C. Alcaraz, J. Lopez, and S. Wolthusen, "Policy Enforcement System for Secure Interoperable Control in Distributed Smart Grid Systems", *Journal of Network and Computer Applications*, vol. 59, pp. 301314, 2016.
NICS Lab. Publications: https://www.nics.uma.es/publications

Policy Enforcement System for Secure Interoperable Control in Distributed Smart Grid Systems

Cristina Alcaraz[1,2], Javier Lopez[1], and Stephen Wolthusen[2,3]

[1]Computer Science Department, University of Málaga, Spain

[2]Information Security Group, Department of Mathematics, Royal Holloway

University of London. Egham TW20 0EX, United Kingdom

[3]Norwegian Information Security Laboratory, Gjøvik University College, Norway

{alcaraz,jlm}@lcc.uma.es

stephen.wolthusen@rhul.ac.uk

August 10, 2017

Abstract

Interoperability of distributed systems in charge of monitoring and maintaining the different critical domains belonging to Smart Grid scenarios comprise the central topic of this paper. Transparency in control transactions under a secure and reliable architecture is the aim of the policy enforcement system proposed here. The approach is based on the degree of observation of a context and on the *role-based access control* model defined by the IEC-62351-8 standard. Only authenticated and authorised entities are able to take control of those distributed elements (e.g., IEC-61850 objects) located at distant geographical locations and close to the critical infrastructures (e.g., substations). To ensure the effectiveness of the approach, it is built on graphical-theoretical formulations corresponding to graph theory, where it is possible to illustrate power control networks through power-law distributions whose monitoring relies on *structural controllability* theory. The interconnection of these distributions is subject to a network architecture based on the concept of the *supernode* where the interoperability depends on a simple rule-based expert system. This expert system focuses not only on accepting or denying access, but also on providing the means to attend to extreme situations, avoiding, as much as possible, the overloading of the communication. Through one practical study we also show the functionalities of the approach and the benefits that the authorisation itself can bring to the *interoperability*.

Keywords: Smart Grid, Distributed Control Systems, Controllability, Interoperability, Policy Enforcement, Access Control

1 INTRODUCTION

We have been witnesses to the enormous progress made in the different Smart Grid domains in recent years [1, 2, 3]. Control systems, (power generation, transmission and distribution) substations, service providers, markets and customers together, make up a whole that enables the exchange of information and optimises the power production according to the true demand. The information is forwarded through complex and dynamic communication infrastructures with the capacity to connect multiple and heterogeneous systems [4, 5]. An array, ranging from local and small networks to large communication systems with full access to control objects (e.g., smart meters, sensors, charging points, RTUs (Remote Terminal Units), gateways, etc.), generally installed in distant locations and close to the critical infrastructures. However, when the proposal consists of moving towards the connectivity of different technologies belonging to different owners, manufactures or vendors with multiple types of access and security policies, issues related to interoperability can arise [4, 3, 6, 7].

Any security breach, conflict of format or operational delay caused by the heterogeneity of systems can trigger *integrity* and *availability* problems in the control, complicating the interpretation of the data itself or the execution of commands. This may even affect the *safety* of the entire power grid, and even its *stability* [6]. For this reason, our aim is not only to interconnect several control infrastructures but also to protect their monitoring and supervision tasks. The Industrial Control System Cyber Emergency Response Team (ICS-CERT) recently reported in [8] the number of vulnerabilities received in control systems in the year 2013 (181 incidents in total). According to this report, the authentication flaws are at the head of the number of incidents reported, considering it to be the most abundant vulnerability in 2013, with a register of 58% of the total. In light of this, the security has to encompass a set of requirements, amongst them: *access control* and *security policy management* because (i) any unauthorised access to restricted devices may become a threat, and (ii) authorised access under different security policies may hamper the supervision tasks.

One way of ensuring a secure and interoperable communication between systems belonging to different organisations could be through intermediary policy enforcement systems with support for dynamic handling of access and security policies. Through them it would be possible to prevent unknown access and filter operations in the field, resulting in a decision-making system with the capability to adapt the access to the type of context. For example, V. Kapsalis *et al.* presented in [9] a dynamic context-awareness access control architecture for the provision of e-services where the system can authenticate and authorise access according to the context, and even learn from said context. This functional feature has also been tailored to the proposal described in this paper together with a Role-Based Access Control (RBAC)-based least privilege scheme defined by the IEC-62351 standard [10, 11]. Concretely, the approach is based on a decision engine driven by a rule-based expert system capable of validating the access according to a set of factors: (i) the roles and permissions assigned to the subject; (ii) the type of context and the criticality of such a context; (iii) the type of action to be executed by an object (the destination node); and (iv) its accessibility degree.

The IEC-62351-8 is part of the IEC-62351 series [12] that establishes end-to-end security in power systems and the protection of the communication channels. In this

case, and through IEC-62351-8, RBAC is recognised as a potentially efficient mechanism for wide use in power systems and distributed services. Only authorised users and automated agents can gain access to restrictive data objects (e.g., IEC-61850 objects [13]) such as measurements, status variables or parameters. Moreover, through RBAC it is possible to reallocate system controls and their security as defined by the organisation policy, where the purpose is: (i) to introduce authorisation aspects under the category of subject-roles-rights; (ii) boost role-based access control in the power system management; and (iii) enable heterogeneity and interoperability between different elements of a system. Moreover, Li *et al.* in [14] underline that the RBAC technique in Smart Grids can enhance the reliability of the connections and survivability with a greater level of granularity. This analysis is also supported by M. Majdalawieh *et al.* in [15] through their generalised RBAC model for SCADA (Supervisory Control and Data Acquisition) systems. Similarly, H. Cheung *et al.* in [16] define a XML (eXtensible Markup Language)-based role-based model for establishing trust and role assignments to users belonging to different microgrid domains under the coordination of their respective central systems. This way of subdividing Smart Grid areas into regions is also considered by Rosic *et al.* in [17] to propose a RBAC-based access control mechanism dependent on the area of responsibility. Regarding policy enforcement and interoperability in Smart Grid environments, N. Kuntze *et al.* in [18] propose the use of smart energy gateways to establish trust relationships between parties (the energy grid, the control system and the customer side) using asymmetric key cryptography and cryptographic hash functions. Similarly, A. Veichtlbauer *et al.* in [19] provides a middle-ware architecture based on RBAC and policy decision and enforcement points to collect data streams from multiple sources connected to the Advanced Metering Infrastructure (AMI) in a standardised format. But beyond this, more investigation is still necessary to expand functionalities and offer more automated solutions.

In order to illustrate monitoring scenarios, our research centres on studies based on graph theory. The deployment of networks depends on graphical-theoretical interpretations where the control is based on the *structural controllability* theory introduced by C. Lin in [20] and on the concept of *power domination* defined by T. Haynes *et al.* in [21]. For the interconnection of these graphs, a *decentralised architecture* based on the concept of the *supernode* is also adapted to identify the Policy Decision Points (PDPs) within the control structure and provide an attractive way to distribute and filter operational activities. Once modelled, our main contributions later concentrate on addressing the interoperability through an expert system capable of understanding the IEC-62351-8 standard and the criticality degree of a context. This also means that the analysis carried out in this paper follows an incremental structure based on three fundamental parts: (i) the logical modelling of virtual control networks (through graph theory, structural controllability and power dominance); (ii) the theoretical construction of a decentralised network architecture (through supernode theory); and (iii) automated interoperability of networks through Policy Enforcement Points (PEPs) and an expert system.

The remainder of this paper is structured in five sections. Section 2 describes the network architecture and the conditions for control, whereas the policy enforcement architecture is presented in Section 3 together with its construction blocks related to authentication, authorisation, security policy management and context. This architec-

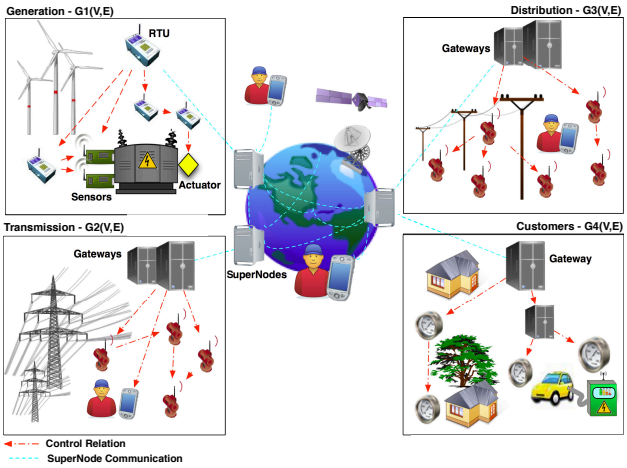

Figure 1: General architecture for distributed control

tures is later analysed using a case study in Section 4 to validate and discuss practical behaviours of authorisation. Finally, our conclusions together with future work are presented in Section 5.

