Review of Security Issues in Industrial Networks

Review of Security Issues in Industrial Networks

Manuel Cheminod, Luca Durante, and Adriano Valenzano, *Senior Member, IEEE*

State of the Art Paper

***Abstract*—Although awareness is constantly rising, that industrial computer networks (in a very broad sense) can be exposed to serious cyber-threats, many people still think that the same countermeasures, developed to protect general-purpose computer networks, can be effectively adopted also in those situations where a physical system is managed/controlled through some distributed Information and Communication Technology (ICT) infrastructure.**

Unfortunately, this is not the case as several examples of successful attacks carried out in the last decade, and more frequently in the very recent past, have dramatically shown. Experts in this area know very well that often the peculiarities of industrial networks prevent the adoption of classical approaches to their security, and in particular of those popular solutions that are mainly based on a detect and patch philosophy.

This paper is a contribution, from the security point of view, to the assessment of the current situation of a wide class of industrial distributed computing systems. In particular, the analysis presented in this paper takes into account the process of ensuring a satisfactory degree of security for a distributed industrial system, with respect to some key elements such as the system characteristics, the current state of the art of standardization and the adoption of suitable controls (countermeasures) that can help in lowering the security risks below a pre-defined, acceptable threshold.

***Index Terms*—Network security, information security, industrial networks, risk assessment, security countermeasures, security analysis and monitoring.**

I. Introduction

INTERCONNECTION through digital communication networks is of primary importance, today, in many distributed heterogeneous environments where people and things, besides services and data, have to be protected against injuries and damages. This is the case, for instance, of critical infrastructures designed for energy, gas and water distribution, transportation systems and air traffic control but, even with different characteristics, the same is also true for other application domains such as Industrial Process Measurement and Control (IPCM), Supervision, Control and Data Acquisition (SCADA), Distributed Control (DC), Metering, Monitoring and Diagnostic (MMD), Networked Electronic Control and Sensing (NECS) and Distributed Automation (DA) systems. Although peculiarities can be identified for each scenario (and network), a set of common security characteristics exists, which allows us to consider these systems as belonging to a single, broad class. With a slight abuse of terminology, we will call this class either *Industrial Networks* or *Industrial Automation Control Systems* (IACS) [1] in the following, provided that no ambiguity could arise.

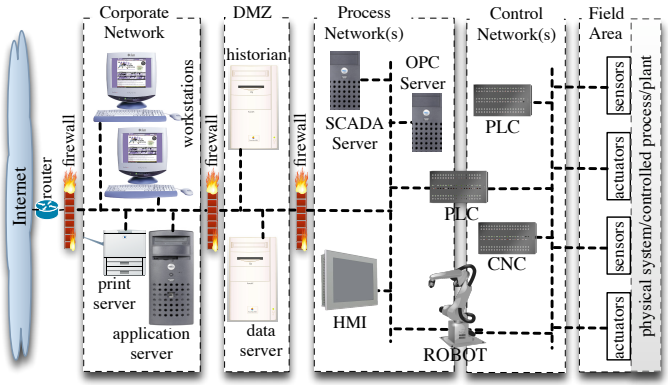


Fig. 1. Typical connections of IACS to corporate networks and the Internet

In the past IACS were mainly conceived as isolated systems, but nowadays, because of the ever growing demand of both highly ubiquitous computing services and location-independent access to ICT resources, they are more and more connected to all kinds of *Desktop and Business Computing Systems* (DBCS) [1] and often to the Internet, as Fig. 1 (inspired by [1] and [2]) shows for a typical situation.

In the case of the picture, the IACS communication infrastructure (three rightmost blocks) can access the Internet through a DBCS network: dashed lines inside each block may represent different kinds of media (i.e. Ethernet cables, phone lines, fiber optics, radio and WiFi links) and proper equipment (routers, gateways, modems, access points and so on). The key-point, however, is that the IACS infrastructure is directly interfaced to a physical system (i.e. the controlled process, automation plant and so on), through its sensors and actuators, while this does not occur in the case of DBCS. Fig. 1 also shows that two main different kinds of (sub)networks can be found in typical IACS, that is control networks responsible, for instance, for enabling the correct and effective behavior of regulation loops according to the system (even hard) real-time requirements, and process networks designed to support supervisory and management functions through SCADAs and other specialized software modules. It is worth remembering that, although process networks are less concerned with real-time than their control counterparts (soft-real time needs

Manuscript received —; revised —

The authors are with the National Research Council of Italy (CNR), Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni (IEIIT) I-10129 Torino, Italy (e-mail: adriano.valenzano@ieiit.cnr.it).

This work was supported in part by the European Commission in the framework "Prevention, Preparedness and Consequence Management of Terrorism and other Security Related Risks" Project INMOTOS "Interdependency Modelling Tools and Simulation-Based Risk Assessment of ICT Critical Infrastructures Contingency Plans" Grant Agreement No.: HOME/2009/CIPS/AG/C2-067, and in part by the IEIIT and Ferrero SpA project NEAT "Network Technologies for Automation".

TABLE I
MAIN DIFFERENCES BETWEEN IACS AND DBCS

	DBCS	IACS
	System Characteristics	
Average node complexity	high (large servers/file systems/databases)	low (simple devices, sensors, actuators)
Number of users	very high	limited
Multi-vendor environment	moderate	frequent
System lifetime (years)	some	some tens
Outage of system availability	often tolerable	rarely/never tolerable
Tolerability of time delays	medium/high	low/none (real-time)
Acceptable processing times	minutes÷days	milliseconds÷minutes
Tolerability of failures	medium/high	low/none
Communication protocol stacks	general purpose (i.e. TCP/IP, UDP)	special purpose/proprietary/real-time
Operating systems	general-purpose (i.e. Windows, Unix)	real-time, embedded, special-purpose
	System Maintenance and Upgrading	
S/w patches and upgrades	very frequent	rare or none
No longer supported s/w versions	rare	often in use
New s/w releases	frequent (extensive changes)	rare (small changes)
Frequency of h/w upgrades/changes	medium/high	very low/low
	Security Practices	
Security awareness	high/very high	usually low (rising)
Availability of security expertise	high/very high	very low/low
Adoption of security audits	frequent	very rare/rare
Online security checks	frequent	rare (often impossible)
	Security Countermeasures	
Use of anti-virus	heavy	rare/none (often impossible)
Physical protection	frequent (site protection and surveillance)	difficult (remote and not guarded sites)
Availability and adoption of firewalls and IDSs	frequent/very frequent	rare/sometimes impossible
	Impact of Negative Events (Cyber-Attacks)	
Losses	information, money	human lives, things, environment, money
Costs of successful attacks	bounded	unbounded
Pre-estimation of losses	possible	often impossible (i.e. human lives, environment damages)

have to be satisfied sometimes), nevertheless they often have to grant satisfactory performance in term of the maximum acceptable response time.

A demilitarized zone (DMZ) allows resources to be shared between the corporate and IACS networks without direct connections (a popular security technique) and under control of firewalls placed at the boundaries. The set of servers in Fig. 1 is only illustrative and not exhaustive, in that several others services, such as those supporting authentication and/or key management, can be present when needed.

The traditional isolation and some characteristics of IACS, such as the widespread adoption of special-purpose, proprietary hardware (h/w), software (s/w) and applications, were often sufficient to prevent them from being concerned with serious security problems affecting their ICT infrastructure (security by insulation and obfuscation).

In a modern scenario such as the one in Fig. 1, instead, the careful management of interconnections is mandatory, since accessibility and openness, besides introducing many appealing advantages, also expose IACS to the same security threats usually experienced by DBCS [3]–[5].

The main goal of this paper is to make an overall assessment of the current situation most industrial distributed computing systems are experiencing, with respect to security. To this purpose we consider the typical steps that have to be followed to ensure a satisfactory security level for IACS, and discuss the main elements involved in this process, such as the system characteristics, the current state of the art of standardization and the adoption of suitable controls (countermeasures) that can be employed to lower the security risks below a predefined, acceptable threshold.

In fact, despite they often share similar interconnection and communication technologies, IACS and DBCS also exhibit deep differences that cannot be ignored when dealing with security. Table I summarizes some main aspects that are relevant in this case and have to be taken carefully into account.

As the table shows in many respects, IACS and DBCS are usually very different from the architectural, management and maintenance points of view, and these differences are mainly due to the kinds of missions they have to support. The physical systems and processes that IACS control and supervise, in fact, put several constraints (for instance on the maximum reaction times, the level of safety to be granted and the maximum system unavailability which can be tolerated) that are either not present or largely ignored in conventional business and office distributed systems. Unavoidably, this also has an impact on the available design alternatives and heavily affects the choices for the h/w and s/w components, and for the underlying communication networks too. For instance, the operating systems and communication protocols IACS rely on, are more concerned with aspects such as real-time processing, jitters limitation and event-notification than DBCS, while devices have more simple h/w architectures and lower computing power in general.

Unfortunately, because of these and other peculiarities (see Table I), also the security scenario is significantly different in the two cases, as the lower part of the table shows. As an immediate consequence, the popular strategies and mechanisms developed to protect DBCS cannot be adopted in most IACS, so that new *information security* challenges have to be tackled in the latter case.

TABLE II
SECURITY REQUIREMENTS IN IACS AND DBCS

	IACS	DBCS
increasing priority ⇑	availability	confidentiality
	integrity	integrity
	confidentiality	availability

TABLE III
DIFFERENT CRITICALITIES BETWEEN IACS AND DBCS

	IACS	DBCS
h/w & s/w patching & upgrading	critical	not critical
real-time constraints	critical	not critical
consequences of failures	critical	not critical
performance & power	critical	not critical

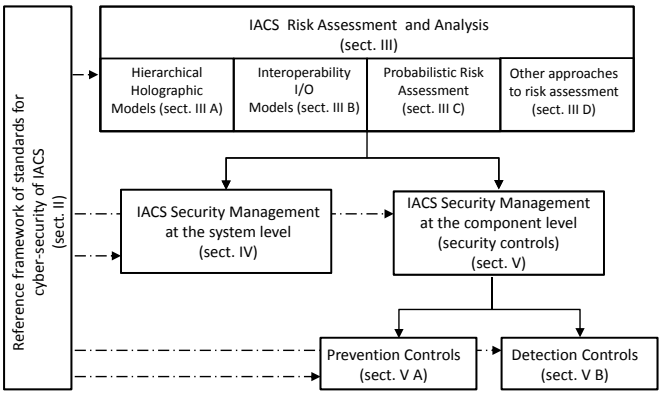


Fig. 2. Basic building blocks for IACS security

Let us consider, for instance, the three well-known basic security requirements [3], [6], that is:

- *availability* - the ability of being accessible and usable upon demand by other entities,
- *integrity* - the ability of safeguarding the accuracy and completeness of assets, and
- *confidentiality* - the guarantee that information is not made available or disclosed to unauthorized individuals, entities, or processes.

Clearly, they are important for both DBCS and IACS, but priorities are not the same as shown in Table II, so that also different security strategies are needed. Moreover, when main criticalities are considered, the situation is that summarized in Table III.

