonstrated their effectiveness for ARP MITM attack detection, namely:

- The Perimeter NIDS, which detects network traffic involving a station that is not part of the topology extracted from asset management systems, together with fake ARP replies from an unknown host, following an unusual teletraffic pattern (inter-arrival rates are too small for normal operation thresholds). Such events are reported to the local correlator for the network domain.
- The OCSVM on the Control Network, that detects an abnormal traffic pattern (ARP packets with rate above the learned alarm threshold), generating an event to the local correlator for the network domain.
- The SSU units, due to several capabilities: ACLs are configured with the IP/MAC addresses of the systems that are authorized to interact with the monitored PLCs; the SSU network traffic analysis modules are able to detect unusual patterns (an excessive ARP reply packet rate); the high-level command flow processing capabilities are able to detect unusual command patterns that do not correspond to the normal operation sequences; finally, the message checker, which was deployed on the HEDVa, was able to detect inconsistencies between the main HMI interactions and Modbus commands arriving at the PLCs.

Once the evidence gathered by the network domain local correlators reaches the global correlator, a chained rule match is triggered, generating an alarm that is sent to the Integrated Risk Predictor, namely CISIApro.

Complementary to the PIDS, good security policies can also be extremely effective against ARP poisoning attacks, such as the use of static ARP lists or managed switches featuring port security or dynamic ARP inspection mechanisms (which a large number of SCADA operators do not use, nevertheless). Moreover, even if such measures are deployed, the PIDS remains effective against other types of MITM attacks, such as Spanning Tree Protocol [48] or routing redirection attacks.

D. CISIApro Results

In order to properly demonstrate the ability of CISIApro, we depicted the results of the experiment, which lasted 40 seconds in a real scenario validation and which is divided in two parts. The first part, lasting from seconds 1 to 10, involves the attacker performing a man-in-the-middle attack on the SCADA network, (Fig. 7) as described on previous sections. The second part, lasting from seconds 11 to 40, involves an infection being spread from the attacker in the aftermath of the MITM attack. The malware reaches a subset of four PLCs in the HEDva (see Figure 5), numbered 3, 4, 6 and 9, in Fig. 7. Those PLCs are physically linked to one electric switch with the same numbering, see Fig. 6. During the malware spreading, the ability to properly telecontrol power switches is downgraded and can not be guaranteed.

For the MITM attack, the spreading rule is related to the distance of the infected node: the greater is the number of hops needed for reach the node, the lower are the effects of the cyber attack and the risk of node malfunction. In our simulation, the operational level of the attacked node (SCADA nodes no. 6 (in Fig. 7) is 0.9, and it is the same for the set of PLCs connected to node no. 6.

For the malware spreading, the propagation is still related to the distance, but each node has an increasing exponential trend for the effect of the malware. The exponential function and its parameters have been obtained starting from literature reviews, expert interviews, historical data (if existing) and from some simulations, and then extracting the best fitting pattern from all available information.

When the malware is detected at 11 seconds, the SCADA telecommunication node is highly affected and the information has a high trustworthiness, see Fig. 10.

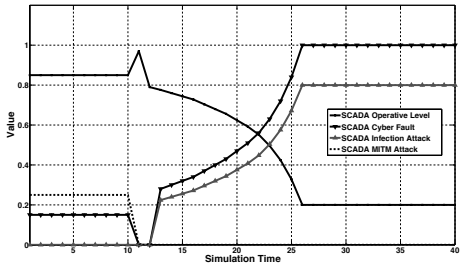

Fig. 10. Operational level of SCADA node number 6

The trends of the entities are related to the distance and therefore the node needs more time to become completely unavailable. The set of PLCs (number 3, 4, 6, and 9 in Fig. 7) linked to the SCADA node no. 6 has a similar trend of the up-stream node with a delay of one time step, see Fig. 11.