2 General Architecture for Complex, Dynamic and Heterogeneous Networks

So as to model control networks from a conceptual standpoint but approximated to real applications, the network architecture proposed here is based on the concept of the supernode [22]. A supernode system is a decentralised architecture composed of fixed nodes with the computational capacities for acting as proxies and offering peer-to-peer connection via the Internet. Through these proxies it is possible to connect different types of networks, the connection of which is also dependent on intermediary nodes serving as gateways with enough capabilities to compute data streams and identify control objects (e.g., RTUs working at $\sim$ 22MHz-200MHz with 256 bytes-64MB RAM, 8KB-32MB flash memory and 16KB-256KB EEPROM). Specifically, these gateways are in charge of controlling the incoming and outgoing communications from their networks towards their closest supernodes, so that part of the communication must include information related to the destination gateway (e.g., its IP – its *gwID*). When the interconnection system is extremely complex and distant, supernodes can also forward the request to one another until reaching the closest supernode with connection to the destination gateway (note that a gateway can work as a supernode). Figure 1 shows the application scenario, where the deployment of each sub-network is based on graphical-theoretical distributions, where each sub-graph $\mathscr{G}_i(V,E)$ constitutes a topological network with V objects (e.g., RTUs, smart sensors, meters, servers, etc.) and E communication links for supervision and data acquisition.

To recreate structures similar to real power control systems, such as those described in [23], each sub-network has to be based on general power-law distributions of the type $y \propto x^{\alpha}$ (e.g., the Power-Law Out-Degree (PLOD) [24]) or on scale-free distributions. A scale-free random graph is a network whose degree distribution exhibits a power law such as described in the Barabási-Albert (BA) model [25] with its associated preferential attachment. In order to embed 'control' inside these virtual networks, topics related to *control theory* with its implicit dominance concept [26] are also considered in our approach. *Controllability* was introduced by R. Kalman in the 60s [27]. It consists in a rigorous and well-understood framework for the design and analysis of, not only control systems, but also of networks in which a control relation between vertices ($v_i \longrightarrow v_j$) exists. The framework follows the formulation given below:

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t),\ x(t_0) \ = \ x_0 \tag{1}$$

where $x(t)$ is a vector $(x_1(t), \ldots, x_n(t))^T$ representing the current state of a system with n nodes at time t; $\mathbf{A}$ is an adjacency matrix $n \times n$ giving the network topology that identifies interaction between nodes, $\mathbf{B}$ an *input* matrix $n \times m$, where $m \leq n$, identifies the set of nodes controlled by a time-dependent *input vector* $u(t)$ which forces the system to a desired state in a finite number of steps. A system as defined in Equation 1 is said to be controllable, if and only if, the Kalman's rank criterion is met; i.e., $\text{rank}[\mathbf{B}, \mathbf{AB}, \mathbf{A}^2\mathbf{B}, \ldots, \mathbf{A}^{n-1}\mathbf{B}] = n$. However, and unfortunately, this formulation becomes prohibitive for large-scale networks like Smart Grids, where the number of nodes exponentially grows. An alternative to this problem is precisely the well-known *structural controllability theory*, which is described in more detail in the next section.

2.1 Structural Controllability in Super Node Architectures

Structural controllability consists in a graphical-theoretical interpretation where $\mathscr{G}(V,E)$ keeps the control conditions given in Equation 1. In this case, $\mathscr{G}(V,E)$ corresponds to an acyclic graph capable of imposing the control direction composed of $V = V_{\mathbf{A}} \cup V_{\mathbf{B}}$ nodes and $E = E_{\mathbf{A}} \cup E_{\mathbf{B}}$ edges. In addition, the input vector u of Equation 1 matches the set of vertices $V_{\mathbf{B}}$ containing those nodes with the ability to inject control (e.g., commands) into the network, also known as *driver nodes*, and hereinafter as n_d.

For the identification and selection of the minimum set of driver nodes ($\mathbf{N}_D$, where $n_{d_i} \in \mathbf{N}_D$) within a given network, the POWER DOMINATING SET (PDS) problem, originally introduced by T. Haynes *et al.* in [21], is a suitable technique for our purpose. This interest lies in the nature of the problem itself, in which the original structures of electric power networks and the need for the efficient 'monitoring' of such networks were considered as part of the analysis in [21]. The basis of this study is also supported by the traditional DOMINATING SET (DS) problem [28], which has proved to be a useful tool in multiple scenarios related to wireless networks and clustering [29, 30].

As also stated in [21], the PDS problem was originally introduced to be computed in function of a set of observation rules, but was later simplified by Kneis *et al.* [31] into two fundamental observation rules:

OR1 *A vertex in* $\mathbf{N}_D$ *observes itself and all its neighbours*; this observation rule is directly associated with the DS problem.

OR2 *If an observed vertex v of degree $d \geq 2$ is adjacent to $d-1$ observed vertices, the remaining unobserved vertex becomes observed as well.* As **OR1** $\subseteq$ **OR2**, it means that **OR1** is implicitly contained by **OR2**, such that **OR1** represents the degree of observation within a network and **OR2** the power dominance.

Algorithm 2.1: STRUCTURAL CONTROLLABILITY IN SUPERNODE SYSTEMS ($nodes, network$)

output $(\mathbf{N}_D, \mathscr{G}(V,E))$
procedure GENERATE NETWORK $(nodes, network, \alpha)$
output $(\mathscr{G}(V,E))$
local $gateway, cyclic$;
$cyclic \leftarrow$ **true** ;
while $cyclic$
do
$\mathscr{G}(V,E) \leftarrow$ NETWORK$(nodes, network, \alpha)$; $gateway \leftarrow nodes + 1$;
for each $v_i \in V$
do
if $parents(v_i) = \oslash$
then $Establish\ relationship\ (gateway, v_i) \in E$;
if ISDAG[a]$(\mathscr{G}(V,E))$ **and** ISCONNECTED$(\mathscr{G}(V,E))$
then
$cyclic \leftarrow$ **false** ;
$V \leftarrow V \cup \{gateway\}$;
return $(\mathscr{G}(V,E);)$

procedure EMBEDDING CONTROL$(\mathscr{G}(V,E))$
output $(\mathbf{N}_D)$
local $N, DS, gateway$;
comment: OR1 starting from the gateway.
$gateway \leftarrow V(last)$; $DS^b \leftarrow \{gateway\}$;
$N^c \leftarrow N \cup N(gateway)\ \forall\ v_i \in V \setminus (gateway, v_i) \in E$
while $V - (DS \cup N) \neq \oslash$
do
$Randomly\ choose\ a\ v_w \in V - (DS \cup N(DS))$;
$DS \leftarrow DS \cup \{v_w\}$;
$N \leftarrow N \cup N(v_w)\ such\ that\ \forall\ v_i \in V \setminus (v_w, v_i) \in E$;
return $(\mathbf{OR2}(\mathscr{G}(V,E), DS))^d$

main
$\mathscr{G}(V,E) \leftarrow$ GENERATE NETWORK$(nodes, network)$;
$\mathbf{N}_D \leftarrow$ EMBEDDING CONTROL$(\mathscr{G}(V,E))$;

[a] DAG is synonymous with a directed acyclic graph (digraph), which can be computed through the well-known *depth first search* algorithm with a complexity order of $O(n+e) = O(n)$, such that $n = \mid V \mid$ and $e = \mid E \mid$.
[b] DS is the set of observed nodes; i.e., the dominating set or **OR1**.
[c] N is the set of neighbours.
[d] **OR2** is a function defined in [32], where **OR1** $\subseteq$ **OR2** with a cost of $O(n^2)$ [34].

Both rules have been extensively analysed in recent papers [32, 33, 34] to explore behaviours when the structural controllability is being perturbed [32, 33], and to evaluate its resilience against threats to the availability of resources (nodes and links) [34]. Regarding the complexity of PDS, T. Haynes *et al.* showed the $\mathscr{NP}$-hardness of the PDS problem, also evaluated by R. Downey *et al.* [35] who concluded that it is only $\Theta(\log n)$-approximable for general graphs. On the other hand, both rules tend to produce, by definition, hierarchical networks with several access points (i.e., several roots)

within a graph (a network), which makes the adaptation of the supernode architecture with connection to unique gateways, complicated. An easy way to address this issue and simplify the access to just one point is to: (i) force the power-law distributions to keep up a dependence on a single node (the gateways); and (ii) inject the control from the gateways. However, these two steps imply a further two fundamental conditions to be met: (i) keep the acyclicity of the network, and the control direction from the gateway; and (ii) respect the control conditions, **OR1** and **OR2**, at all times.