- The popular practice of *h/w and s/w patching and upgrading* (a routine activity in DBCS that helps in coping with new vulnerabilities as they are discovered) usually requires that (part of) the controlled system/plant be set temporarily offline. Unfortunately, this action has often to be planned far earlier (even weeks or months) than the scheduled time. In other words, the *availability* requirement could hardly be compatible with the patching/upgrading activities.
- *Real-time constraints* of most IACS can make asynchronous and/or sporadic actions, such as anti-virus upgrades, difficult or even impossible to be carried out. Also, the adoption of firewalls and complex filters commonly used in DBCS, can introduce unpredictable or unacceptable delays in control and process networks.
- *Consequences of failures* in DBCS are usually restricted to financial and/or reputation losses, because only *data* (information) need protection. This aspect is surely important for IACS too but, in this case, failures can also cause serious, maybe catastrophic, damages to the environment together with injuries to human beings and losses of lives (see Table I). This is mainly due to the strict interaction between IACS and the physical world, so that a strong connection of security to *safety* is also established in this case.
- *Performance* and *power* are critical in IACS, where many field and control devices have reduced computing capabilities and/or limited energy availability (e.g. battery powered devices in sensor networks). This makes unfeasible, for instance, the use of sophisticated encryption mechanisms and security protocols, that are usually eager for computing resources.

Because of the complex scenario introduced above, the problem of securing IACS cannot be solved by simply adopting the same techniques developed for DBCS, although the basic ingredients building up the whole process can be the same as the block diagram in Fig. 2 shows. In most cases, in fact, different scientific and technological solutions are needed.

Roughly speaking, current researches dealing explicitly with the security of IACS can be classified in two main categories. The first one takes into account the system as a whole, and deals with its characteristics from a global point of view. These studies include, for instance, some innovative approaches to the design and development of a secure system, the design of security analysis techniques and tools and the assessment, evaluation and management of risks at the *system level*. The second broad category includes those scientific activities carried out to tackle specific security problems at the *component level*. For our purposes, the term component refers to any (collection of) h/w and/or s/w mechanism(s) that can be used to improve the security of (a part of) the system. Typical examples of components are security protocols, authentication schemes and algorithms, firewalls, intrusion detection systems and so on. Obviously, system-level strategies often rely on or make use of mechanisms and solutions designed and implemented at the component level.

The remaining part of this paper is then organized as follows: Section II briefly discusses some relevant standardization efforts that are needed to set up the reference framework and enable the development of secure IACS and their interconnections. Sections III and IV deal with the security of IACS at the system level. In particular, Section III focuses on those aspects concerning the assessment and management of security risks, while Section IV presents some promising approaches for the system design and security analysis. Section V deals with security aspects at the component level, by introducing some countermeasures that can be adopted to prevent and detect security attacks to IACS. Finally, some conclusions are drawn in Section VI. The reader may also refer to Fig. 2 for a compact view of how the different elements fit together and to keep track of where the related discussions can be found in

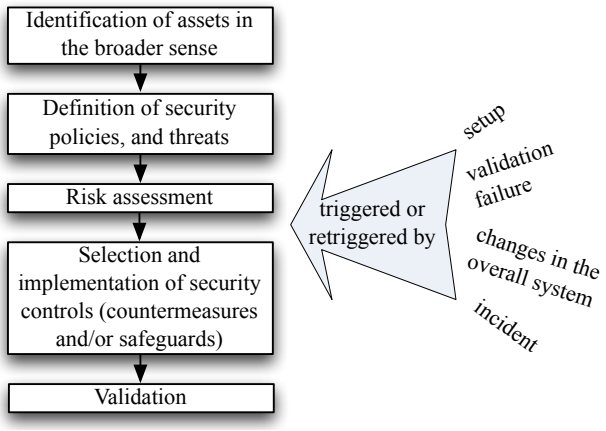


Fig. 3. Main phases of the ISM process

this paper.

II. IACS Security Standard Framework

From a historical perspective, security requirements of IACS were traditionally specified by organizations that were active in a number of critical infrastructure domains including, for example:

- water and gas distribution,
- electricity transmission and distribution [7], [8],
- gas and oil production [9], [10],
- food production and distribution [11],
- transportation systems.

In all these areas the importance of security has always been recognized as progressively increasing since ever. The heterogeneity in standardization approaches, however, enabled the development of a number of *ad hoc* security guidelines and recommendations [3], tailored to the specific needs of the application contexts which they were conceived for.

Subsequently, common requirements were identified, merged and amalgamated by international and national standardization bodies in several different documents. Today, the information security management (ISM) is almost universally regarded as a *process* also called *ISM System* (ISMS) by ISO/IEC [6], *Cyber Security Management System* (CSMS) by ANSI/ISA [1] and *Information Security Program* (ISP) by NIST [12]. The key-point, however, is that ISM concerns the whole organization of a company including, for instance [6]:

- training and commitment of employees and managers,
- relationships with partners, suppliers and customers,
- business continuity,
- legal and contractual requirements,
- compliance with security policies and standards,
- technical compliance,
- asset management,
- access control,
- communications and operations management,
- physical and environmental security

and more.

Note that all aspects listed above are strictly related: for instance, the commitment of management ensures necessary resources and investments (training, equipment, audits), while the training of employees enables the understanding of security mechanisms and techniques, as long as the correct implementation of policies and procedures.

TABLE IV
Contents of main standard documents

	ISO/IEC	NIST	ANSI/ISA
Terminology / Overview	[15]		[1]
Normative	[6]	[14]	[13]
General guidelines	[16]	[12]	
Specific guidelines	[17] [18] [20] [22] [25]	[2] [19] [21] [23]	[24]

Main documents focusing on ISM such as [1], [6], [12] adopt similar conceptual models that, in practice, are based on the logical and temporal steps depicted in the left part of Fig. 3. Events appearing in the right part of the picture, instead, are responsible for (re)starting and iterating the ISM process until a satisfactory situation is reached and confirmed by the validation phase. Validation is aimed at proving that the overall risk has been lowered below an acceptable threshold and usually involves both offline (i.e. new risk assessment sessions) and run-time (i.e. monitoring and measurements) activities.

The whole sequence of steps is then repeated whenever 1) the results checked in the validation phase do not match expectations (inadequate risk reduction), or 2) changes are introduced in any part/component of the overall system, including equipment, policies, risk levels, business, regulatory or legal requirements, newly discovered threats or vulnerabilities and so on, or 3) the run-time monitoring activities detect a security incident with consequences exceeding the acceptable severity threshold (estimated consequences are part of the results produced during the risk assessment).

It is worth noting that the ANSI/ISA approach (also known as ISA99) specifically deals with the security of industrial automation and control systems, although some guidelines are under development and not yet available at present. The normative document [13], however, also contains suitable informative elements for developing an ISM system.

Both ISO/IEC [6] and NIST [14] are mainly oriented to generic ICT systems, thus they do not provide specific guidelines, and this independence from any specific application context makes the two documents a general reference, which is suitable for the widest class of IACS. On the other hand, however, people have to face security issues pragmatically, for well-defined systems and in specific scenarios, thus both ISO/IEC and NIST developed sets of guidelines suited to this purpose. ISO/IEC is not oriented to IACS in particular, whereas NIST addresses well-defined IACS issues in [2]. Table IV summarizes the situation and goals of the main documents published by the three standardization bodies mentioned above.

In the table, the normative refers to those ISM aspects that are considered mandatory ([6], [14], and [13]), while the terminology overview deals with introductory concepts ([15]

and [1]). As Table IV shows, ANSI/ISA offers both general and specific guidelines in [13], whereas some technological aspects are investigated in [24].

General guidelines, strictly related to normative parts can be found in [16] and [12] for ISO and NIST, respectively. Moreover, [17] and [18] are specializations of [16] for telecommunication and health systems respectively, whereas [2] concerns IACS in particular.

With respect to Fig. 3, risk management is addressed in [20] by ISO/IEC and in [19] and [21] by NIST. [22] deals with the measures and measurements needed to assess the implementation of the controls of [6], as [23] does for the NIST suite. Finally, [25] helps with the implementation of [6].

Providing standard guidelines tailored to any kind of systems is really impossible, so that companies and organizations, wishing to deal with specific security needs, have sometimes to make proper selections of only those elements suited to satisfy their requirements. A significant step in this direction was made by the the working group on information security for Electric Power Utilities (EPUs) of the Council on Large Electric Systems (CIGRÉ) [26], [27], whose members first selected topics and procedures pertaining to EPUs from the standard documents mentioned above, and then carried out their reorganization according to the commonly agreed structure of Fig. 3.

Independently of their different structure and degree of generality, all main standardization efforts agree, in practice, on the same basic concept, that is

> security *risks* have to be minimized by means of suitable *controls* addressing the *vulnerabilities* exploitable by possible *threats*, whose goal is to abuse and/or damage *assets*

where the words risk, control, vulnerability, threat and asset have the following, conventional meanings [16]:

- *Risk* is a combination of the probability of a (negative) event occurrence and its consequent loss of value for the protected system.
- (Security) *control* is a means of managing risk, including policies, procedures, guidelines, practices or organizational structures, which can be of administrative, technical, management or legal nature.
- *Vulnerability* is a weakness of an asset or group of assets that can be exploited by one or more threats.
- *Threat* is a potential cause of an unwanted incident, which may result in harms to a system or organization.
- *Asset* is whatever has a value to the organization.

We will use these terms with the meanings listed above in the remaining sections of this paper. Moreover, security controls/countermeasures can be usefully classified with respect to the way they address the exploitation of vulnerabilities, so that the following hierarchical lines of defense can be defined:

- *Prevention* is the *first line of defense*. Its main goal is to avoid the exploitation of vulnerabilities. For instance, encryption techniques, designed to guarantee confidentiality even in presence of eavesdroppers, belong to this class.
- *Detection* is the *second line of defense*. It is not able to prevent the exploitation of vulnerabilities, but can detect it and trigger alarms consequently. For instance, Intrusion Detection Systems (IDSs) belong to this class.
- *Reaction/Recovery* is the *third (deepest) line of defense*. It is able to trigger the system reaction, when a successful exploitation of vulnerabilities occurs, in order to both minimize its cascading negative effects and guarantee the vital functions and activities of the protected system.

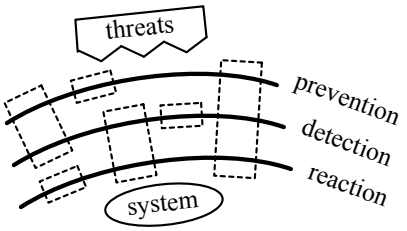


Fig. 4. Defense lines and security controls

Many research papers propose solutions to achieve or improve the security of IACS, that usually are based on controls belonging to multiple categories. Typical examples are depicted in Fig. 4, where dashed rectangles crossing two or more defense lines refer to controls conceived to cope with several defense actions at the same time.

III. IACS Risk Assessment

The identification, analysis and evaluation of assets and related risks are at the basis of any information security management system. In large and complex ISM systems, however, they are of utmost importance, since priorities have to be clearly defined for the proper allocation of resources where they are mostly needed. This part of ISM is perhaps better known in the literature as *risk assessment*. Risk assessment concerns security at the system level and can be thought of as a global modeling framework that includes more specific activities such as, for instance, the analysis of vulnerabilities and the evaluation and measurement of possible damages. In this section, however, we will consider risk assessment as a whole.

The constant growth of interest in the security of IACS and the consequent involvement of standard bodies and agencies in the definition of ISM, have led some of them to address the problem of risk assessment in specific application domains such as, for instance, Industrial Control Systems (ICSs) and SCADAs [19]–[21], where both safety and security aspects are sometimes dealt with jointly.