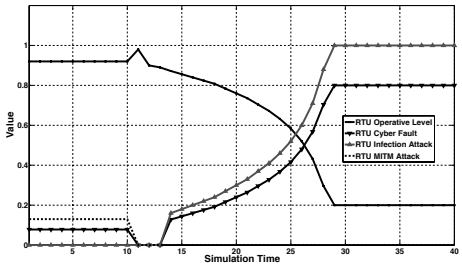

Fig. 11. Operational level of a subset of PLCs (numbers 3, 4, 6, 9)

The main aim of the CISIApro simulator is to help the decision making of the operator. The reconfiguration procedure of an electrical grid is a very easy and common task for the operator, but requires interconnected infrastructures. Supposing a fault in the power grid, depicted in Fig. 6 as a yellow explosion, two alternative configuration are considered: FISR no. 1 (opening breakers no. 4 and no. 6; breakers no. 7 and no. 5 are already open; the only disconnected customer is number 4; load number 3 is fed from the substation no. 2) and FISR no. 2 (opening breakers no. 4 and no.6; customers no. 4 and no. 3 are isolated).

The two reconfiguration procedures are affected differently due to infection spreading: the first FISR is less risky than the second one because involves PLCs that are not affected by the malware. Therefore, the platform is able to suggest the less-risky reconfiguration option, in order to improve electrical operator readiness in case of cyber attacks, where quick response time is mandatory.

In more complex scenarios, where several configurations are possible, the network reconfiguration algorithm can be implemented for finding all the possible configurations. Then, they are ranked using a multi-criteria decision making algorithm that mimics the operator behavior. The multi-criteria decision making algorithm [42], [49] is a heuristic for multi-objective optimization problems that rank different alternatives (i.e., configurations) based on several criteria. One of the reason why we choose this method is related to the ability to change ranking based on different priority of the operator, by means of changing the criteria weights.

E. Effectiveness of the Proposed Platform

The cyber detection platform has been designed in order to recognize several and heterogeneous possible threats within the SCADA ICS. The list of detectable attacks contains among the others: man-in-the-middle, denial-of-service, worms, trojans, device impersonation and non-authorized tempering. The architecture can be deployed in different contexts with different possible configurations that are due to specific constraints of the SCADA ICS infrastructure. The configuration of the cyber detection capability is usually defined by the installed probes on network.

The results coming from CISIApro are presented to the operator using a Graphical User Interface (GUI). Thanks to the interactions between the cyber detection layer and the mitigation one, we tested different scenarios provided by the stakeholders (IEC corporation): best and worst cases in terms

of response time for services to customer restoration have been evaluated, and the overall performances increased around 50% respect to classical approaches.

VI. Conclusions and Future Works

The main contribution of this paper consists of an unique framework which enables to increase the CI operator readiness in critical situations. In order to describe its operation, a complete functional information flow description is provided, from detecting a cyber attack up to evaluating consequences on equipment, in order to suggest a different re-configuration strategy. The IDMEF events are exchanged in order to transmit updated messages on the actual state of the physical system (from SCADA control centers) and of the cyber attacks (from the detection platform). All the messages are collected by CISIApro, and the interdependency model is able to provide the propagation on physical device and on services. For evaluation and validation purposes, the authors considered a reconfiguration service realized by electrical operators, for which the proposed framework was able to suggest the minimum risk decision.

Ongoing developments include a continuous update of all the described platforms:

- Complementing the detection platform with active reaction capabilities allowing to change the configuration of the telecommunication network in real-time, by taking advantage of the Software Defined Network paradigm;
- Improving the CISIApro simulator with other infrastructures and more detailed information on interdependencies; ongoing work is related to different decision support systems based on optimization algorithms for different purposes (i.e., natural disaster response, or energy demand/response balance);

References

[1] National Council on Public Works Improvement (U.S.), *Fragile Foundations :a Report on America's Public Works: Final Report to the President and the Congress*. The Council, 1988.

[2] U.S. Dept. of Homeland Security, "National infrastructure protection plan." Available online at: www.dhs.gov/nipp, 2006.

[3] M. Panteli and P. Mancarella, "Modeling and Evaluating the Resilience of Critical Electrical Power Infrastructure to Extreme Weather Events," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1733–1742, sep 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7036086/

[4] V. M. Igure, S. A. Laughter, and R. D. Williams, "Security issues in SCADA networks," *Computers & Security*, vol. 25, no. 7, pp. 498 – 506, 2006.

[5] N. Falliere, L. O. Murchu, and E. Chien, "W32. stuxnet dossier," *White paper, Symantec Corp., Security Response*, vol. 5, 2011.