Note that these two conditions are outlined in Algorithm 2.1, which needs, as an input parameter, the type of network distribution to be produced (e.g., BA or PLOD) together with its connectivity degree, α. The α value has to be small (e.g., $\alpha \sim 0.1, 0.2$ for PLOD or $\alpha \sim 3$ for BA) to illustrate sparse graphs with similar structures to real scenarios. With this distribution $\mathscr{G}(V,E)$, the procedure '*Generate Network*' has to define a new node in V (the gateway) with a direct connectivity to those nodes of $\mathscr{G}(V,E)$ with no parents (i.e., $\forall\ v_i$, $(v_i,v) \notin E$), such that the resulting graph has to be digraph and connected. Regarding the latter condition, Algorithm 2.1 has to start the first observation rule (**OR1**) using the gateway as the first observation element so that it can be observed by, at least, itself, and in this way satisfy **OR1**. Once the gateway has become part of the monitoring of the network, the selection of the rest of the driver nodes is completely arbitrary, as described in [32]. Also note that we are aware that both the supernodes and gateways are single failure points for supervision, but protection against faults is beyond of the scope of the research presented here and should form part of future work.

The result is a roadmap of interconnections capable of representing control contexts of the real world where their elements could be situated in distant locations over large-scale distributions. This is a very attractive feature that requires protection mechanisms for those who are monitoring cyber-physical elements (e.g., smart meters, RTUs, gateways, etc.) against connections from anywhere, at any time and in anyway. This protection should consist of mechanisms with minimal services related to *authentication, authorisation* and *interoperability*, which are described in depth in the following section.

As for the security of communication channels, it is strongly assumed that they are protected following, for example, the security measures given by the IEC-62351 series (see Table 1), which suggests TCP/IP security services. This includes TLS (Transport Layer Security)/SSL (Secure Sockets Layer) together with key exchange algorithms such as Diffie-Hellman (DH) or RSA; digital signature through DSS (Digital Signature Standard) and RSA; encryption algorithms such as RCA-128, 3DES (Triple-Data Encryption Standard) or AES (Advanced Encryption Standard)-128/256 bits of key size; as well as the secure hash algorithm, also known as SHA. For example, the IEC-62351-4 standard [36] specifically recommends the cypher suite TLS_DH_DSS_WITH_AES_256_SHA for communications between the control center and substations; whereas the IEC-62351-6 [37] recommends the suite TLS_DH_RSA_WITH_AES_128_SHA for communications within substations based on IEC-61850 objects. However, these measures are not sufficient to guarantee protection of the channels [38, 39]. It is also necessary to configure Virtual Private Networks (VPNs) between peers; depend on protection mechanisms such as firewalls and intrusion detection systems; and use additional security approaches [39]. These approaches could, for example, help the obfuscation of

Table 1: Security policies retrieved from the IEC-62351 series

Key Exchange	Signature	Encryption	Hash
TLS_RSA_		WITH_RCA_128_	SHA
TLS_RSA_		WITH_3DES_sDE_CBC_	SHA
TLS_DH_	DSS_	WITH_3DES_sDE_CBC_	SHA
TLS_DH_	RSA_	WITH_3DES_sDE_CBC_	SHA
TLS_DHE_	DSS_	WITH_3DES_sDE_CBC_	SHA
TLS_DHE_	RSA_	WITH_3DES_sDE_CBC_	SHA
TLS_DH_	DSS_	WITH_AES_128_	SHA
TLS_DH_	RSA_	WITH_AES_256_	SHA
TLS_DH_	DSS_	WITH_AES_256_	SHA
TLS_DH_		WITH_AES_128_	SHA
TLS_DH_		WITH_AES_256_	SHA

IP addresses such as the MT6D proposed by Groat et.al. in [40], or offer the means to ensure wide-area situational awareness, forensics and learning, trust management and privacy, self-healing, etc. Many of them described in detail in [38, 39].

3 Policy Enforcement for Structural Controllability Protection

With the network architecture proposed in Section 2 in mind, this section establishes the means by which control objects can be protected from external access. A control object is a necessary element for supervision and data acquisition, and it represents either a $n_{d_i} \in \mathbf{N}_D$ (e.g., gateways, base stations, servers, RTUs, etc.) or an element $\notin \mathbf{N}_D$ (e.g., sensors, actuators, smart meters, etc.) under the 'observation' of, at least, one n_{d_i}; i.e., an RTU $\longrightarrow$ a sensor. These elements have to be protected from physical or logical entities (e.g., human operators, automated software processes, manufactures, utilities or devices) that wish to manipulate critical data or collapse peers; a need becomes that much greater when the control objects present strong hardware and software constraints (e.g., smart sensors with $\sim$ 4MHz-8MHz, 4KB-16KB RAM and 48KB- 256KB ROM or smart meters $\sim$ 8-50MHz, 4KB-32KB RAM and 32-512KB flash memory).

This way of connecting with control objects is not too far removed from the control systems in the real world. SCADA systems are modernising their architectures so as to connect with private and public networks, and adapting diverse technologies and applications over large-scale distributions (e.g., real-time control of smart cities and their advanced metering infrastructures). Cloud-computing, backhaul, wide area, local area, field area and neighbourhood area networks are all clear examples of this progress. However, it is also clear that these advances may also bring about serious problems associated with the secure and reliable interconnection of dynamic and complex networks whose connections might come from any location [38, 39]. Management of unauthorised access, security controls according to the organisation's policies, and the authorised access in extreme situations should be predominant requirements in these new monitoring environments.

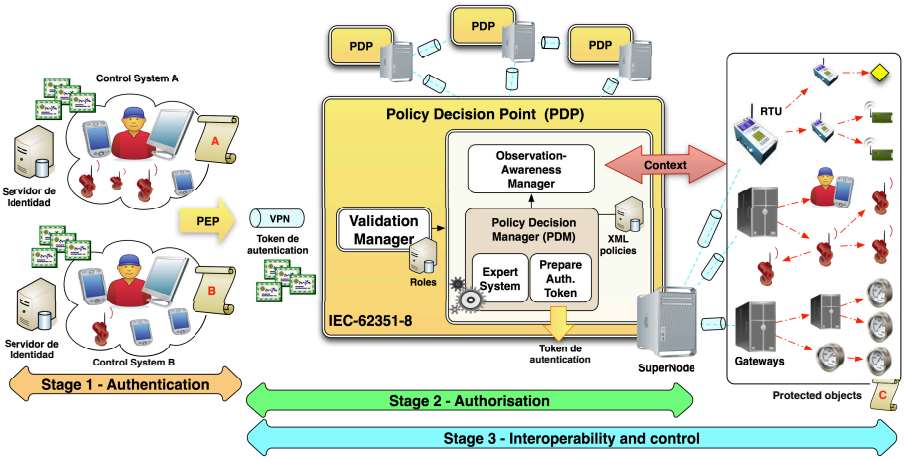

Figure 2: Policy enforcement architecture together with its construction components

Given this, a policy enforcement point together with its distributed PDPs are proposed here to guarantee interoperability between two or more peers. The architecture is based on the concept of the least privilege through RBAC so as to ensure authentication, authorisation and interoperability, and on the properties of graph theory to revise the observational degree of a context. All these aspects are discussed in detail in the following two sections.

3.1 Authentication, Authorisation, Interoperability

As mentioned, this section is devoted to the general architecture of our approach integrated inside a supernode system, and presents its construction components related to authentication, authorisation and interoperability. Figure 2 illustrates the three main execution phases of our approach:

Stage 1: Authentication In this stage each subject belonging to an entity has to authenticate itself to its own identity server, and hence in its own infrastructure. Once the subject has been authenticated, the server provides it with an *authentication token* containing, at least, the information about the subject such as its identification and roles assigned, and information about the destination node. A simple structure of an authentication token is as follows: $\{objectID,\ subjectID,\ roleID_{primary},\ \{roleID_1,...,roleID_n\}_{secoundary},\ time\ stamp,\ expiration\ of\ the\ token,\ action/intention\ of\ the\ subject\ on\ the\ destination,\ area\ of\ responsability,\ etc.\}$.

Stage 2: Authorisation This refers to the interconnection point for the authorisation and access to a protected object, where an *authorisation token* has to be produced. To request this procedure, a PEP service associated with the infrastructure has to connect with the closest PDP (the primary one) to the infrastructure, providing it with the authentication token and the type of action (e.g., for reading

or control) on the destination node. In extreme situations where the primary PDP is in active, the PEP has to transparently connect to another distributed PDP (a secondary one) taking into account the use of a delegation scheme (beyond the scope of this paper).

Stage 3: Interoperability This corresponds to the state in which the PEP can transparently apply the security policies contained inside the authorisation token, and perform the action desired by the subject.

In stage 2, the two principal components that comprise the PDP are: (i) a *validation manager*, and (iii) a *Policy Decision Manager* (denoted here as PDM). The former is closely related to the validation and verification of each authentication token received from each requesting system (e.g., verify the type, size and content of the token format), and the validation of roles and permissions assigned to the requester. In contrast, the second component aims to evaluate the access, taking into account, the set of input parameters described below, and prepare the authorisation token only in the case of accepting the access. According to the IEC-62351-8 the tokens can become one of three following classes: (i) an ID certificate (profile A for TLS/SSL environments), (ii) an attribute certificate (profile B for TLS/SSL environments), or (iii) a simple token (profile C without any additional security), with a specific object identifier (OID = 1.2.840.10070.8.1) for the three profiles, plus the information assigned to the subject.