Risk consequences are often measured in terms of monetary losses, since this metric is widely understood and popular at the management level, although it could appear somewhat improper when referred to injuries or environmental damages. It is worth noting that other types of indexes too have been proposed, to this purpose, in the literature [28]–[31], but the definition and adoption of metrics that might be suitable and effective for real IACS surely needs further studies.

Many methodologies are now well-established, that are able to cope with the problem of risk assessment in general. In the case of IACS, however, their peculiarities must be constantly taken into account, starting with the identification

of assets and threats. These tasks can be difficult and hard to perform in large systems (such as large critical infrastructures), where all elements and their dependencies must be considered together with all possible interactions (inter-dependencies) between different infrastructures [32]–[38]. In fact, the overall complexity can significantly affect the number and type of alternatives that can be adopted.

Risk assessment techniques, which have been explicitly developed for IACS so far, can be classified in three main categories [39], depending on the way the model of the system is developed, that is: *Hierarchical Holographic Models* (HHMs) [40], *Inoperability Input-Output Models* (IIMs) [41], and *Probabilistic Risk Assessment* (PRA) [42], [43]. In addition, PRA can be split into methodologies based on either *deductive* (backward) or *inductive* (forward) analysis.

In Table V some significant papers, which have appeared in the literature and represent the current state of the art in this area, are grouped with respect to the proposed methodology.

TABLE V
MAIN METHODOLOGIES FOR RISK ASSESSMENT

HHM [40]	IIM [41]	PRA [42], [43]	
[44], [45]	[46], [47]	deductive analysis	inductive analysis
		[48]–[50]	[51]–[54]

A. Hierarchical Holographic Models

HHM [40] is a methodology conceived to decompose a complex system with inter-dependencies into several independent views (subsystems), each one focusing on different aspects and needs (e.g. the description of the short/long term behavior of the system with not commensurable time scale, its representation with diverse levels of abstraction that are useful to different people such as technicians and managers, and so on). After views have been specified, HHM allows to combine all "specific" models in a coherent way and to capture all possible sources of risk.

In order to rank, filter and manage the identified risks, [44] enhanced the work in [40] by introducing a *Risk Filtering, Ranking and Management* (RFRM) technique, that is mainly intended to both refine/prioritize the most meaningful risks, and prune those which can be considered as negligible, through a step-based approach.

In another relevant paper [45], instead, risk is ranked by means of a *Mean Time-to-Compromise* (MTTC) metric, so that evaluations and comparisons are made possible. MTTC takes into account the probability of a cyber-attack and the time needed to perform the attack itself. The implementation of security controls affects the MTTC value and the costs/benefits ratio can be evaluated accordingly. Authors conjecture that the same technique, although very preliminary to some extent, could likely be extended to model physical attacks and other kinds of vulnerabilities. In this case the best candidate to combine the different models would probably be HHM.

B. Inoperability Input-Output Models

IMM [41] overcomes some limitations of the HHM approach for systems with complex inter-dependencies among their components. In IMM the system is hierarchically decomposed into a number of subsystems which interact exchanging resources. The input of the risk analyzer is the initial perturbation triggered by an attack, while produced results are the possible cascading inoperabilities and economic losses.

The analysis of simple costs is a general limit of most techniques available today. Some studies have started to circumvent this problem with the introduction of operational data [46] to estimate the consequences of inoperability in highly interdependent infrastructures. As estimations are unavoidably provided by sector-specific experts, a methodology has also been proposed in [46], which is based on fuzzy-numbers, to deal with the problem of subjectivity.

The need of a precise estimation of the model parameters, that is usually carried out through a huge amount of real data (e.g. fine-grained agent-based models), is another critical (and still unsolved) problem. An interesting approach to tackle this issue has been presented in [47], where the abstraction capability of IIM and the fine-grained agent-based models have been merged into an *Agent-Based Inoperability Input-Output Model*.

C. Probabilistic Risk Assessment

The broad notion of *Probabilistic Risk Assessment* [42], [43] includes a number of methodologies and tools based on a shared characterization of the concept of *risk*, that is the severity (magnitude) of the consequences of an event and the likelihood that the event itself can occur [39]. Usually the underlying models of the system belong to the wide category of graphs (sometimes reduced to trees when dealing with simpler systems and/or inter-dependencies, or when a coarser grained analysis can be considered satisfactory). In most cases graph *vertexes* represent the system components while *edges* describe dependencies. On the other hand, the ways graphs are analyzed fall in two sub-categories of *PRA*, that is either *deductive* (backward) or *inductive* (forward) analysis techniques.

1) Deductive analysis: Deductive analyzers define a so-called *top* event representing the unwanted consequences of attacks or failures. Starting from the affected system components, the model is then explored until the origins of the attack or failure are found. Typical examples of deductive analysis are the *Fault Tree Analysis* (FTA) [48], dealing with faults, and the *Attack Tree Analysis* [49], [50], where the top event is the attacker goal rather than a fault.

2) Inductive analysis: Inductive analyzers start from a triggering event and compute all its possible consequences. The work presented in [51] is a case of inductive analysis where *Binary Decision Diagrams* (BDD) are adopted to improve the performance of the analysis.

Other significant examples are the *Failure Mode and Effects Analysis* (FMEA) and *Failure Mode, Effects and Criticality Analysis* (FMECA) approaches [52]. Both of them deal with failures, i.e. only safety and reliability are considered, whereas the *Failure (Intrusion) Mode, Effects and Criticality Analysis* (F(I)MECA), proposed more recently [53], also takes into account security aspects. Note that F(I)MECA is mainly

suitable to deal with single-point failures/intrusions [48]. From this point of view, more realistic attack models have still to be studied and developed. Most modeling techniques, in fact, are explicitly or implicitly based on the assumption that only a low percentage of nodes in the system can be compromised, but this has recently proven to be wrong. More investigations are also needed, in general, for probabilistic approaches to security: in very large and complex situations the exhaustive computation of all possible attacks is often impossible or simply not practical. For many applications, the detection of attacks with a very high probability (e.g. 85 %–95 %) could be enough, and this could make the modeling and analysis tasks significantly simpler.

To deal with very complex systems, [54] proposed a hierarchical model based on hyper-graphs, where each vertex, in its turn, can be further replaced by another hyper-graph. The analysis expands a vertex into the corresponding hyper-graph only if this is really needed (some necessary conditions exist to decide when the recursive expansion can be stopped). This approach tends to keep a fine-grained analysis as simple as possible.

D. Other Approaches

Some probability-based approaches, which do not fall in our rough taxonomy, can be found in [28]–[31]. In particular, two methodologies, among those surveyed in [28], are very interesting:

1) since probability (likelihood) is very subjective in many cases, and consequently difficult to estimate and error prone, it can be replaced with the concept of *uncertainty*. Examples of this can be found in [29] where the definition of risk is changed to a combination of possible consequences and related *uncertainties*.
2) Probability can be completely neglected, and only the value of the system components (as perceived by stakeholders) is taken into account to allocate resources for risk mitigation [30], [31]. The main drawback in this case is that, in practical situations, resources are limited and insufficient to protect all assets exhaustively, so disregarding probabilities of accidents may lead to implement sub-optimal solutions [28].

In some SCADAs, a number of heterogeneous networks are used to connect parts of a large and complex system. Typical examples are power grids, where proprietary and public telecommunication networks cooperate to grant the expected functionalities and performance. Risk analysis in such a kind of scenarios can be profitably performed by employing a set of heterogeneous tools and methodologies as shown in [55], where probabilistic dependability analysis, worst case analysis and real-time performance evaluation were suitably combined to deal with security and performance aspects of a real energy distribution system.

It is worth noting that the quantitative assessment of risk obtained with the methodologies in [55] could also be useful in evaluating the effectiveness of policies and configurations proposed for risk mitigation.

IV. IACS System-Level Security

From a systemic point of view a very big challenge, demanding for deep technical innovations, is the development of a new kind of IACS which are security-aware. Until recently, in fact, security issues have not been considered too seriously in the early planning phases of a new system. The main reason is that security is often perceived as a sort of (even important) add-on, that may be included in the system at a later time or, howsoever, whenever it is needed [56]. This way of thinking has influenced the research community for quite a long time, and is still affecting many scientific and technical works also today. Most papers appeared in the literature, indeed, present techniques and solutions to either introduce/improve security mechanisms in some existing system or superimpose security after a system has been conceived and developed to satisfy its functional, application and performance requirements. From a certain point of view, this approach might also be considered reasonable, at least up to a point, because

- re-designing (parts of) existing IACS, is simply unfeasible or exaggeratedly expensive in most cases, and
- many IACS are designed and deployed to work continuously, with very little or no changes, for years or tens of years and cannot be easily halted and replaced (see Table I).

Recently, however, some researchers have start proposing different, promising approaches to conceive and design a new generation of IACS, where security requirements are taken into account at a very early stage in the specification of the system. This will have a deep influence on both the traditional design philosophy and techniques that have to be radically changed and, of course, demand for further, large investigations.

A remarkable example is represented by the innovative way of designing security-aware linear control systems introduced in [57], where the design of countermeasures includes some knowledge of the physical system under control. The change of perspective, in this case, is that the authors started with studying the consequences of attacks on the estimation and control algorithms, in order to develop attack models for the controlled system. This enables the designers to focus on the final goal of the possible attacks, rather than on the exploitation of known vulnerabilities. A more detailed analysis and refinement of this technique was subsequently presented in [58]. As far as we know, this was the first time that specifications of a control system included a security goal, which was explicitly provided to protect the conventional operational goals even in presence of denial of service (DoS) and deception attacks (deception attacks are a way to compromise the system integrity by replacing, with false information, sensor data sent over the network). Attacks were then modeled by suitable modifications of the control equations. The significant advantage with respect to more conventional detection techniques, based for example on intrusion detection systems, is that a model is created of the physical system, instead of the network and software components. In this way, a security threat can be quickly detected as an abnormal response of the physical system to the control commands, independently of the characteristics of h/w and s/w elements included in the ICT infrastructure.

A recent paper [59] has shed additional light on a number of challenging topics, already introduced in [58], that still demand adequate scientific solutions. By means of more complex and sophisticate models, in fact, the authors have considered the behavior of adaptive adversaries that are aware of the detection countermeasures and attempt to evade them according to three different strategies, which are particularly important in the case of IACS, that is

- maximize the damage as soon as the system is penetrated, or
- slowly and silently modify the behavior of the system and maximize the damage once it has reached a highly vulnerable state, or
- introduce small perturbations in the system for a very long time.

Note that the second and third types of attacks can be very subtle and hard to detect, representing a real nightmare for IACS, but their consequences can be catastrophic nevertheless.

Models and experiments presented in [59] show the feasibility of the proposed solution, at least for the kind of control systems considered, although a lot of work has still to be done in this direction.

Another topical issue tackled in [59] concerns the ability of the system *to survive* attacks. Most studies about IACS security focus on prevention and/or detection techniques, but relatively little research has been done about the response to threats. In this case, instead, the authors suggest to adopt the estimate of the state of the physical system, produced by their secure controller, as the natural (automatic) response strategy when an anomaly is detected. The automatic response should be used only temporarily, while waiting for some human intervention that can often occur after a significant amount of time (i.e. several hours). Fast cicatrization of damages produced by attacks is a current, open research issue, together with the management of false alarms (false positives) and their potential side-effects.