[6] N. H. A. Rahman and K.-K. R. Choo, "A survey of information security incident handling in the cloud," *Computers & Security*, vol. 49, pp. 45 – 69, 2015.

[7] J. Slay and M. Miller, *Lessons Learned from the Maroochy Water Breach*. Boston, MA: Springer US, 2008, pp. 73–82.

[8] C. Choi. (2015, 10) Nuclear cybersecurity woefully inadequate. [Online]. Available: http://spectrum.ieee.org/energywise/telecom/security/nuclear-cybersecurity-woefully-inadequate

[9] W. Beckner, "Information notice 2003-14: Potential vulnerability of plant computer network to worm infection," *United States Nuclear Regulatory Commission*, vol. 14, 2003.

[10] ISA, ANSI, "ISA–99.00. 01–2007 security for industrial automation and control systems part 1: Terminology, concepts, and models," *International Society for Automation*, 2007.

[11] B. Zhu, A. Joseph, and S. Sastry, "A taxonomy of cyber attacks on SCADA systems," in *Proc. of the 4th International Conference on Cyber, Physical and Social Computing,*, 2011, pp. 380–388.

[12] R. L. Krutz, *Securing SCADA Systems,*. Wiley Publishing, 2006.

[13] E. Naess, D. Frincke, A. McKinnon, and D. Bakken, "Configurable middleware- level intrusion detection for embedded systems," in *Proc. of the 25th IEEE Int. Conf. on Dist. Computing Systems Workshops*, 2005, pp. 144–151.

[14] J. Rushi and K. D. Kang, "Detecting anomalies in process control networks," in *Proc. of the 3rd IFIP WG 11. 10 International Critical Infrastructure Protection Conference Conference*. Springer, 2009, pp. 151–165.

[15] J. Zaddach, L. Bruno, A. Fracillon, and D. Balzarotti, "Avatar: A framework to support dynamic security analysis of embedded systems' firmwares," in *Proc. of Net. and Distributed System Security (NDSS) Symposium*, 2014, pp. 1–16.

[16] R. Karri and J. Rajendran, "Trustworthy hardware: Identifying and classifying hardware trojans," *Computer*, vol. 43, no. 10, pp. 39–46, 2010.

[17] MICIE Consortium, "MICIE FP7-ICT-SEC-2007-1 225353."

[18] CockpitCI Consortium, "CockpitCI FP7-SEC-2011-1 285647."

[19] M. Ouyang, "Review on modeling and simulation of interdependent critical infrastructure systems," *Reliability Engineering & System Safety*, vol. 121, pp. 43 – 60, 2014.

[20] R. Setola, V. Rosato, E. Kyriakides, and E. Rome, Eds., *Managing the Complexity of Critical Infrastructures*, ser. Studies in Systems, Decision and Control. Cham: Springer International Publishing, 2016, vol. 90. [Online]. Available: http://link.springer.com/10.1007/978-3-319-51043-9

[21] A. Nieuwenhuijs, E. Luiijf, and M. Klaver, "Modeling dependencies in critical infrastructures," in *Critical Infrastructure Protection II*, ser. The International Federation for Information Processing, M. Papa and S. Shenoi, Eds. Springer US, 2008, vol. 290, pp. 205–213.

[22] "CISIApro: interdependency modeling and simulation made easy for critical infrastructures," http://cisiapro.dia.uniroma3.it, University of Roma Tre.

[23] S. De Porcellinis, R. Setola, S. Panzieri, and G. Ulivi, "Simulation of heterogeneous and interdependent critical infrastructures," *Int. Journal of Critical Infrastructures*, vol. 4, no. 1/2, pp. 110–128, 2008.

[24] Y. Cherdantseva, P. Burnap, A. Blyth, P. Eden, K. Jones, H. Soulsby, and K. Stoddart, "A review of cyber security risk assessment methods for SCADA systems," *Computers & Security*, vol. 56, pp. 1–27, feb 2016. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0167404815001388

[25] S. Imbrogno, C. Foglietta, C. Palazzo, and S. Panzieri, "Managing Decisions for Smart Grid Using Interdependency Modeling," in *2016 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA 2016)*, San Diego, USA, 2016, pp. 198–204.