As each control infrastructure has to depend on an identity server, which is in charge not only of authenticating the subject but also of preparing the access token, the IEC-62351-8 also recommends all control systems to trust a third entity responsible for assigning roles to users and managing access tokens (e.g., the security administrator); apply the usual security tuple, username and password, in conjunction with X.509 certificates; and to configure a repository for contrasting the authentication information and retrieving the access token. For this reason, our approach depends on the simplest token based on the *A* profile, and on LDAP (Lightweight Directory Access Protocol) directory version 3 with support for SSL/TLS, where each LDAP object should be subject to the RFC-4524 [41] representing RBAC relationships according to a unique attribute; in our case, the ID of control entities.

For the token management, the IEC-62351-8 provides two implementation models: (i) the PULL model or (ii) the PUSH model. The former forces control entities to first authenticate themselves to the control objects, so that these can verify the access and get their tokens from their identity server. The PUSH model, to the contrary, consists in first fetching the access token from the identity server before accessing the control object. Although both models are well-known and they are recommended by the standard, the on-demand PULL model requires additional communication for the authentication and agreement process, which could produce serious operational delays [9], and therefore infringe one of the five control requirements identified in [6]; i.e., the operational performance. To further prevent this degradation, the approach proposed here follows the *PUSH model* but tailored to a set of architectural conditions. For example, gateways should not only serve as the main interfaces in each sub-network, but also serve as data caches, as considered by Honeywell International in [42]. In this way, the system guarantees a rapid access to the data and a reduced overhead in the destination node.

For the validation of the authentication token, the validation manager has to validate both the type and the format of the token received, the type of action (or intention – reading of measurements, active generator, etc.) on the protected object with respect to the initial access conditions granted by the identity server. To validate roles, the IEC-62351-8 offers unique identifiers to classify different types of roles (RolesID): (i) seven specific roles for power and control applications (see Table 2), (ii) 32.760 reserved for security applications within the IEC-62351, and (iii) 32.767 for private use. If all this information is correct, it is then passed on as input parameters to the PDM, which has to determine the final access according to: (i) the natural status of the context and the accessibility of the required object; (ii) the type and nature of the object to be accessed with respect to the intention of the authenticated entity; and (iii) the requirements given by the organisation policy and the permissions according to the IEC-62351-8. On the other hand, we can also observe from Figure 2 that the validation manager contains a database to verify roles and associated rights, so the continued maintenance of this database is fundamental to ensure the access from different systems at all times. Still, this design requirement entails a compromise by part of the infrastructures involved, as they have to share their hierarchical organisational structure in common points, i.e, the PDPs. To offer more restrictive upgrades without requiring a continual and complete sharing of information, we intend to extend the approach to consider this issue taking into account the technology of cloud-computing in a future work.

In order to make the computation of all of these parameters easier, an *expert system* based on simple rules is integrated inside the PDM, where each rule analyses a set of attributes related to:

- **Context**: the observation level to outline the criticality degree of the network. This value is obtained from a context manager, which is described later on.
- **Control object**: ID and type of protected object (e.g., controller, sensor or actuator), the operations assigned to such an object, the accessibility degree from its gateway, and the type of security policy (see Table 1 with the cipher suite given by IEC-62351 [12]).
- **Control subject**: ID of the requester and its intention in the control object, together with its roles and permissions.

If requesters fulfil the necessary requirements for the access, and control objects are accessible from their respective gateways, the expert system is then able to obtain the type of security policy linked to these objects and prepare the access token. The security policies could be stored in XML structures, since these provide generalised and simple formats that can help encode policies in an easily readable and processable format for machines.

3.2 Context-Awareness and Dynamic Separation of Duty

A *context manager*, also integrated inside the PDM, is responsible for reviewing the criticality degree of a network in relation to the accessibility degree of its protected

Table 2: Roles and rights belonging to IEC-62351-8

	Rights associated with IEC-62351-8 roles										
Roles	**View**	**Read**	**Dataset**	**Reporting**	**Fileread**	**Filewrite**	**Filemgnt**	**Control**	**Config**	**Settinggroup**	**Security**
Viewer[a]	✓			✓							
Operator[b]	✓	✓		✓				✓			
Engineer[c]	✓	✓	✓	✓		✓	✓		✓		
Installer[d]	✓	✓		✓		✓			✓		
SECADM[e]	✓	✓	✓			✓	✓	✓	✓	✓	✓
SECAUD[f]	✓	✓		✓	✓						
RBACMNT[g]	✓	✓					✓		✓	✓	

[a] Capacity to view data objects.
[b] Capacity to view data objects and values, and perform control.
[c] Capacity to view data objects and values, access datasets and files, and configure servers.
[d] Capacity to view data objects and values, write files and configure servers.
[e] Capacity to manage users-roles-rights, and change security setting.
[f] Capacity to audit the system by viewing audit logs.
[g] Hereditary role from the SECADM with only the ability to manage roles and rights.

objects (see Figure 2). To carry out this task, the manager first needs the collaboration of its closest gateways, receiving from them, information related to the rate of unobserved nodes that violate **OR1**. Namely, gateways have to periodically execute Algorithm 3.1 in order to verify whether or not the first observation rule has been reached (see also [32]) by each $n_d \in \mathbf{N}_D$ of their graph. Note that in real applications, this study of the context status should consist of Network and System Management (NSM) data objects as specified in the IEC-62351-7 standard [43]. Namely, NMS objects are in charge of dynamically monitoring the health of power networks and systems such as network configurations, security parameters, quality of service, or states of redundant systems. However, we simplify the application of these data objects by addressing a more theoretical-practical study based on the topological changes, constraints of structural controllability, and on the degree of accessibility to the gateway taking into account the network diameter.

Algorithm 3.1: UNOBSERVED NODES $(\mathscr{G}(V,E),\mathbf{N}_D)$

output (U)
local n_d, U, DS, N;
$U^a \leftarrow V - \mathbf{N}_D{}^b$; $DS \leftarrow \emptyset$; $N \leftarrow \emptyset$;
while $(U \neq \emptyset)$ **and** $(b \leq |\mathbf{N}_D|)$
do
- *Randomly choose a vertex* $n_d \in \mathbf{N}_D$;
- **if** $(n_d \notin (DS \cup N))$
  - **then**
    - $DS \leftarrow DS \cup \{n_d\}$;
    - **for each** $v \in V$
      - **do**
        - **if** $(n_d, v) \in E$
          - **then**
            - $N \leftarrow N \cup \{v\}$;
            - $U \leftarrow U \setminus \{v\}$;
    - $U \leftarrow U \setminus \{n_d\}$;

return (U)

[a] U is the set of unobserved nodes.
[b] Remember that the gateway is part of the set of driver nodes.

More specifically, two *criticality thresholds* related to the observation degree of the network are established so as to limit access in extreme situations and reduce the communication overhead as much as possible. This access constraint is closely linked to the security policies and requirements of the organisation/s (e.g., two or more SCADA systems) implicated in the interaction. For example, we assume in this paper, that only authorised personnel with the capacity to lead '*Control*' (action reserved for operator and SECADM in Table 2) are able to enter the affected network and take over the situation and/or restore variables, states, connectivities or parameters. In contrast, requests recognised by the system but with roles $\notin \{$operator, SECADM$\}$ such as viewer, installer, engineer, etc. should not be accepted to avoid communication overhead or bottlenecks. This means that the PDP (and its integrated PDM) not only works as an authentic '*authorisation firewall*' but also as a protector of critical environments. The two aforementioned thresholds are as follows:

- **Criticality of the context** (**CCont**): this states the key point at which the network requires the specific protection of the closest PDP and the activation of one of the essential features of RBAC, which is associated with the *Dynamic Separation of Duty* (DSD). This is a security property that aims to assign multiple mutually exclusive roles (e.g., either engineer or operator) to an entity, and can employ them in the same session independently but not at the same time or simultaneously [44]. This is the reason why, in Section 3.1, we define two types of roles: $\text{roleID}_{primary}$ and $\{\text{roleID}_1, \ldots, \text{roleID}_n\}_{secoundary}$. The primary role is the role by default active in the session of the subject, and the secondary roles are those ones that can be activated for DSD.

  To guarantee the effectiveness of the DSD in critical situations and prevent a communication overhead in the network by avoiding minor priority access when the network is suffering a significant degradation in the control, two further criticality thresholds are declared within the **CCont**:

- **MaxCCont**: the limit point where the DSD should be activated in the PDP, and therefore the point at which the PDP has to start to filter the access.
- **MinCCont**: the critical/extreme point where it is necessary to restore the entire network such that **MinCCont** $<$ **MaxCCont** $<$ 100.0, where 100.0 denotes the best situation in which there is no risk. For example, **MinCCont** could mean that the control has been completely lost, or more than 90-95% nodes have disconnected from the gateway.