A second big challenge, where a radical change of direction is needed, is in the way IACS security problems are tackled and solved today. In fact, most techniques and solutions developed so far, have been based on a "static" view of security, but systems, components, threats and attacks change continuously and new challenges have always to be faced [56]. This demands for new methodologies and information security support to evaluate and assess the security level of IACS, to check their vulnerability to new and different types of attacks and to suggest the adoption of suitable countermeasures, which can be developed only after a significant turn of mentality in the approach.

Fortunately enough, although IACS can be very complex systems, they usually have a reduced *network dynamics* when compared to DBCS, since the set of users and protocols involved is smaller and almost fixed, while system topologies are simpler [57]. In perspective, this factor can be leveraged to simplify the development of models and analysis techniques and the introduction of countermeasures.

Security analysis and management at the system level are manifold processes that, given the size and complexity of most IACS, cannot be carried out by hand. Awareness of this is continuously rising, so that preliminary studies have started to appear for computer-aided techniques, that are able to model the security characteristics of the system from a global point of view, and to carry out analysis and evaluations at different levels of abstraction.

Vulnerability analysis, besides being a basic element of risk assessment, is perhaps the most important (and to several extents critical) step in assessing the security of a system, so that several authors have paid particular attention to this topic. The work presented in [60] is an attempt to deal with IACS vulnerabilities at the system level. In the proposed approach, different kinds of information are collected from different sources concerning, in particular, known vulnerabilities to security attacks and their consequences, faults that can affect components in the system and descriptions of the system topology, connections, protocols, installed s/w modules, running processes and so on. Information is then used to develop, in a fully automatic way, a complex and very detailed security model of the system itself, and the model is then processed by an analysis engine written in Prolog. The engine is able to predict if (even complex) security attacks, built by exploiting sequences of known vulnerabilities, may be successfully carried out on the system. The usefulness of this solution is twofold: in fact, when securing a system, the results produced by the computer-aided tool are of help in both detecting weaknesses and introducing suitable countermeasures, whose efficacy can be checked (at least from the functional point of view) before their deployment in the physical system. Moreover, when even a small change is introduced in the system h/w and/or s/w, the new resulting configuration can be automatically analyzed to verify if the security requirements are still satisfied. The same occurs when, for instance, a new vulnerability is discovered that can potentially affect some system component. In practice, the ability to carry out some kind of "what if" analysis on the model offers the valuable advantage of quickly studying new security scenarios without the need of any intervention on the real system.

To be fair, however, the reader must be warned that several issues are still open and need further investigations, before the approach proposed in [60] could be considered mature enough for widespread adoption in the analysis of real systems. In particular, the (digital) description of the system is a critical point. In our experience, information needed to developed the model in sufficient details is often neither available from the users, nor it can be realistically managed by hand. This is why, in [60] the use of publicly-available databases has been considered for known vulnerabilities, which unfortunately have been designed for reference by humans rather than machines. Vulnerability databases particularly tailored to IACS are not so popular yet, moreover this is only one "view" of the system, since similar information is needed for faults, topologies, components and so on. The neat result is that the automatic analyzer has to make assumptions when some (piece of) information is unavailable. The most conservative choice in the analysis is considering the worst possible case (largest damage) caused by the missing/unknown data, but this leads to take into account also a number of false positives (i.e. attacks that cannot occur in the real system but that nevertheless

cannot be ignored because of the incompleteness of the model) that have to be dealt with in a post-analysis step.

Another problem concerns the classification and management of attacks sequences detected by the analyzer. Even for medium-sized systems, in fact, the number and complexity of possible attack patterns can be very large. Additional studies are then needed about techniques for "grouping" similar attacks (e.g. threats that can produce the same final effects) and for (semi)automatically ranking attacks according to their dangerousness, for instance by considering the damage they can cause to the system. This aspect becomes important when priorities of interventions have to be established in deploying countermeasures in real IACS.

At a higher level of abstraction, users and security experts deal with policies rather than countermeasures. Mapping policies onto actual mechanisms (i.e. profiles, accounts, privileges, and firewall rules) of real devices is often an error-prone activity that can benefit from the availability of suitable information technology-based support [61]. This motivation has led to extend the techniques presented in [60] with a following paper [62], where the model description is enriched with information concerning user roles (e.g. administration, technical, maintenance staff) and permissions. The automatic analysis, in this case, is able to verify whether the high-level policies (i.e. no clerk can read data directly from the shop-floor) really match the expected security requirements and no internal inconsistency in the policy set (conflicting policies) exists. On the other hand, the proposed techniques also enable to check whether the actual implementation of policies in a running system still satisfies the original security requirements, for instance when changes or updates are introduced. A case study [63], based on real configurations coming from an industrial plant, shows the feasibility of the proposed technique.

Although automated tools in this area are mostly experimental prototypes, their application to real IACS is forthcoming, as confirmed by the experiences described in [64]. In this case, in fact, the authors discuss how a special-purpose automatic analyzer for policies can be profitably used to simplify the configuration of firewalls in real, large-sized industrial control networks. However, the most interesting aspect of this work is perhaps that the authors show how real-world systems have complexities which are not considered as scientifically relevant, so they are often overlooked in academic research. Tackling those complexities is, instead, mandatory if the proposed solutions have to be successfully introduced in industrial systems.

V. IACS Security Controls

While security strategies and policies are mainly dealt with at the system level, mechanisms to enforce and support them are usually of interest of the component level. As already mentioned before, in this paper we use the term component with a meaning broad enough to include a number of security-related controls and techniques such as, for instance, cryptography and cryptographic protocols, which are adopted for ensuring privacy and authentication in the communication. This section, in particular, focuses on those controls concerning (intrusion) prevention, detection and reaction to security attacks. Although these three aspects are conceptually distinct, they are rarely considered separately, as in many practical situations countermeasures are conceived to tackle two of them (typically detection and reaction), or even them all, at the same time (see Fig. 4). Thus, the reader should be warned that the rough separation adopted for the following two sub-sections is mainly for paper organization reasons, and significant overlapping between them can instead be found in the literature.

TABLE VI
Main features of IACS prevention controls

	explicit security policies or requirements	explicit threat or attacker model	security analysis and validation	performance impact evaluation	prototype implementation
[67]	N	N	N	N	P
[68]	Y	N	N	N	P
[69]	P	N	N	N	Y
[70], [71]	Y	P	P	Y	N
[72]	Y	Y	P	N	N
[73]	Y	Y	P	Y	Y
[74]	Y	N/A	N/A	Y	P
[75]	Y	P	Y	P	N
[76]	Y	Y	Y	N/A	N/A
[77]	Y	Y	P	P	N/A
[78]	Y	Y	N	Y	N
[79]	Y	P	P	P	N
[80]	Y	P	P	N	N
[81]	Y	Y	Y	P	Y
[82]	Y	Y	Y	P	Y
[83]	Y	Y	P	P	P
[57], [59]	Y	Y	P	Y	N
$\frac{c(Y)+0.5\cdot c(P)}{c(Y)+c(P)+c(N)}$	92 %	69 %	50 %	56 %	35 %
Y: yes N: no P: partially N/A: not applicable					

Another important aspect to be taken into account is that, in the past, most attacks to a company assets originated from inside the company itself and were carried out by rogue employees or ex-employees [65], while in the last years the situation has become opposite and most security threats now come from the outside [66]. While security controls can be of some help to limit the effect of external menaces, they can do little or nothing against hostile insiders. Moreover, IACS can even be more exposed to this kind of threats than DBCS because of their strict connections with the controlled physical systems (see, for instance. the difficulties in providing physical surveillance of remote sites or locations distributed over wide areas mentioned in Table I). Planning and implementing suitable security policies, instead, can be useful in this respect, since they can reduce, to some extent, the ability of (rogue) employees to perform attacks by granting them only those rights that are strictly necessary to carry out their job.

A. Prevention Controls

In principle, contributions to IACS intrusion prevention should follow a well-established sequence of four steps consisting of

1) the definition of the security goals (i.e. explicit security policies or requirements),
2) the implicit/explicit development of one or more models of the attacker/threat that could violate the above policies,
3) some kind of security analysis and/or validation to prove that the proposed security controls are able to satisfy the requirements, even against the modeled attacker/threat, and
4) some performance evaluation to check that the proposed controls do not affect the system behavior negatively (e.g. with respect to the real-time and/or power constraints listed in Section I).

The situation for some relevant solution published in the literature is summarized in Table VI, where the presence/absence of the four elements mentioned above is approximately measured by means of an attribute, whose computation is shown in the bottom row of the table. In the last column, moreover, the availability of prototype implementations for the proposed techniques has been considered. The last but one row is the percentage of considered papers that satisfy the requirement in each column (papers marked N/A were not included in the computation). In the equation of Table VI, $c(X)$ is the number of cells in a column that contain the "X" value. Moreover, in order to make the computation consistent, the value "P" of the attribute has been weighted 0.5. The computed index is a very rough indication of how much each basic step has been tackled so far in relevant academic contributions. It is easy to see that the three rightmost aspects are neglected in almost half of the cases or even more: this should motivate the need of further research and technical efforts in the following areas:

1) a-posteriori verification methods able to prove that the controls, selected to implement a given security policy or requirement, really match the expected behavior against the feared threat or attacker,
2) some kind of performance impact analysis (hopefully) showing that the various overheads introduced by the proposed controls do not ditch the system performance,
3) the implementation and experimentation of the proposed techniques, enabling everybody to carry out her/his own evaluation and comparisons, and even further developments and customization. This aspect, in fact, is often underestimated and many interesting proposals remain in the state of hypothetical solutions, because of the impossibility to test and validate them in real, or at least realistic, application scenarios.

The secure, safe and predictable connection of automation networks and devices via public (i.e. insecure) networks was addressed in the Virtual Automation Networks (VAN) project [67]. The proposed solution for the prevention of cyber-attacks in VAN is vendor-independent and makes use respectively of HTTPS protocol for web applications, Virtual Private Network (VPN) tunnels for point-to-point communications, access control for automation functions of VAN devices, and packet filtering managed through web services. A Public Key Infrastructure (PKI) enables trusted relationships between devices. Unfortunately, [67] presents informal concepts, but it neither provides implementation results and performance analysis, nor it shows how the security challenges listed in Section I were addressed in VAN.

The security requirements in the highly demanded collaborative control of distributed device networks under open and dynamic environments were addressed in [68], by inserting a Security Agent (SA) layer between each entity and the insecure network environment. Through a PKI, SA should be able to guarantee all the desired security properties, though no formal proof is provided that performance and functional requirements are really satisfied.

Recommendations on how securing existing SCADA networks can be found in [69]. The authors' proposal takes into account both the security controls which, in general, could be implemented in traditional SCADAs, and a forensic system to log all communications and enable *post mortem* analysis of security breaches. Some implementation details about the forensic system can also be found in [84].

The focus of [70] and [71] is, instead, on some critical issues concerning key management and broadcast/multicast of confidential messages in SCADA networks. In particular, [70] shows how secure broadcast and multicast communications are enabled by means of pre-shared symmetric keys used to originate fresh session keys. The computational power of the involved nodes is carefully considered, and the highest effort is imposed on most powerful nodes, while simplest devices are not requested to carry out demanding computations. The efficiency of multicast communications is then improved in [71]. Both [70] and [71] deal with some qualitative security analysis, aimed at proving that the proposed solution prevents a number of possible attacks, but the correctness of the adopted cryptographic protocols is not assessed through formal verification [85], [86].