[26] E. Shahbazian, D. E. Blodgett, and P. Labbé, "The extended OODA model for data fusion systems," in *Proceedings of 4th International Conference on Information Fusion*, pp. 1–7.

[27] T. Cruz, L. Rosa, J. Proenca, L. Maglaras, M. Aubigny, L. Lev, J. Jiang, and P. Simoes, "A cyber security detection framework for supervisory control and data acquisition systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2236 – 2246, December 2016.

[28] L. Spitzner, *Honeypots: Tracking hackers*. Boston: Addison-Wesley Professional, 2002.

[29] P. Simões, T. Cruz, J. Proença, and E. Monteiro, "Specialized honeypots for SCADA systems," in *Cyber Security: Analytics, Technology and Automation*, ser. Intelligent Systems, Control and Automation: Science and Engineering, M. Lehto and P. Neittaanmäki, Eds. Springer International Publishing, 2015, vol. 78, pp. 251–269.

[30] J. Ma and S. Perkins, "Time-series novelty detection using one-class support vector machines," in *Neural Networks, 2003. Proceedings of the International Joint Conference on*, vol. 3, July 2003, pp. 1741–1745.

[31] L. Maglaras, J. Jiang, and T. Cruz, "Integrated OCSVM mechanism for intrusion detection in SCADA systems," *Electronics Letters*, vol. 50, no. 25, pp. 1935–1936, 2014.

[32] H. Debar, D. Curry, and B. Feinstein. (2007, 3) The intrusion detection message exchange format. [Online]. Available: http://www.ietf.org/rfc/rfc4765.txt

[33] AMQP Working Group, "Advanced message queuing protocol," 2012.

[34] Cisco Systems, Inc., "SNORT network IDS," http://www.snort.org.

[35] "OSSEC: open source security," http://www.ossec.net, Trend Micro.

[36] P. Simões, T. Cruz, J. Gomes, and E. Monteiro, "On the use of honeypots for detecting cyber attacks on industrial control networks," in *12th European Conference on Information Warfare and Security (ECIW 2013)*, 2013, pp. 263–270.
[37] T. Cruz, J. Barrigas, J. Proenca, A. Graziano, S. Panzieri, L. Lev, and P. Simoes, "Improving network security monitoring for industrial control systems," in *Integrated Network Management (IM), 2015 IFIP/IEEE Int. Symposium on*, May 2015, pp. 878–881.
[38] FACIES Consortium, "FACIES HOME/2011/CIPS/AG/4000002115," http://facies.dia.uniroma3.it.
[39] M. Kivela, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *Journal of Complex Networks*, vol. 2, no. 3, pp. 203–271, sep 2014. [Online]. Available: http://comnet.oxfordjournals.org/cgi/doi/10.1093/comnet/cnu016
[40] C. Foglietta, C. Palazzo, R. Santini, and S. Panzieri, *Assessing Cyber Risk Using the CISIApro Simulator*. Cham: Springer International Publishing, 2015, ch. Critical Infrastructure Protection IX: 9th IFIP 11.10 International Conference, ICCIP 2015, Arlington, VA, USA, March 16-18, pp. 315–331.
[41] S. D. Porcellinis, S. Panzieri, and R. Setola, "Modelling critical infrastructure via a mixed holistic reductionistic approach," *International Journal of Critical Infrastructures*, vol. 5, no. 1-2, pp. 86–99, 2009.
[42] D. Masucci, C. Foglietta, C. Palazzo, and S. Panzieri, "Improved multi-criteria distribution network reconfiguration with information fusion," in *2016 19th International Conference on Information Fusion (FUSION)*, July 2016, pp. 256–263.
[43] L. Rosa, C. T., P. Simoes, E. Monteiro, and L. Lev., "Attacking scada systems: A practical perspective," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, May 2017, pp. 741–746.
[44] Y. Yang, K. McLaughlin, T. Littler, S. Sezer, E. G. Im, Z. Yao, B. Pranggono, and H. Wang, "Man-in-the-middle attack test-bed investigating cyber-security vulnerabilities in smart grid SCADA systems," in *Sustainable Power Generation and Supply (SUPERGEN 2012), International Conference on*, Sept 2012, pp. 1–8.
[45] R. Siles, "Real world ARP spoofing," *GIAC Certified Incident Handler (GCIH) Practical, Version*, vol. 2, 2003.
[46] A. Ornaghi and M. Valleri, "Man in the middle attacks," in *Blackhat Conference Europe*, 2003.
[47] P. Biondi, "Scapy project," http://www.secdev.org/projects/scapy, 2010.
[48] E. Vyncke and C. Paggen, *LAN switch security: What hackers know about your switches*. Cisco Press, 2007.
[49] D. Masucci, C. Palazzo, C. Foglietta, and S. Panzieri, "Enhancing decision support with interdependency modeling," in *Critical Infrastructure Protection X: 10th IFIP WG 11.10 International Conference, ICCIP 2016, Arlington, VA, USA, March 14-16, 2016, Revised Selected Papers 10*. Springer International Publishing, 2016, pp. 169–183.