Therefore, **MinCCont** $\leq$ **CCont** $\leq$ **MaxCCont** and their threshold values should be agreed by all organisations involved in the interconnection, and declared under a common security policy. Nonetheless, we believe that their values should also depend on the characteristics of the application context, the natural conditions of the network (e.g., quality of service, constraints of nodes, access requirements, etc.) and the type of heterogeneity of the interconnection. Moreover, these two thresholds could even be beneficial to those gateways with integrated alarm managers. They could, for example, supervise, in local, the changes made by **CCont**, and warn the corresponding central system of specific situations through a dedicated *alarm manager*. This capacity can also be extended in each PDM (see Figure 2), whose alarms could be defined depending on a range of alerts [**MaxCCont** ... **MinCCont**].

- **Criticality of the object** (**CObj**): this threshold defines when to start analyzing the real reach of a determined object, irrespective of the natural conditions of the network (the **CCont**). In this way, it is possible to ensure the access and control of an object even in crisis situations, but restricted to the type of role or priority permitted (described below).

  To compute **CObj**, the context manager has to receive further information from the gateway of the network affected, the value of which is computed when the **MaxCCont** has been overtaken. In this case, we take into account, on the one hand, the network diameter from the gateway to the control object to verify connectivity. The diameter is calculated in our simulations using the traditional *breadth-first search* method for unweighted digraphs with a complexity order of $O(n+e) = O(n)$. On the other hand, we also quantify (to percentage level) the number of reachable paths from the gateway to the destination node, considering in this case all those connected neighbours of the gateway that reach such an object.

This way of defining criticality thresholds associated with **CCont** and **CObj** can also help identify when to apply a subset of *priority thresholds* so as to filter actions taking into account the rights defined for the IEC-62351-8 roles in Table 2. Namely, one easy way to verify whether the control object can execute a specific operation at a given moment on a specific context, and allow the authorised access according to the operations contained in Table 2, would be through $\mathbf{prior}X_{CCont}$ and $\mathbf{prior}X_{CObj}$, both $\in$ $[0.0-100.0]$. Depending on the restrictions of each organisation, the action to execute and the criticality of the context, these thresholds can vary so as to constraint the access in the field and reduce overhead in the destination network. For example, we determine

that for the experiments developed in next section that the priority to view critical data (**priorView**) in crisis situations should be more restrictive than the priority assigned to the **priorConfig**, and be much more restrictive than **priorControl**; i.e., **priorView** $<$ **priorConfig** $<$ **priorControl**.

3.3 A Simple Rule-based Engine for Authorisation

Regarding the rule-based engine, it basically consists of verifying the fulfilment of all these values and asserting the corresponding security policy. The construction of these rules follows the structure <**rule**> **:=** <**condition**>⇒<**action**> such that <condition> contains the predicates associated with <subject><object>, i.e.:

Generic Rule (

<**subject**> := <*subjectID*><*infrID*$_s$ ><*roleID*$_{primary}$ ><*rolesID*$_{secondary}$ ><*rights*>
<*intent*><*objectID*$_s$ ><*netID*$_s$ >
such that:
<infrID$_s$ ><subjectID><objectID$_s$ ><netID$_s$ > := <value>
<roleID$_{primary}$ > := Viewer |[1] Installer | ... | SECAUD (cf. Table 2), such that
<roleID$_{primary}$ > $\neq$ " " (no empty)
<rolesID$_{secondary}$ > : = Viewer | Installer | ... | SECAUD
<intent>, <rights> := View | Read | Dataset | ... | Security (cf. Table 2)

<***object***> *:=* <*objectID*$_o$ ><*objType*><*operations*><*gwID*><*criticality*><*context*>
<*infrID_List*><*netID*$_o$ ><*accessPolicy*>
such that:
<objectID$_o$ ><gwID><netID$_o$ > := <value>
<objectID$_o$ > := <objectID$_s$ > = <objectID$_o$ >
<infrID_List> := {infrID$_1$, infrID$_2$, ..., infrID$_n$} & <infrID$_s$ > $\in$ <infrID_List>
<netID$_o$ > := <netID$_s$ > = <netID$_o$ >
<objType> := **N**$_D$ | actuator | sensor
<operations> := View | Read | Dataset | ... | Security (cf. Table 2)
<criticality> := **CObj** $\geq$ **prior**X_{CObj}
<context> := **CCont** $\geq$ **prior**X_{CCont}
<accessPolicy> := TLS_RSA_WITH_RCA_128_SHA | ... (cf. Table 1)

⇒
<**action**> := <*subjectID*><*objectID*><*idGw*><*accessPolicy*><*result*><*DSD*>
such that:
<subjectID><objectID><gwID> := <value>
<accessPolicy> := TLS_RSA_WITH_RCA_128_SHA | ... (cf. Table 1)
<result> := FALSE (by default) | TRUE
<DSD> : = FALSE (by default) | TRUE

)

From this rule definition, <subject> states the characteristics of the control subject, <object> the properties of the control object and its context, and <action> the security

[1] | - OR, & - AND.

policies (see Table 3). Given that <subject> and <object> are quite dependent on the roles and rights specified in IEC-62351-8, Table 4 tries to particularise in the most relevant values of their predicates. Concretely, 14 rules related to permissions and a set of 8 exception rules for unauthorised access and termination of the engine have been defined for our study; all of them sorted according to their salience.

In other words, we assume that the *Control* rule holds a greater salience with respect to the rules associated with *Database, View* or *Exceptions*. But even so, this assumption must be in relation to the security policies of the organisations involved in the interoperability action; a condition that also occurs when defining those particular scenarios in charge of facilitating or disrupting the access in extreme situations. For this reason, we also assume the existence of several particular cases linked to specific actions associated with *Control, Reporting* and *Read/View*, in which only authorised staff with specific roles can take control of a determined situation. A particular scenario is when the context suffers a strong topological change that affects its observation degree such that **MinCCont** $\leq$ **CCont** $\leq$ **MaxCCont**, and the DSD has to be activated. Its rule would be the same as that specified above but with the difference that the output action must include the condition <DSD> : = TRUE.

Similarly, a set of operations to be executed by control objects also has been predefined[2]. For example, objects that $\in \mathbf{N}_D$ are able to address commands of *control, database, filemgnt, view, read, security, fileread, filewrite, config, settinggroup* or *reporting*, whereas objects that $\notin \mathbf{N}_D$, such as sensors and actuators, can address specific operations according to the class of the object. In the case of sensors, the operations are related to *view, read, security, config, settinggroup* and *reporting*, whereas in the case of actuators, they are assigned with *control, security, config* and *settinggroup*.

Table 3: Properties related to subject, object and action

Property	Definition
subjectID, objectID	ID of the subject and the object, respectively
$\text{roleID}_{primary}$, $\text{roleID}_{secondary}$	Primary role assigned to a subject and its secondary roles granted
rights	The permissions associated with the roles
infrID_s	ID of the infrastructure in which the subjects belong
infrID_List	A list of IDs related to those infrastructures in which the object can interact
intention	The action to be executed by a control object
netID_s	ID of that network where the subject wants to execute an action
netID_o, gwID	ID of the network where the object is deployed, and the ID of its gateway
objType	Type of object in which an action has to be executed (e.g., controller, sensor, actuator)
operations	Operations that can carry out a control object
criticality, context	Indicators that state **CObj** and **CCont**
accessPolicy	Security police related to a control object
DSD	Indicator for dynamic of separation of duty, and its activation

Taking Table 4 into account, the correctness proof of the interoperability problem is solved when the following requirements are satisfied:

[2]The configurations (software agents, thresholds and operations) given here are completely fictitious. For real applications, these should be subject to the requirements and security policies defined by the organisation/s.