Similarly, [72] deals with the same problems discussed in [70] and [71], but proposes a different management scheme, although still needing pre-shared symmetric keys. Some more effort is devoted, in this case, to give formal evidence of the correctness of the proposed protocols.

As summarized in Table VI, all papers mentioned above provide satisfactory analysis for threats and SCADA security needs, however they also exhibit a common limitation to their key management schemes. In fact, they are based on the assumption that public and private-key systems could not be implemented in SCADAs due to resource constraints, even if they would mitigate the disadvantage of pre-loading a possibly huge number of keys. Fortunately, at present this does not seem to be completely true any longer, thanks to the advances offered by promising approaches adopting the Elliptic Curve Cryptography (ECC) [3], [76], [77], [81], [87]. Moreover, some recent advances [88] have also improved the computational performance of the classical Rivest-Shamir-Adleman (RSA) public-key cryptosystem.

Key management schemes are necessary in IACS, since most industrial devices cannot rely on trusted human operators (who enter, for instance, personal identification numbers or passwords) for authentication in the distributed environment. PKI is very attractive in this context, but a number of practical issues, affecting the interoperability among products of

different vendors, have to be satisfactorily solved yet. In fact, even though the set of different vendors is somewhat kept small in many IACS, nevertheless the need for redundancy and reliability often pushes for the adoption of multiple vendors. Standardization, in this case, can be of help in dealing with interoperability also from this security point of view, by identifying basic and optional functionalities, offering guidelines for device design and implementation and specifying requirements and procedures for conformance testing and validation.

Power saving is an important issue when mobile devices, such as operator hand-held terminals, are involved and this affects security management too. For instance, [73] presented a solution (including a mathematical model and prototype implementation) for energy-aware robust authentication based on secret keys and identification numbers shared between mobile devices and a server. The main achievement of the proposed protocol is its intrusion resilience (implicit reaction), since a successful attack to a device cannot compromise the server, and vice-versa. Moreover, the server can detect a compromised device. The correctness of the protocol was justified in an informal way, while good threat and attacker models were developed.

Power consumption is also considered and evaluated through an analytic model in [78], where techniques of symmetrical forwarding are used in wireless communications to detect *black hole* attacks (i.e. malicious behaviors which stop the forwarding of packets towards their intended destination).

Limitations in standard authentication mechanisms, when applied to industrial control networks, were discussed in [75]. In particular, the authors highlight risks caused by ignoring the security status of each device accessing the network. Indeed a device, which might be affected by severe vulnerabilities, should never be allowed to participate in the communication. A Trusted Network (TN) architecture for industrial control systems can provide adequate support through several security services: authentication, comprehensive network device admission control, end-device health check, policy-based access control and traffic filtering, automated remediation of non-compliant devices and auditing.

Security mechanisms constrained by real-time requirements have been studied and validated in [74], where the computational overhead needed to provide security (with a certain quality of service - QoS) has been successfully introduced in the traditional real-time scheduling algorithm Earliest Deadline First (EDF).

Security issues for particular kinds of networks and application scenarios have also been addressed in the recent past, such as those concerning, for instance, Wireless Sensor Networks (WSNs), Building Automation Systems (BASs) or Supply Chain Management (SCM) and Process Control (PC) networks.

In particular, the security of WSNs has been extensively studied in [76] where the reader is led through the canonical securing steps ([1], [6], [12]) which start with the exact definition of the security properties and the development of the threat models (e.g. *outsider attacks*, *insider attacks* and *key-compromise attacks*). Since securing everything and everywhere is usually not feasible for cost reasons, authors carry out the risk assessment phase by both developing a suitable security metric and analyzing the available controls to properly secure the network. They also confirm that the use of Elliptic Curve Cryptography [87] can make the key-management protocols simpler and lighter, so as to overcome the computational drawback that makes public-key cryptography schemes unsuitable for many IACS.

A detailed analysis, similar to [76] can be found in [77] for those WSN technologies already including some form of security controls (i.e. ZigBee PRO, WirelessHART and ISA100.11.a). Some weaknesses of the protocols are discussed in the light of the security properties and threat models, and countermeasures are consequently proposed and analyzed, with particular emphasis on the need of adopting formal methods to audit security procedures and policies in such a kind of IACS networks.

Authentication with anonymity in WSNs was discussed in [79], where a detailed security analysis for the proposal appeared in [89] is presented. To overcome some weaknesses of [89], authors then proposed an enhanced version of the protocol, but the security analysis of the amended protocol is only qualitative.

Both [80] and [81] focus on security in building automation systems. [80] mainly deals with the life-cycle model (from design to deployment) of systems integrating safety- and security-critical services, showing how to apply the proposed approach by means of a use case. By contrast, [81] follows the canonical steps to secure a BAS ([1], [6], [12]) and discusses the security controls that are currently available. In this case too, authors recommend the adoption of Elliptic Curve Cryptography, moreover the problem of broadcast and multicast confidential messages is also considered as in [71] and [72].

The security needs of systems for the automatic identification of goods and products, which are based on radio-frequency identification (RFID) tags, were studied in [82]. After the definition of the expected security properties and the development of the attacker/threat models, authors introduced a new architecture, based on RFID class 2 tags. In this solution a single static key (for each tag) is shared with the authentication server, so that there is no need for a centralized database to host all information (i.e. absence of a single point of failure) and the system availability is largely improved. The correctness of the confidentiality mechanisms was formally proven, whereas only qualitative justifications were given for other proposed mechanisms. The feasibility of the proposal, however, has been supported by both implementation and simulation.

Similar topics are dealt with in [83], but the solution proposed there is less sophisticated and lacks a formal correctness proof.

B. Detection Controls

Preventing any threat to assets is clearly not possible and this is true, in particular, for IACS, where the dynamics of changes in h/w and s/w during the system lifetime is by far slower than the evolution of attack methods and technologies

(see Table I). Keeping the system under continuous monitoring is then essential, both to rapidly notify the people in charge when dangerous situations occur, and to trigger (automatic) reactions for fault mitigation and healing. In fact, this is the primary goal of *intrusion detection* controls.

Intrusion detection in computer networks is a well-known and established issue, which dates back to the eighties [90] at least. Intrusion Detection Systems are designed to quickly discover the presence of attacks in progress or the occurrence of failures, by means of some *evidence* gathered from the live system, while it is performing its operations. Not only ideal IDSs should avoid that some attacks go undetected (false negatives), but they are also requested not to cause false positives, that is alarms raised when no attack is in progress. In the following, we will call *accuracy* this characteristic which is one of the main areas where continuous research and development are needed.

A basic classification of IDSs, mainly coming from experience with DBCS systems, is possible if two main aspects are taken into account [91], [92], that is:

- Source of information: *host-based* and *network-based* IDSs are the alternatives, which do not refer only to whether the elements (sensors) devoted to evidence gathering are either concentrated on a single host or distributed in several nodes of the network. Indeed, host-based IDSs are stand-alone, in that they work by checking the interactions with the local operating system, while network-based IDSs analyze the whole network either in a centralized or distributed fashion.
- Detection technique: according to this point of view, two approaches are possible: *signature-based* IDSs look for actions (e.g. traffic patterns, message contents, bandwidth consumption) generated by known attacks, while *anomaly detection-based* IDSs check for anomalies with respect to the *expected or normal* system behavior. The expected behavior is then obtained based on either automated training or suitable (manual) specifications.

The direct introduction of IDSs in IACS environments, which have been developed for conventional DBCS, is often unfeasible due to the constraints and limitations already discussed in the previous sections. For this reason, no host-based solution appears in Table VII, which shows a technique-based classification of most significant literature papers dealing with IACS IDSs.

TABLE VII
MAIN METHODOLOGIES FOR DETECTION

	network-based	
signature-based	[91]	
anomaly detection-based	stateless	stateful
	[93]–[96] [98]–[102]	[92], [97] [103]–[105]

Signature-based techniques require the explicit definition of "signatures" of known attacks in terms of characteristic message patterns. Unfortunately, two main drawbacks have to be carefully considered in this case: first the exact characterization of attacks (in terms of messages involved) is a difficult task which can significantly affect the effectiveness of detection. This means that the derivation of suitable signatures (that are not available from other application domains because of peculiar protocols and messages adopted in IACS), has to start almost from scratch.

Second, discovering new (unknown) attacks (the so called zero-day attacks) is impossible. It is worth noting that the latter limitation does not affect, at least in theory, the anomaly-based detection systems.

Drawbacks affecting the signature based-detection techniques explain, to some extent, why most studies have focused on the detection of anomalies, as the bottom row of Table VII shows. A remarkable exception, however, is presented in [91] where a set of DoS, password crack and confidentiality attacks were launched against a simulated IEC 61850 network (automated electric substations). In that case, in fact, the signature-based IDS was able to detected all the corresponding (known) attacks accurately, without imposing any significant penalty on the system performance, thanks to its connection to a mirrored port of a switch placed between the office and IEC 61850 networks. The authors also suggested, as a further prevention/detection policy, to increase the monitoring activity in those situations where either the energy demand exhibits peaks or the human alertness is low.

Some IACS peculiarities (i.e. the reduced set of users and protocols with respect to DBCS, the simpler system topology/configuration and the equations of the controlled process/plant which are known a priori) can be exploited to derive a satisfactory description of the *normal* system behavior and to conceive and develop a new generation of IDSs for the industrial arena. Appealing approaches range from solutions which rely on the analysis of network traffic only, to IDSs that take into account the status of devices to some extent, and even to countermeasures driven by the state of the whole system. We have classified these alternatives in two simple classes, that is *stateful* IDSs (not to be confused with stateful firewalls which keep track of the network connections and are frequently used in DBCS), that make use of information at the system level, and *stateless* IDSs that include all other approaches. In general, we can say that the precision of IDSs increases with the amount of system-level information they use, so that in some advanced cases not only detection is more effective but even some kind of prevention is made possible. Consequently, the attacker notion mastered by the IDS is progressively shifted from simple tracks (i.e. deviations from the expected traffic stream) to a clear view of the attacker's goals, depending on the features of the target system.

It is worth noting that the stateful approach appears to be viable only for IACS and could hardly be proposed for DBCS. In the former case, in fact, information concerning the system goal, its normal/anomalous behavior and safe/unsafe state is well-known a priori and usually concerns a restricted, enumerable set of possible situations. When DBCS are considered, instead, this is no longer true and keeping track of the overall system conditions is unfeasible in most cases.

Of course, powerful IDSs are more demanding in terms of computational power, communication bandwidth and sensors, so that a reasonable trade-off between accuracy, cost and performance penalty is often needed. This fundamental aspect

has to be carefully taken into account in reading the remaining part of this section.

1) Stateless IDSs: DoS attacks to a generic control system (sampling rate equal to 0.02 s, controller and plant interconnected through the Internet), were simulated in [93]. In particular, the characterization in terms of packet delays, jitters and losses and their correlation to the rise and settling times of the controlled system were used to measure how much the system performance could be affected by DoS. Authors then proposed to deploy IDSs on the network routers, and showed how the rise and settling times of the controlled system improved under the same attacks. Accuracy and performance impact of IDSs were not explicitly evaluated, but only considered through the enhanced rise and settling times.