Chiara Foglietta (M'11) is currently an Assistant Professor at the Department of Engineering, at the University of "Roma Tre", where she received is Ph.D degree in Computer Science and Automation in 2013. Her research interests are in the field of control of smart power systems, hierarchical state estimators for smart grids, and information fusion algorithms, especially Evidence Theory. She has been also involved in several EU projects.

Dario Masucci is a junior researcher at the University of "Roma Tre" since 2015, when he received his master degree in Automation. His research interests include Multi-Objective optimization algorithms and energy sustainability. In the last two years, he has been working on several European projects by developing and designing multi-criteria decision making algorithms, interdependencies models for Critical Infrastructures and emergency management tools.

Cosimo Palazzo is a Ph.D student in Computer Science and Automation at the Department of Engineering at the University of "Roma Tre" where he received is master degree on Automation in 2014. His research interests are in the field of Critical Infrastructures modelling (designing and implementing CISIApro), Decision Support Systems and Internet of Things (IoT). He has been involved in several EU projects, such as FACIES, URANIUM, FP7 CockpitCI and H2020 ATENA.

Riccardo Santini (S'15 - M'17) received is Ph.D degree in Computer Science and Automation at the University of "Roma Tre" in 2017. He is working as junior researcher at Models for the Critical Infrastructure Protection Laboratory (MCIP lab). His research activities are related to models, security and control of Cyber-Physical Systems, topology design and distributed control over networks, SCADA and Industrial Control Systems, mobile and industrial robotics.

Stefano Panzieri (M'93) is Associate Professor at the Department of Engineering at the University of "Roma Tre" since 1996 and he is the director of Models for Critical Infrastructure Protection Laboratory (MCIP lab). He received the Laurea degree in Electronic Engineering in 1989 and the Ph.D. in Systems Engineering in 1994, both from the University of Roma La Sapienza. His research interests are in the field of industrial control systems, robotics and sensor fusion.

Luis Rosa is a PhD student in Informatics Engineering at the University of Coimbra with research interests in Security, Event Management and Critical Infrastructure Protection. He completed his M.Sc. degree in Informatics Engineering at Higher School of Technology and Management of the Polytechnic Institute of Coimbra in 2013. Since then, he has been working as junior researcher at the Centre for Informatics and Systems of the University of Coimbra, where he participates in several research projects in those fields.

Tiago Cruz (M'13) is Assistant Professor at the Department of Informatics Engineering at the University of Coimbra since December 2013, where he obtained his PhD in Informatics Engineering, in 2012. His research interests cover areas such as management systems for communications infrastructures and services, critical infrastructure security, broadband access network device and service management, internet of things, software defined networking and network function virtualization (among others). He is member of the IEEE Communications Society.

Leonid Lev (SM'05) has received his Ph.D. degree in 1991 in design of computer based systems from the Ivanovo Energy Institute, Russian Federation (USSR). Dr. Leonid Lev holds the position of senior expert engineer in Israel Electric Corp. In his capacity he is responsible for system engineering, for conceptual design of command and control systems and for H2020 projects at IEC ICT division. He was technical leader and integrator in the projects of computer based systems for control and management of IEC telecommunication and electrical networks. Dr. Leonid Lev is an IEEE Senior Member.