Table 4: Rules and the minimum predicates, sorted by salience

$\forall$ *the rules*	common	Subject	<subjectID> := $ID_{subject}$
		Object	<context> := CCont $\geq$ priorX_{CCont}; <criticality> := CObj $\geq$ prioryX_{CObj} <infrID_List> := <infrID$_s$ > $\in$ <infrID_List> <netID$_o$ > := <netID$_s$ > = <netID$_o$ >
		Action	<result> := TRUE (**termination case**); <DSD> := FALSE (by default) <objType> := $\mathbf{N}_D$ | actuator | sensor <accessPolicy> := TLS_RSA_WITH_RCA_128_SHA | ... (cf. Table 1))
		Otherwise	The breach of all the rules entails go to exceptions
Control	common	Subject	<intent> := "Control"; <rights> := "Control" & <others>
		Object	<objType> := $\mathbf{N}_D$ | actuator
	normal	Subject	<roleID$_{primary}$ > := 'SECADM' | 'Operator' (see Table 2) <rolesID$_{secondary}$ > := Viewer | Installer | ... | SECAUD
		Object	<context> := **MaxCCont** < **CCont** $\leq$ **100.0**
	critical - **DSD**	Subject	<roleID$_{primary}$ > := <roleID$_{primary}$ > $\neq$ " " & $\in$ [Viewer | Installer | ...] <rolesID$_{secondary}$ > : = ("SECADM" | "Operator") & <rolesID$_{secondary}$ >
		Object	<context> := **MinCCont** $\leq$ **CCont** $\leq$ **MaxCCont** <criticality> := CObj $\geq$ prioryControl$^*_{CObj}$
		Action	<result> := TRUE; <DSD> := **TRUE**
Security		Subject	<roleID$_{primary}$ > := 'SECADM' <intent> := "Security"; <rights> := "Security" & <others>
		Object	<criticality> := CObj $\geq$ priorSecurity
Reporting	common	Subject	<intent> := "Reporting"; <rights> := "Reporting" & <others>
		Object	<objType> := $\mathbf{N}_D$ | sensor
	normal	Subject	<roleID$_{primary}$ > := 'Engineer' | 'Operator' | 'Installer' | 'Viewer' | 'SECAUD'
	critical	Subject	<roleID$_{primary}$ > := 'Engineer' | 'Operator' | 'Installer'
		Object	<context> := CCont $\geq$ prioryReporting$^*_{CCont}$ <criticality> := CObj $\geq$ prioryReporing$^*_{CObj}$
Read/View	common	Subject	<intent> := "Read" | "View"; <rights> := ("Read" | "View") & <others>
		Object	<objType> := $\mathbf{N}_D$ | sensor
	normal	Subject	<roleID$_{primary}$ > := 'Engineer' | 'Operator' | 'Engineer' | 'Installer' | 'SECAUD' | 'SECADM' | 'RBACMNT'
		Object	<context> := CCont $\geq$ prioryRead/View$_{CCont}$ <criticality> := CObj $\geq$ prioryRead/View$_{CObj}$
	critical	Subject	<roleID$_{primary}$ > := 'Operator' | 'SECADM'
		Object	<context> := CCont $\geq$ prioryRead/View$^*_{CCont}$ <criticality> := CObj $\geq$ prioryRead/View$^*_{CObj}$
Settinggroup		Subject	<roleID$_{primary}$ > := 'SECADM' | 'RBACMNT' <intent> := "Settinggroup"; <rights> := "Settinggroup" & <others>
Config		Subject	<roleID$_{primary}$ > := 'Engineer' | 'Installer' | 'SECADM' | 'RBACMNT' <intent> := "Config"; <rights> := "Config" & <others>
Database		Subject	<roleID$_{primary}$ > := 'Engineer' | 'SECADM' <intent> := "Database"; <rights> := "Database" & <others>
		Object	<objType> := $\mathbf{N}_D$
Filewrite		Subject	<roleID$_{primary}$ > := 'Engineer' | 'Installer' | 'SECADM' <intent> := "Filewrite"; <rights> := "Filewrite" & <others>
		Object	<objType> := $\mathbf{N}_D$
Fileread		Subject	<roleID$_{primary}$ > := 'SECAUD' <intent> := "Fileread"; <rights> := "Fileread" & <others>
		Object	<objType> := $\mathbf{N}_D$
Filemngt		Subject	<roleID$_{primary}$ > := 'Engineer' | 'SECADM' | 'RBACMNT' <intent> := "Filemngt"; <rights> := "Filemngt" & <others>
		Object	<objType> := $\mathbf{N}_D$
View		Subject	<roleID$_{primary}$ > := 'Engineer' | 'Operator' | 'Engineer' | 'Installer' | 'SECAUD' | 'SECADM' | 'RBACMNT' | 'Viewer' <intent> := "View"; <rights> := "View" & <others>
Unauthorised access	Exceptions	Subject	<roleID$_{primary}$ > = " " | $\notin$ [Viewer | Installer | ...] (cf. Table 2)
		Object	<context> := (CObj < prioryX_{CCont}) | (0.0 $\leq$ CCont $\leq$ MinCCont) <criticality> := CObj < prioryX_{CObj} <infrID_List> := <infrID$_s$ > $\notin$ <infrID_List> <netID$_o$ > := <netID$_s$ > $\neq$ <netID$_o$ >
		End-case	There is not a specific constraint $\forall$ rules such that it is subsumed in the rest of rules, containing <subject> := <subjectID> as one sole predicate.
		Action	<accessPolicy> := NIL & <result> := FALSE (**termination case**)

Authorisation The engine that decides is able to determine the access degree according to the restrictions of the subject and of the object, as well as the criticality level of the context.

Termination The engine is able to finish the decision process in a finite state.

Validity The engine is able to terminate and provide a determined action linked to a security policy (either NIL or a recognised policy by the organisation).

For the former requirement, we follow the technique of subsumed rules given by Nguyen *et al.* in [45]. The technique consists in finding two or more rules which have the same conclusions, but one holds extra constraints in a determined situation, in which it will succeed [46]. If we observe the generic rule given above and the rules specified in Table 4, we can observe that all rules: (i) share the output action (e.g., either <result> := TRUE in successful cases or <result> := FALSE in cases of error); (ii) contain at least one different constraint depending on the kind of right/action to execute in the field; and (iii) all of them are sorted according to their salience. Namely, most of the rules follow the structure $P(x)$ & $Q(x)$ & $R(x) \longrightarrow S(x)$ with high salience and $P(z)$ & $Q(z)$ & $R(z) \longrightarrow S(z)$ with less salience such that $R(x) \neq R(z)$. The engine verifies each structured rule until reaching the exception scenario in which $P(y) \longrightarrow S(y)$ corresponding to the 'End-case' defined in Table 4, which can be interpreted as '*if the engine does not know* $Q(x)$, $R(x)$, $Q(z)$ *and* $R(z)$, *do* $S(y)$ *anyway*'. This way of characterising rules means that the knowledge domain becomes unique, guaranteeing a greater certainty during the authorisation.

With respect to the termination, the engine always ensures an output action since it is previously initialised to facts with values by default, such as <result> := FALSE; and this action can be successful or not, depending on the attributes assigned to <subject> <object>. Namely, if $\exists$ a rule related to the rights/actions of a determined object and it is in relation to the intention and the roles assigned to the subject, and the criticality level is conditioned to the access degree associated with such rights/actions, then $\exists$ a successful output condition such that <result> := TRUE and <accessPolicy> contains the security policy related to the destination. In contrast, if no rule associated with rights/actions is satisfied in a time t (restrictive to 'End-case'), then $\exists$ always an exception which makes the engine assert a security policy with value NIL and <result> := FALSE. This also indicates that the latter requirement is also satisfied since the engine finishes and delivers a determined value of security policy.

To make good use of all this knowledge, a case study is presented in the next section. In this way, it is possible to show the functionality of the approach and its practical validity for general cases (for small, medium and large control networks).

4 Case Study: Power Dominance in Heterogeneous Control Domains

The network architecture proposed in Section 2 and the policy enforcement approach presented in Section 3 have been implemented in Matlab and Java, respectively. The

first part contains the implementation of Algorithm 2.1 with the two power-law distributions (PLOD with $\alpha = 0.1, 0.2$ and BA with $\alpha = 3$). This part also achieves the power domination by performing **OR1** and **OR2**. The result is a compendium of subgraphs $\mathscr{G}_i(V,E)$ holding an implicit hierarchy with respect to their gateways, and a sub-set of driver nodes $\mathbf{N}_D$ in each subgraph, each one responsible for its own monitoring. The second part, to the contrary, includes the construction components for the PDP following the recommendations given by the IEC-62351-8.

4.1 Experimentation Design for Validation

As this standard recommends configuring an LDAPv3 server for the access token management, the Apache Directory Studio™ [47] has been configured as it is compatible with the ApacheDS and codified in Java. For simplicity's sake, we apply the same repository to all the networks and for all the control subjects, instead of instating several repositories depending on the number of infrastructures. For the insertion of entries in LDAP, the approach follows the RFC-2798 [48] under the attribute *inetOrgPerson:userCertificate* so as to store the encoded X.509 certificates together with relevant information associated with granted roles and rights. The rule-based engine of the PDM is written in JESS (Java™ Expert System Shell) which is able to process the conventional language CLISP (the Common LISP) in Java [49].

Based on this implementation, a case study has been developed to explore the behaviour of the approach when underlying networks are being threatened by unexpected topological perturbations. The approaches proposed in [32, 33] have been extended here not only to consider threats to the availability of nodes and links (*isolation* of control nodes (removing of all edges) and the *arbitrary removal* of a few edges), but also to exploit other types of threats related to the deliberate *injection of control links* (i.e., insert new links). Note that many of these disturbances do not necessarily come from faults occasioned within communication networks. They may also come from the frequent variations of the underlying physical infrastructure (e.g., overload of (renewable) power sources), which may cause secondary effects toward the control network; e.g., isolation of nodes by lacking energy supply. Nonetheless, and regardless of the origin of these disturbances in the control network, the purpose is now to thoroughly evaluate the degree of access to unstable monitoring environments, subject to diverse types of logical threats with more than $|V|/2$ random targets.