DoS attacks are also dealt with in [94] for generic automation systems. In this case, the authors have addressed prevention by means of a computational-expensive two phases authentication protocol, detection by employing customized IDSs which analyze the network traffic only, and reaction by isolating the attack sources with virtual bridges and redundant paths. The impact on performance, unfortunately, has been discussed only with respect to prevention, whereas the evaluation of the IDS accuracy has been neglected.

[95] and [96] contain detailed works based on the analysis of network traffic. In both cases, suitable reference models are built during a *learning* phase and then used to make comparisons with real traffic extracted from the message streams. The learning activity in [96], however, only considers the expected traffic, while also intrusions are taken into account in [95]. Other main differences concern the way traffic streams are rearranged, what fields are considered meaningful and how differences are computed. To this purpose, [95] makes use of neural networks, whereas [96] adopt geometric representations and distances. [95] refers to a generic fluid flow control system as a test-bed for validation, and shows that perfect accuracy is reached with simulated attacks. No performance impact is provided for the proposed technique, although it is expected to be low as it mainly depends on computations carried out on gathered frames, while the passive sniffing contribution is negligible. On the other hand, [96] adopts a SCADA validation test-bed. Simulation of real attacks, in this case, has led to a detection rate of about 90% with 0.2% false positives. The performance impact has also been precisely evaluated in terms of throughput.

An initial step towards the exploitation of further system information can be found in [98], where an IDS has been proposed that makes use of the expected communication patterns among devices in Modbus TCP networks, and is also able to detect changes in server or service availability. In practice, the notion of "who is allowed to initiate communication with whom", established for each device, is paired with the model of the normal network traffic (frame formats, payload and so on). Deviations from the expected behavior are then detected by means of pattern matching techniques. Moreover, a service discovery mechanism allows to detect suspicious changes in either the server configuration or the service availability. Validation, carried out through a generic SCADA network, shows a good accuracy for the proposed solution. Finally, its impact on performance is only affected by the amount of computations carried out on the analyzed frames.
As a side comment it is worth noting that a "proof of concept" of an approach similar to [98] has subsequently appeared in [99].

The work described in [100] shares several points of contact with [96] however, in the former case, traffic and usage profiles are created by means of working statistics for devices (i.e. CPU time) too, by assuming that they are fully committed to process control activities. This enhanced knowledge of the monitored system also enables the detection of legitimate users' misbehaviors (changes of access levels, anomalous activities and so on). No evaluation of the performance impact has been provided for the proposed technique, which unavoidably requires the careful planning and management of any network monitoring activity, in particular with respect to the real-time communications requirements of the controlled system. Nevertheless, experiments on a simulated system show that satisfactory accuracy may likely be reached.

Legal commands for each device and expected device settings are used for comparisons in [101] during the system operation. Details about the comparison technique, such as those provided in [95], [96], [100], are unfortunately missing here. The up-time (availability) of devices is also monitored as already discussed for [98], but performance impact and accuracy evaluation have not been considered in this case.

While papers discussed above deal with wired devices and networks, [102] focuses on wireless industrial sensor networks (WISNs) and presents a hierarchical framework for intrusion detection and prevention in WISNs. Detailed results about the accuracy of the proposed method are then given, whereas performance issues are addressed only indirectly through references to other papers appeared in the literature.

2) Stateful IDSs: When information concerning the whole system is exploited, both attacks and faults can be detected and even predicted. This also enables IDSs to reason about the attacker's goals instead of the attack mechanisms, a characteristic which can be particularly useful when dealing with threats conceived to slowly shift the system behavior to an unsafe state. This kind of menaces, in fact, is likely to remain undetected when only the monitoring of traffic is used, while more powerful countermeasures might be deployed by taking into account the whole system operation.

The IDS proposed in [97] relies on an image of the system states where potentially unsafe situations (i.e. a tank filled beyond a predefined maximum limit with safety valves closed) are clearly identified. By monitoring the network traffic, the status of devices and their settings, the IDS computes some kind of *distance* between the current system state and the unsafe situations stored in its memory. If the distance falls below a given threshold, an alarm is raised.

Validation of the proposed technique has been carried out on a simplified boiling water reactor and, unsurprisingly, accuracy and performance impact have been found to be antithetic. Improvements to accuracy, in fact, require precise information on the current state of the real system by the IDS, and to achieve this, data must be frequently collected from the network. Unavoidably this leads to traffic peaks which affect

the real-time (control) communications.

In a previous paper [92] the same authors implemented their technique on a Modbus network and showed the feasibility of their method in different conditions. In that case, however, they also compared the obtained accuracy to more traditional signature-based IDSs, to show that attacks, which conventional IDSs are able to detect are, a proper subset of those captured with their approach.

Similar principles inspired the work presented in [103] where the IDS has the notions of both the unsafe and current system state, called *work-flow* there. The IDS captures the device commands before they are issued to the proper recipients, and simulates their effect on the work-flow: if the system might be driven too close to an unsafe state, commands are replaced with proper outputs to keep the running system away from the dangerous situation. By contrast, they are simply forwarded to the intended destination if no alarming condition is detected. A main advantage of this approach is the ability to manage both attacks and faults at the same time. From a global point of view this IDS is more powerful than those described in [97] and [92] but it is also more intrusive, thus its performance impact could be critical.

In the power systems scenario, [104] proposed an engine for the on-line analysis of system security. The engine ”knows” the system in terms of equations and, by acquiring data in real-time through a number of distributed sensors, it is able first to compute the current state and second to foresee all possible evolutions by means of simulation. When possible critical situations are discovered, checks are performed to verify whether the system can tolerate them and alarms are raised when needed. Authors claim that the reaction time of such a system can be estimated in minutes, nevertheless, “parallel processing seems to be the only viable solution to speed up the simulations and obtain results in useful time”. This confirms that the performance impact of such a kind of approach is critical.

A major departure from the solutions discussed above can be found in [105], where meticulous knowledge at the system level is used off-line to automatically generate firewall and IDS rules as needed. In this case, the obtainable accuracy is known a priori (i.e. it is embedded in the firewall and IDS rules) and the performance impact is quite low because the overhead only consists of evaluating the rules themselves.

Finally, a rough estimation of main IDS issues covered in research papers could be derived in a way similar to the discussion already carried out in the prevention subsection. In the case of IDSs, however, accuracy and performance impact are the two topics of utmost importance.

From this point of view (and adopting strict selection criteria) less than 67 % of the published papers has dealt with accuracy, whereas performance has been explicitly tackled and discussed only by 27 % of them. These two indicators are low enough to conclude that much more effort and future studies are strongly needed in this area.

VI. Conclusions

Some recent striking events, that have also caused echoes in the news, media and non-specialized press, have shown that cyber-threats to IACS can no longer be considered as unlikely possibilities but, unfortunately, they are real facts.

Indeed, there is increasing concern among security experts, managers and people in charge of administering private and public critical infrastructures, that IACS could be easily targeted and damaged by means of attacks to their underlying networks. Excessive scaremongering has to be avoided, of course, but it is clear that consequences might be very serious in that case, as injuries to people and damages to things and the environment could likely occur.

While advanced techniques have been continuously developing, since several years, to protect office and business networks from information technology-based attacks, the same has not happened for IACS, mainly because of their peculiarities and priorities in security requirements, that make them different from conventional computing systems. So, while sophistication in cyber-attacks always improves, security management in IACS has remained more or less the same (that is often at a very unsatisfactory level) until recently. The interconnection of subsystems through public communication networks and the Internet, the introduction of wireless communication technologies and the increasing adoption of general-purpose operating systems and s/w available off-the-shelf, has then significantly contributed to increase the exposure of IACS to security threats.

This paper has dealt with the current situation of security in IACS. We have shown that, nowadays, the most relevant standard proposals agree on considering the management of security in IACS as a never-ending cyclical process that moves through a well-defined set of main phases, including risk assessment, development and deployment of countermeasures and validation and monitoring of results. Each phase has then been addressed in this paper, with respect to the current state of the art, to give an idea of the problems and scientific/technical challenges that have to be tackled in order to reduce the security risks under a pre-defined, acceptable threshold.

We have also shown that the management of security in IACS involves two main hierarchical levels, which consider the whole system and its security controls respectively. The boundary between these two layers, however, is not sharp and (partially) overlapping areas and shared elements exist for both of them. Stateful IDSs, for instance, are a clear example where the global knowledge of the system can help in developing effective countermeasures at the component level. On the other hand, the efficacy of security techniques based on innovative approaches in the design of control systems also depends on the degree of integration of mechanisms implemented at the lower level.

In this framework, the study and development of automatic/semiautomatic analysis IT techniques and tools that are able to deal with security at a global (system) level, can be of significant help in making each phase of the management process easier and more efficient. Indeed, we think that, because of the complexity and size of many IACS, quick and effective security management decisions and (re)actions will become harder to take in the near future, so that the scientific community is expected to propose and develop new, advanced techniques to support IACS security experts and managers in

carrying out their tasks.

REFERENCES

[1] *Security for Industrial Automation and Control Systems Part 1: Terminology, Concepts, and Models*, ANSI/ISA Std. 99.00.01-2007.

[2] K. Stouffer, J. Falco, and K. Scarfone, "Guide to Industrial Control Systems (ICS) Security," NIST SP 800-82, 2008.

[3] D. Dzung, M. Naedele, T. P. von Hoff, and M. Crevatin, "Security for Industrial Control Systems," *Proc. IEEE*, vol. 93, no. 6, pp. 1152–1177, 2005.

[4] G. N. Ericsson, "Cyber Security and Power System Communication - Essential Parts of a Smart Grid Infrastructure," *IEEE Trans. Power Del.*, vol. 25, no. 3, pp. 1501–1507, 2010.

[5] L. Pièrre-Cambacédès, M. Tritschler, and G. N. Ericsson, "Cybersecurity Myths on Power Control Systems: 21 Misconceptions and False Beliefs," *IEEE Trans. Power Del.*, vol. 26, no. 1, pp. 161–172, 2011.

[6] *Information Technology - Security Techniques - Information Security Management Systems - Requirements*, ISO/IEC Std. 27 001, 2005.

[7] *Reliability Standards - CIP (Critical Infrastructure Protection)*, NERC (North American Electric Reliability Corporation).

[8] R. C. Parks and E. Rogers, "Vulnerability Assessment for Critical Infrastructure Control Systems," *IEEE Security Privacy*, vol. 6, no. 6, pp. 37–43, 2008.

[9] *AGA 12, Part 1 - Cryptographic Protection of SCADA Communications Part 1: Background, Policies and Test Plan*, American Gas Association, 2006.

[10] *AGA 12, Part 2 - Performance Test Plan*, American Gas Association, 2006.

[11] *Electronic Records; Electronic Signatures (21CFR11). Code of Federal Regulations, Title 21, Volume 1, Part 11*, U.S. Food & Drug Admin. - Dept. of Health & Human Services, 2001.

[12] "Recommended Security Controls for Federal Information Systems and Organizations," NIST SP 800-53 rev. 3, 2010.

[13] *Security for Industrial Automation and Control Systems: Establishing an Industrial Automation and Control Systems Security Program*, ANSI/ISA Std. 99.02.01-2009.