The simulations are specifically based on the interconnection of three power-law networks presenting different scales with 100-500 nodes (small networks), 500-1000 (medium networks), and $\geq$ 1000 nodes (large networks). Based on these distributions, we determine that each of these three networks are associated with three independent infrastructures (infrID - netID) such that (infrID_1 - netD_1), (infrID_2 - netD_2), and (infrID_3 - netD_3), and for each of these networks access from more than one infrastructure can exist. For the simulations, we consider that subjects belonging to infrID_1 can interact with netD_1, netD_2 and netD_3; subjects belonging to infrID_2 can interact with netD_1 and netD_2; and infrID_3 with netD_1 and netD_3. As regards the criticality thresholds, we have declared **MaxCCont** for a value of 0.85, whereas the **MinCCont** has been defined for a value of 0.25.

In order to tinker with different control objects, we first obtain the controller nodes

(or driver nodes $\in \mathbf{N}_D$) using **OR1** and **OR2** defined in Algorithm 2.1 – '*Embedding Control*'. From the rest of nodes (the ones being observed), we randomly extract a subset of nodes with the role of sensor in charge of perceiving an environment and sending the information to its neighbouring controllers, and a subset of nodes with the role of actuator to allow control to be injected into the field. This categorisation of nodes is based on the real construction of monitoring substations, which are based on the interconnection of several RTUs (the driver nodes) in charge of (i) collecting evidence from sensors to be sent to the central system later, and/or (ii) forwarding control signals from the central system to actuators to be executed in end-field devices.

Table 5: Roles and rights belonging to IEC-62351-8

	Control entities attempting to access restricted networks											
	Small-Medium Net.						**Medium-Large Net.**					
Access	**E1**	**E2**	**E3**	**E4**	**E5**	**E?**	**E1**	**E2**	**E3**	**E4**	**E5**	**E?**
	Network 1 - PLOD α=0.1 - 1000 Nodes						**Network 1 - PLOD α=0.1 - 2000 Nodes**					
Total	55	43	57	32	41	46	11	7	14	8	9	8
Normal	27.77	25.58	40.35	6.25	26.82	0.0	81.81	57.14	92.85	0.0	66.66	0.0
Denied	72.72	74.41	59.64	93.75	73.17	100.0	18.18	42.85	7.14	100.0	33.33	100.0
DSD	0.0	0.0	0.0	100.0	9.09	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	Network 2 - BA α=3 - 500 Nodes						**Network 2 - BA α=3 - 1500 Nodes**					
Total	41	39	39	40	35	43	6	14	10	8	9	12
Normal	29.26	23.07	33.33	0.0	20.0	0.0	100.0	14.28	60.0	0.0	44.44	0.0
Denied	70.73	76.92	66.66	100.0	80.0	100.0	0.0	85.71	40.0	100.0	55.55	100.0
DSD	0.0	0.0	0.0	0.0	42.85	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	Network 3 - PLOD α=0.2 - 100 Nodes						**Network 3 - PLOD α= 0.2 - 1000 Nodes**					
Total	41	42	41	40	44	45	9	14	7	10	8	10
Normal	21.95	0.0	9.75	2.50	0.0	0.0	100.0	0.0	100.0	0.0	0.0	0.0
Denied	78.04	100.0	90.24	97.50	100.0	100.0	0.0	100.0	0.0	100.0	100.0	100.0
DSD	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Table 5 summarises the accessibility degree of six control subjects, considering in this case the total access, the rate of accepted and denied access, and the percentage of access in DSD mode (a sub-part of the rate of accepted access). The profiles correspond to six software agents, written in Java, capable of dynamically and randomly producing access requests in the different network distributions. Their profiles are as follows:

E1 - profile: $\text{roleID}_{primary}$:= SECADM; $\text{roleID}_{secoundary}$:= " "; infrID := 1 with access to netD_1, netD_2 and netD_3; intention := Control (it refers to acting on controllers or actuators); $\text{priorControl}_{CObj} := \geq 0.10$; and $\text{priorControl}_{CCont} := \geq$ MinCCont. Although the normal procedure is to define a SECADM for each infrastructure, here we define a unique SECADM belonging to infrID_1 for simulations, making it possible to observe the behaviour of several profiles. As for the attribute **priorControl**, it means that SECADM can attend to any situation at any time with the necessary leadership to carry out control tasks (e.g., execute commands, obtain measurements from $\mathbf{N}_D$, etc.). But if the criticality of the context exceeds the value of **MinCCont**, the system, in this case, only needs to recover the status rather than permitting the access (an extremely-critical situation).

E2 - profile: $\text{roleID}_{primary}$:= SECAUD; $\text{roleID}_{secoundary}$:= " "; infrID := 2 with access to netD_1 and netD_2; intention := Read (it aims to read evidences or data objects from a final device); $\text{priorRead}_{CObj} := \geq 0.60$; entities of this category can read logs and files until the current network situation reaches the value of 0.60; and $\text{priorRead}_{CCont} := \geq$ MinCCont.

E3 - profile: roleID$_{primary}$:= Operator; roleID$_{secoundary}$:= " "; infrID := 1 with access to netD$_1$, netD$_2$ and netD$_3$; intention := Control; priorControl$_{CObj}$:= $\geq$ 0.10, with a similar behaviour to **E1**: and priorControl$_{CCont}$:= $\geq$ MinCCont.

E4 - profile: roleID$_{primary}$:= Engineer; roleID$_{secoundary}$:= Operator; infrID := 3 with access to netD$_1$ and netD$_3$; intention := Report (it deals with reporting evidences or data objects from a final device); and priorReport$_{CObj}$:= $\geq$ 0.30; and priorReport$_{CCont}$:= $\geq$ MinCCont.

E5 - profile: roleID$_{primary}$:= Installer; roleID$_{secoundary}$:= Engineer & Operator; infrID := 2 with access to netD$_1$ and netD$_2$; intention := Config (it aims to config servers or controllers); priorConfig$_{CObj}$:= $\geq$ 0.10; and priorConfig$_{CContj}$:= $\geq$ MinCCont.

E?: an unknown entity with the intention of *control* in netD$_1$, netD$_2$ and netD$_3$.

The simulations have been executed for the interconnection of three networks (netD$_1$, netD$_2$ and netD$_3$) for 20 minutes, the configurations of which are explicitly stated in Table 5. Concretely, Table 5 contains the access level of the agents inside such contexts and its results are aligned with the results depicted in Figure 3. This figure illustrates how the context of each network is updated over time and according to the rate of the unobserved nodes. Figure 3 also shows the robustness degree of the PLOD distributions against combined attacks, whereas BA distributions seem to be less resilient with a loss in the control and supervision of almost 70%.

4.2 Discussions and Complexities

The configurations described earlier are essential to help the expert system correlate the intention of a subject with the functional features of a control object, and thus to help it in accepting or denying actions in the field. This capacity is also represented in Table 5 and Figure 3, where authorised entities with the intention of Control (**E1**, **E3**) are able to take control of the networks in most cases but limited to certain objects (N_D and actuators). This also means that the subjects' intentions should be subjected to the type of object and its capacities. For example, the SECAUD (**E2**) wishes to 'Read' data objects from a destination, but these data objects should belong to driver nodes or sensors with the ability to read or report evidences (similar for **E4**); and **E5** with the intention of 'Configuring' servers or controllers (both $\in N_D$). Nonetheless, and again, these assignations are closely related to the type of context, heterogeneity of resources and the security policies defined by the organisations.

From Figure 3, it is also possible to observe how practically all the distributions exceed the **MaxCCont**, except for netD$_1$ and netD$_3$ belonging to the second experiment (related to the interconnection of medium and large networks). This means that DSD can dynamically be activated at any time for **E4** or **E5** by being assigned the sub-role of Operator. However, **E4** can only attend to networks netD$_1$ and netD$_3$ in extreme situations, whereas **E5** can only assist the networks netD$_1$ and netD$_2$ as specified in Table 5. Indeed, **E4** is able to access with a total of 100% in DSD mode in the small-medium networks with identifiers netD$_1$ and netD$_3$, and **E5** gets in with 9.09% in netD$_1$ and

with 42.85% in $netD_2$ (in smal-medium contexts). In contrast, neither **E4** nor **E5** are able to enter in DSD mode in the simulation carried out for medium-large networks because: (i) $netD_1$ and $netD_3$ do not reach the threshold **MaxCCont**, and (ii) $netD_2$ is slightly affected, so the access request by the part of **E5** in the control elements (such as driver nodes and actuators) is performed from a completely randomised viewpoint. Therefore, the randomness of how access is requested by subjects and the type of element for the access (controller, sensor or actuator), plays a fundamental role within this study. For example, **E1** (the SECADM) has great problems entering normal mode in small-medium networks, probably caused by the number of unobserved nodes (e.g., isolations or disconnections) or the type of node demanded (e.g., trying to access a sensor to execute an active action – control – in the field). Note that this also happens to the rest of the software agents. Lastly, unknown requests (e.g., **E?**) are refused in all cases, and irrespective of the dimension of the network or its natural conditions, something that again highlights the potential features of the expert system for protection.