[14] *Minimum Security Requirements for Federal Information and Information Systems*, NIST Std. FIPS 200, 2006.

[15] *Information Technology - Security Techniques - Information Security Management Systems - Overview and Vocabulary*, ISO/IEC Std. 27 000, 2009.

[16] *Information Technology - Security Techniques - Code of Practice for Information Security Management*, ISO/IEC Std. 27 002, 2005.

[17] *Information Technology - Security Techniques - Information Security Management Guidelines for Telecommunications Organizations Based on ISO/IEC 27002*, ISO/IEC Std. 27 011, 2008.

[18] *Health Informatics – Information Security Management in Health Using ISO/IEC 27002*, ISO/IEC Std. 27 799, 2008.

[19] "Guide for Applying the Risk Management Framework to Federal Information Systems - A Security Life Cycle Approach," NIST SP 800-37 rev. 1, 2010.

[20] *Information Technology - Security Techniques - Information Security Risk Management*, ISO/IEC Std. 27 005, 2008.

[21] "Managing Information Security Risk - Organization, Mission, and Information System View," NIST SP 800-39, 2011.

[22] *Information Technology - Security Techniques - Information Security Management - Measurement*, ISO/IEC Std. 27 004, 2009.

[23] E. Chew, M. Swanson, K. Stine, N. Bartol, A. Brown, and W. Robinson, "Performance Measurement Guide for Information Security," NIST SP 800-55 rev. 1, 2008.

[24] ANSI/ISA, "Security Technologies for Industrial Automation and Control Systems," ANSI/ISA, Tech. Rep. TR99.00.01-2007, 2007.

[25] *Information Technology - Security Techniques - Information Security Management System Implementation Guidance*, ISO/IEC Std. 27 003, 2010.

[26] G. N. Ericsson, "Information Security for Electric Power Utilities (EPUs) - CIGRÉ Developments on Frameworks, Risk Assessment, and Technology," *IEEE Trans. Power Del.*, vol. 24, no. 3, pp. 1174–1181, 2009.

[27] ——, "Toward a Framework for Managing Information Security for an Electric Power Utility - CIGRÉ Experiences," *IEEE Trans. Power Del.*, vol. 22, no. 3, pp. 1461–1469, 2007.

[28] S. D. Guikema and T. Aven, "Assessing risk from intelligent attacks: A perspective on approaches," *Reliability Engineering & System Safety*, vol. 95, no. 5, pp. 478–483, 2010.

[29] T. Aven, "A unified framework for risk and vulnerability analysis covering both safety and security," *Reliability Engineering & System Safety*, vol. 92, no. 6, pp. 745–754, 2007.

[30] A. M. Koonce, G. E. Apostolakis, and B. K. Cook, "Bulk Power Risk Analysis: Ranking Infrastructure Elements According to Their Risk Significance," *Int. Journal of Electrical Power & Energy Systems*, vol. 30, no. 3, pp. 169–183, 2008.

[31] M. H. Henry, R. M. Layer, K. Z. Snow, and D. R. Zaret, "Evaluating the risk of cyber attacks on SCADA systems via Petri net analysis with application to hazardous liquid loading operations," in *Proc. IEEE Conf. on Technologies for Homeland Security (HST)*, 2009, pp. 607–614.

[32] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly, "Identifying, Understanding, and Analyzing Critical Infrastructure Interdependencies," *IEEE Control Syst. Mag.*, vol. 21, no. 6, pp. 11–25, 2001.

[33] N. K. Svendsen and S. D. Wolthusen, "Connectivity Models of Interdependency in Mixed-Type Critical Infrastructure Networks," *Information Security Technical Report*, vol. 12, no. 1, pp. 44–55, 2007.

[34] W. Kröger, "Critical infrastructures at risk: A need for a new conceptual approach and extended analytical tools," *Reliability Engineering & System Safety*, vol. 93, no. 12, pp. 1781–1787, 2008.

[35] E. Zio and G. Sansavini, "Modeling Interdependent Network Systems for Identifying Cascade-Safe Operating Margins," *IEEE Trans. Rel.*, vol. 60, no. 1, pp. 94–101, 2011.

[36] M. Ouyang, L. Hong, Z.-J. Mao, M.-H. Yu, and F. Qi, "A methodological approach to analyze vulnerability of interdependent infrastructures," *Simulation Modelling Practice and Theory*, vol. 17, no. 5, pp. 817–828, 2009.

[37] C. W. Johnson and K. McLean, "Tools for Local Critical Infrastructure Protection: Computational Support for Identifying Safety and Security Interdependencies between Local Critical Infrastructures," in *Proc. 3rd IET Int. Conf. on System Safety*, 2008, pp. 1–6.

[38] S. Patel and J. Zaveri, "A Risk-Assessment Model for Cyber Attacks on Information Systems," *Journal of Computers*, vol. 5, no. 3, pp. 352–359, 2010.

[39] P. A. S. Ralston, J. H. Graham, and J. L. Hieb, "Cyber Security Risk Assessment for SCADA and DCS Networks," *ISA Transactions*, vol. 46, no. 4, pp. 583–594, 2007.

[40] Y. Y. Haimes, "Hierarchical Holographic Modeling," *IEEE Trans. Syst., Man, Cybern. A*, vol. 11, no. 9, pp. 606–617, 1981.

[41] K. G. Crowther and Y. Y. Haimes, "Application of the Inoperability Input-Output Model (IIM) for Systemic Risk Assessment and Management of Interdependent Infrastructures," *Systems Engineering*, vol. 8, no. 4, pp. 323–341, 2005.

[42] H. Kumamoto and E. J. Henley, *Probabilistic Risk Assessment and Management for Engineers and Scientists*. Wiley-IEEE Press, 2000.

[43] "Probabilistic Risk Assessment Procedure Guide for NASA Managers and Practitioners," NASA/SP-2011-3421, 2nd ed., 2011.

[44] Y. Y. Haimes, S. Kaplan, and J. H. Lambert, "Risk filtering, ranking, and management framework using hierarchical holographic modeling," *Risk Analysis*, vol. 22, no. 2, pp. 383–397, 2002.

[45] D. J. Leversage and E. J. Byres, "Estimating a System's Mean Time-to-Compromise," *IEEE Security Privacy*, vol. 6, no. 1, pp. 52–60, 2008.

[46] R. Setola, S. De Porcellinis, and M. Sforna, "Critical Infrastructure Dependency Assessment Using the Input-Output Inoperability Model," *Int. Journal of Critical Infrastructure Protection*, vol. 2, no. 4, pp. 170–178, 2009.

[47] G. Oliva, S. Panzieri, and R. Setola, "Agent-Based Input-Output Interdependency Model," *Int. Journal of Critical Infrastructure Protection*, vol. 3, no. 2, pp. 76–82, 2010.

[48] W. S. Lee, D. L. Grosh, F. A. Tillman, and C. H. Lie, "Fault Tree Analysis, Methods, and Applications. A Review," *IEEE Trans. Rel.*, vol. 34, no. 3, pp. 194–203, 1985.

[49] C.-W. Ten, G. Manimaran, and C.-C. Liu, "Cybersecurity for Critical Infrastructures: Attack and Defense Modeling," *IEEE Trans. Syst., Man, Cybern.*, vol. 40, no. 4, pp. 853–865, 2010.

[50] B. Schneier, *Secrets and Lies: Digital Security in a Networked World*. Wiley Publishing, 2000.

[51] J. Andrews and S. Dunnett, "Event-Tree Analysis Using Binary Decision Diagrams," *IEEE Trans. Rel.*, vol. 49, no. 2, pp. 230–238, 2000.

[52] B. Wei, "A Unified Approach to Failure Mode, Effects and Criticality Analysis (FMECA)," in *Proc. Annual Symp. on Reliability and Maintainability*, 1991, pp. 260–271.

[53] E. Babeshko, V. Kharchenko, and A. Gorbenko, "Applying F(I)MEA-technique for SCADA-Based Industrial Control Systems Dependability Assessment and Ensuring," in *Proc. 3rd Int. Conf. on Dependability of Computer Systems (DepCoS-RELCOMEX)*, 2008, pp. 309–315.

[54] F. Baiardi, C. Telmon, and D. Sgandurra, "Hierarchical, model-based risk management of critical infrastructures," *Reliability Engineering & System Safety*, vol. 94, no. 9, pp. 1403–1415, 2009.

[55] A. Bobbio, E. Ciancamerla, S. Di Blasi, A. Iacomini, F. Mari, I. Melatti, M. Minichino, A. Scarlatti, E. Tronci, R. Terruggia, and E. Zendri, "Risk Analysis Via Heterogeneous Models of SCADA Interconnecting Power Grids and Telco Networks," in *Proc. 4th Int. Conf. on Risks and Security of Internet and Systems (CRiSIS)*, 2009, pp. 90–97.

[56] E. H. Spafford, "Cyber Security: Assessing Our Vulnerabilities and Developing an Effective Defense," in *Proc. 2nd Annual Workshop on Information Privacy and National Security (ISIPS)*, ser. LNCS, vol. 5661, 2009, pp. 20–33.

[57] A. A. Cárdenas, S. Amin, and S. Sastry, "Research Challenges for the Security of Control Systems," in *Proc. 3rd USENIX Workshop on Hot Topics in Security (HOTSEC)*, 2008, pp. 1–6.

[58] A. A. Cárdenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. Sastry, "Challenges for Securing Cyber Physical Systems," in *Int. Workshop on Future Directions in Cyber-physical Systems Security*, 2009.

[59] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks Against Process Control Systems: Risk Assessment, Detection, and Response," in *Proc. 6th ACM Symp. on Information, Computer and Communications Security (ASIACCS)*, 2011, pp. 355–366.

[60] M. Cheminod, I. Cibrario Bertolotti, L. Durante, P. Maggi, D. Pozza, R. Sisto, and A. Valenzano, "Detecting Chains of Vulnerabilities in Industrial Networks," *IEEE Trans. Ind. Informat.*, vol. 5, no. 2, pp. 181–193, 2009.

[61] D. M. Nicol, W. H. Sanders, S. Singh, and M. Seri, "Usable Global Network Access Policy for Process Control Systems," *IEEE Security Privacy*, vol. 6, no. 6, pp. 30–36, 2008.

[62] M. Cheminod, I. Cibrario Bertolotti, L. Durante, and A. Valenzano, "Automatic Analysis of Security Policies in Industrial Networks," in *Proc. 8th IEEE Int. Workshop on Factory Communication Systems (WFCS)*, 2010, pp. 109–118.

[63] M. Cheminod, L. Durante, and A. Valenzano, "System Configuration Check Against Security Policies in Industrial Networks," in *Proc. 7th IEEE Int. Symp. on Industrial Embedded Systems (SIES)*, 2012, in press.

[64] D. M. Nicol, W. H. Sanders, M. Seri, and S. Singh, "Experiences Validating the Access Policy Tool in Industrial Settings," in *Proc. 43rd Hawaii Int. Conf. on System Sciences (HICSS)*, 2010, pp. 1–8.

[65] T. Stephanou, "Assessing and Exploiting the Internal Security of an Organization," SANS Institute, 2001.

[66] E. Nash. (2003) Hackers bigger threat than rogue staff. [Online]. Available: http://seclists.org/isn/2003/May/65

[67] D. Reinelt and M. Wolframm, "Security in Virtual Automation Networks," in *Proc. 13th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*, 2008, pp. 480–483.

[68] X. Yuefei, R. Song, L. Korba, L. Wang, W. Shen, and S. Lang, "Distributed Device Networks With Security Constraints," *IEEE Trans. Ind. Informat.*, vol. 1, no. 4, pp. 217–225, 2005.

[69] R. Chandia, J. Gonzalez, T. Kilpatrick, M. Papa, and S. Shenoi, "Security Strategies for SCADA Networks," in *Critical Infrastructure Protection*, ser. IFIP International Federation for Information Processing, 2007, vol. 253, pp. 117–131.

[70] D. Choi, H. Kim, D. Won, and S. Kim, "Advanced Key-Management Architecture for Secure SCADA Communications," *IEEE Trans. Power Del.*, vol. 24, no. 3, pp. 1154–1163, 2009.

[71] D. Choi, S. Lee, D. Won, and S. Kim, "Efficient Secure Group Communications for SCADA," *IEEE Trans. Power Del.*, vol. 25, no. 2, pp. 714–722, 2010.

[72] Y. Wang, "sSCADA: securing SCADA infrastructure communications," *Int. J. Commun. Netw. Distrib. Syst.*, vol. 6, no. 1, pp. 59–78, 2011.

[73] M. Long and C.-H. J. Wu, "Energy-Efficient and Intrusion-Resilient Authentication for Ubiquitous Access to Factory Floor Information," *IEEE Trans. Ind. Informat.*, vol. 2, no. 1, pp. 40–47, 2006.

[74] M. Lin, L. Xu, L. T. Yang, X. Qin, N. Zheng, Z. Wu, and M. Qiu, "Static Security Optimization for Real-Time Systems," *IEEE Trans. Ind. Informat.*, vol. 5, no. 1, pp. 22–37, 2009.

[75] H. Okhravi and D. M. Nicol, "Application of trusted network technology to industrial control networks," *Int. Journal of Critical Infrastructure Protection*, vol. 2, no. 3, pp. 84–94, 2009.

[76] A. A. Cárdenas, T. Roosta, and S. Sastry, "Rethinking Security Properties, Threat Models, and the Design Space in Sensor Networks: a Case Study in SCADA Systems," *Ad Hoc Networks*, vol. 7, no. 8, pp. 1434–1447, 2009.

[77] C. Alcaraz and J. Lopez, "A Security Analysis for Wireless Sensor Mesh Networks in Highly Critical Systems," *IEEE Trans. Syst., Man, Cybern. C*, vol. 40, no. 4, pp. 419–428, 2010.

[78] Z. Karakehayov, "Security - Lifetime Tradeoffs for Wireless Sensor Networks," in *Proc. 12th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*, 2007, pp. 646–650.

[79] C.-C. Lee, M.-S. Hwang, and I.-E. Liao, "Security Enhancement on a New Authentication Scheme With Anonymity for Wireless Environments," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1683–1687, 2006.

[80] T. Novak and A. Gerstinger, "Safety- and Security-Critical Services in Building Automation and Control Systems," *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3614–3621, 2010.

[81] W. Granzer, F. Praus, and W. Kastner, "Security in Building Automation Systems," *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3622–3630, 2010.

[82] M. Henseler, M. Rossberg, and G. Schaefer, "Credential Management for Automatic Identification Solutions in Supply Chain Management," *IEEE Trans. Ind. Informat.*, vol. 4, no. 4, pp. 303–314, 2008.

[83] J. O. Lauf and H. Sauff, "Secure Lightweight Tunnel for Monitoring Transport Containers," in *Proc. 3rd Int. Conf. on Security and Privacy in Communications Networks (SecureComm)*, 2007, pp. 484–493.

[84] T. Kilpatrick, J. Gonzalez, R. Chandia, M. Papa, and S. Shenoi, "Forensic analysis of SCADA systems and networks," *Int. J. Secur. Netw.*, vol. 3, no. 2, pp. 95–102, 2008.

[85] M. Cheminod, I. Cibrario Bertolotti, L. Durante, R. Sisto, and A. Valenzano, "Tools for cryptographic protocols analysis: A technical and experimental comparison," *Computer Standards & Interfaces*, vol. 31, no. 5, pp. 954–961, 2009.

[86] M. Cheminod, A. Pironti, and R. Sisto, "Formal Vulnerability Analysis of a Security System for Remote Fieldbus Access," *IEEE Trans. Ind. Informat.*, vol. 7, no. 1, pp. 30–40, 2011.

[87] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*. Springer, 2004.

[88] G. D. Sutter, J.-P. Deschamps, and J. L. Imana, "Modular Multiplication and Exponentiation Architectures for Fast RSA Cryptosystem Based on Digit Serial Computation," *IEEE Trans. Ind. Electron.*, vol. 58, no. 7, pp. 3101–3109, 2011.

[89] J. Zhu and J. Ma, "A New Authentication Scheme with Anonymity for Wireless Environments," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 231–235, 2004.

[90] D. E. Denning, "An Intrusion-Detection Model," *IEEE Trans. Softw. Eng.*, vol. 13, no. 2, pp. 222–232, 1987.

[91] U. Premaratne, J. Samarabandu, T. Sidhu, R. Beresh, and J.-C. Tan, "An intrusion detection system for iec61850 automated substations," *IEEE Trans. Power Del.*, vol. 25, no. 4, pp. 2376–2383, 2010.

[92] A. Carcano, I. Fovino, M. Masera, and A. Trombetta, "State-Based Network Intrusion Detection Systems for SCADA Protocols: A Proof of Concept," in *Proc. 4th Int. Workshop on Critical Information Infrastructures Security (CRITIS)*, ser. LNCS, vol. 6027, 2010, pp. 138–150.

[93] M. Long, C.-H. J. Wu, and J. Y. Hung, "Denial of Service Attacks on Network-Based Control Systems: Impact and Mitigation," *IEEE Trans. Ind. Informat.*, vol. 1, no. 2, pp. 85–96, 2005.

[94] W. Granzer, C. Reinisch, and W. Kastner, "Denial-of-Service in Automation Systems," in *Proc. 13th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*, 2008, pp. 468–471.

[95] O. Linda, T. Vollmer, and M. Manic, "Neural Network based Intrusion Detection System for Critical Infrastructures," in *Proc. Int. IEEE - INNS - ENNS Joint Conf. on Neural Networks (IJCNN)*, 2009, pp. 1827–1834.

[96] P. Düssel, C. Gehl, P. Laskov, J.-U. Bußer, C. Störmann, and J. Kästner, "Cyber-Critical Infrastructure Protection Using Real-Time Payload-Based Anomaly Detection," in *Proc. 4th Int. Workshop on Critical Information Infrastructures Security (CRITIS)*, ser. LNCS, vol. 6027, 2010, pp. 85–97.

[97] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. Nai Fovino, and A. Trombetta, "A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems," *IEEE Trans. Ind. Informat.*, vol. 7, no. 2, pp. 179–186, 2011.

[98] S. Cheung, B. Dutertre, M. Fong, U. Lindqvist, K. Skinner, and A. Valdes, "Using Model-Based Intrusion Detection for SCADA Networks," in *Proc. of SCADA Security Scientific Symp.*, 2007.

[99] R. Barbosa and A. Pras, "Intrusion Detection in SCADA Networks," in *Mechanisms for Autonomous Management of Networks and Services - Proc. 4th Int. Conf. on Autonomous Infrastructure, Management, and Security (AIMS)*, ser. LNCS, vol. 6155, 2010, pp. 163–166.

[100] D. Yang, E. Usynin, and J. W. Hines, "Anomaly-Based Intrusion Detection for SCADA Systems," in *Proc. 5th American Nuclear Society Int. Topical Meeting on Nuclear Plant Instrumentation, Controls, and Human Machine Interface Technology (ANS)*, 2006.

[101] P. Oman and M. Phillips, "Intrusion Detection and Event Monitoring in SCADA Networks," in *Critical Infrastructure Protection*, ser. IFIP International Federation for Information Processing, 2007, vol. 253, pp. 161–173.

[102] S. Shin, T. Kwon, G.-Y. Jo, Y. Park, and H. Rhy, "An Experimental Study of Hierarchical Intrusion Detection for Wireless Industrial Sensor Networks," *IEEE Trans. Ind. Informat.*, vol. 6, no. 4, pp. 744–757, 2010.

[103] K. Xiao, N. Chen, S. Ren, L. Shen, X. Sun, K. Kwiat, and M. Macalik, "A Workflow-Based Non-intrusive Approach for Enhancing the Survivability of Critical Infrastructures in Cyber Environment," in *Proc. 3rd Int. Workshop on Software Engineering for Secure Systems (SESS)*, 2007, pp. 4–10.

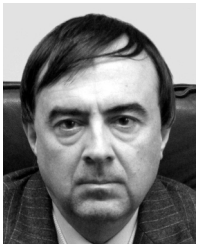
[104] M. Di Santo, A. Vaccaro, D. Villacci, and E. Zimeo, "A Distributed Architecture for Online Power Systems Security Analysis," *IEEE Trans. Ind. Electron.*, vol. 51, no. 6, pp. 1238–1248, 2004.

[105] H. Hadeli, R. Schierholz, M. Braendle, and C. Tuduce, "Leveraging Determinism in Industrial Control Systems for Advanced Anomaly Detection and Reliable Security Configuration," in *Proc. 14th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*, 2009, pp. 1–8.

Manuel Cheminod received the M.S. and Ph.D. degrees in computer engineering from Politecnico di Torino, Torino, Italy, in 2005 and 2010 respectively.

He is now working with the Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni (IEIIT). His current research interests include formal verification of cryptographic protocols and formal methods applied to vulnerability and dependability analysis in distributed networks.

Luca Durante is Senior Researcher with the Italian National Research Council (CNR). He is currently with Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni (IEIIT). He graduated in Electronic Engineering in 1992, and received a PhD degree in Computer Engineering in 1996, both from the Politecnico di Torino. He has co-authored about 40 scientific journal, conference papers and technical reports in the area of industrial communication protocols and formal techniques for distributed systems. He also served as a technical referee for several international conferences and journals. Currently his research interests include formal verification of cryptographic protocols, source-level model checking of software and network vulnerability and dependability analysis.



Adriano Valenzano (SM'09) is Director of Research with the National Research Council of Italy (CNR). He is currently with Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni (IEIIT), Torino, Italy, where he is responsible for researches concerning distributed computer systems, local area networks and communication protocols. Since 1983, he has been involved in many national and international research projects and has led a number of research teams in the information and communication technology areas. He has co-authored about 200 refereed journal and conference papers in the area of computer engineering. Adriano Valenzano was also awarded as the co-author of the best papers presented at the 5th and 8th IEEE Workshops on Factory Communication Systems (WFCS 2004 and WFCS 2010). He has served as a technical referee for several international journals and conferences, also taking part in the program committees of international events of primary importance. He served as a general co-chairman of the 6th IEEE Int. Workshop on Factory Communication Systems (WFCS 2006), and since 2008 has taken part in both the Steering Committee of the IEEE WFCS Series and the Advisory Committee of the IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA) Series. Since 2007 he has been serving as an Associate Editor for the IEEE Transactions on Industrial Informatics. Adriano Valenzano is a Senior Member of the IEEE. He is also vice-president of the Piedmont chapter of the Italian National Association for Automation (ANIPLA).