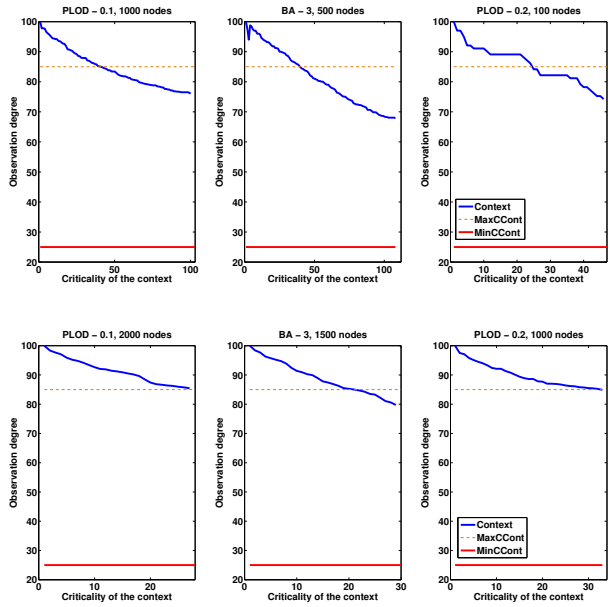

Figure 3: Context-awareness

As for complexities, it is important to consider here two important phases. On the one hand, the commissioning phase in which ('virtual') control networks have to be generated, and on the other hand, the interoperability phase in which the PDP components have to be activated. For the former phase, we consider the computational cost required in processing Algorithm 2.1. As the generation of networks can be arbitrary depending on the generation algorithm, the cost of computing '*Generate Network*' becomes dependent on the complexity associated with the selected algorithm. But this

charge becomes independent from the cost involved in evaluating the acyclicity degree, establishing connectivity with the gateway and embedding control. The first two may demand an additional cost of $O(kn)$ in the worst scenario, whereas the latter can entail an extra cost of $O(n^2)$ [34] because **OR1** $\subseteq$ **OR2** [32] and the selection of the gateway follows a completely lineal process.

Regarding the interoperability phase, it is appreciable that the greater cost primarily falls on PDP and gateways because they contain all the decision-making processes associated with the interoperability action. This fact does not become worrying because we had already assumed (cf. Section 2) that are devices with sufficient capacities to address not only these tasks, but also to frequently calculate the accessibility through the diameter ($O(n)$) and Algorithm 3.1. As described in Section 3.2, this algorithm aims to obtain those nodes that are not being observed by at least one member of the set of driver nodes such that the overhead can become $O(n^2)$ if $|\mathbf{N}_D| \sim |V|$. This complexity also becomes notable in PDM, as its authorisation engine is based on an expert system composed of 14 straightforward rules and 8 simple exception rules (see Table 4), whose processing not only terminates in a finite time t (cf. Section 3.3) but also ensures a fast response due to the inherent features of expert systems [50]. Any action executed after PDP and gateways is outside the scope of the interoperability, and it is part of the general automation of a SCADA system. Moreover, the choice of the PUSH model for the connectivity and the purposes of monitoring the context through a context manager and gateways (see Figure 2) have also become key in releasing communication channels from unauthorised accesses, thereby reducing communication overhead.

Finally it is left to say that our approach differs from [15, 16, 18, 42, 17, 19] in several aspects. Firstly, the majority of existing approaches [16, 42, 17, 19] base their interoperability level on the decisions of each control center without looking at, for example, the accessibility level of a determined demanding context/region as stated in [9]. Secondly, many of these approaches build their role constraints according to a set of authorisation principles (area of responsibility, type of operation (generally associated with the communication protocol [15] such as DNP3, Modbus/TCP, etc.), type of organisational hierarchy, ...) but discard the possibility of following specific security standards, which can help ensure a better interoperability and sustainability of the system. Likewise, the automation during the authorisation process is not always guaranteed where the policy decision points generally depend on each organisation [16], instead of being shared by a group of organisations (e.g., SCADA 1 - SCADA 2 - Providers [18]). This criterion is also sustained by U. Lang *et al.* in [51] and NIST 7628 [52]. Both state the need to enforce automated authorisation mechanisms to protect any information flow within and between interconnected Smart Grid information systems, and in relation to the applicable security policies.

5 Conclusions and Future Work

Connectivity of heterogeneous networks belonging to Smart Grid environments with connections coming from anywhere, at any time and in anyway, involves the provision of specialised policy enforcement mechanisms that transparently help protect the monitoring elements from unauthenticated and/or unauthorised entities. For this reason, a

policy enforcement system based on the context and driven by the least privilege defined by IEC-62351-8 through RBAC has been proposed in this paper, where the core of the approach is constituted by a simple rule-based expert system.

In this approach, graph theory has been the central topic for the deployment of large power-law distributions with similar structures to control networks of electrical systems, and for the representation of the domination through structural controllability. The selection of the elements embedding the power domination follows two fundamental observation rules within the domination, where one of them describes the observation degree related to the control level of an application context. For the interconnection of sub-graphs (sub-networks), a decentralised network architecture based on the concept of the supernode with connection to nodes working as gateways has been proposed here. This new architecture has involved a change in the construction of the power-law distributions and the power domination so as to establish a dependence on a single point in each network. Through this network construction, the approach has been validated so as to show the functionalities of RBAC both for normal and extreme scenarios, in which the networks have been perturbed with different types of threats to the availability. The simulations have highlighted that unknown users are unable to connect to control objects; authorised entities are under the conditions of the least privilege; and only authorised entities with certain roles and rights are able to attend to a determined situation.

Apart from the future work mentioned throughout this paper, we are also interested in expanding the approach with protection methods to help RBAC schemes and its roles be in active at all times. These methods are principally associated with early detection and self-healing to prevent the control system against unforeseen faults. Note that many of these faults do not necessarily have to come from the control network. They can also come from within the power network whose effect may be transferred to the control network due to their implicit interdependencies.

Acknowledgments

The research led by C. Alcaraz has received funding from the Marie-Curie COFUND programme U-Mobility, co-financed by the University of Málaga, the EC FP7 under GA No. 246550 and the Spanish Ministerio de Economía y Competitividad (COFUND2013-40259). Nonetheless, this work has also been partially supported by the research projects PISCIS (P10-TIC-06334) and PERSIST (TIN2013-41739-R).

References

[1] X. Fang, S. Misra, G. Xue, D. Yang, Smart grid - the new and improved power grid: a survey, IEEE on Communications Surveys & Tutorials, 14(4) (2012), pp. 944-980.

[2] W. Wang, X. Yi Xu, K. Mohit, A survey on the communication architectures in smart grid, Computer Networks, 55(15) (2011), pp. 3604-3629.

[3] NIST, NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0, NIST Special Publication 1108R2, 2012 http://www.nist.gov/smartgrid/upload/NIST_Framework_Release_2-0_corr.pdf, retrieved April 2015.

[4] R. Kyusakov, J. Eliasson, J. van Deventer, J. Delsing, R. Cragie, Emerging energy management standards and technologies - Challenges and application prospects, IEEE 17th Conference on Emerging Technologies & Factory Automation (2012), pp. 1-8.

[5] V. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, G. Hancke, Smart grid technologies: communication technologies and standards, IEEE Transactions on Industrial informatics, 7(4) (2011), pp. 529-538.

[6] C. Alcaraz, and J. Lopez, Analysis of requirements for critical control systems, International Journal of Critical Infrastructure Protection, 2(3-4) (2012), pp. 137-145.

[7] H. Farhangi, The path of the smart grid, IEEE on Power and Energy Magazine, 8(1) (2010), pp. 18-28.

[8] National Cybersecurity and Communications Integration Center (NCCIC), Internet accessible control systems at risk, ICS-CERT (2014), https://ics-cert.us-cert.gov/sites/default/files/Monitors/ICS-CERT_Monitor_\%20Jan-April2014.pdf, retrieved April 2015.

[9] V. Kapsalis, L. Hadellis, D. Karelis, S. Koubias, A dynamic context-aware access control architecture for e-services, Computers & Security, 25(7) (2006), pp. 507-521.

[10] IEC-62351-8, Power systems management and associated information exchange - Data and communications security - Part 8: Role-based access control, International Electrotechnical Commission, 2011, http://www.iec.ch/smartgrid/standards/, retrieved April 2015.

[11] International Electrotechnical Comision, IEC- Smart Grid, http://www.iec.ch/smartgrid/standards/, retrieved April 2015.

[12] IEC-62351 Parts 1-8: Information Security for Power System Control Operations, International Electrotechnical Commission, 2007-2011, http://www.iec.ch/smartgrid/standards/, retrieved April 2015.

[13] IEC-61850, Power Utility Automation - Parts 1-9, http://www.iec.ch/smartgrid/standards/, 2003-2005, retrieved April 2015.

[14] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, H. Zhu, Securing smart grid: cyber attacks, countermeasures, and challenges, IEEE Communications Magazine, 50(8) (2012), pp. 38-45.

[15] M. Majdalawieh, F. Parisi-Presicce, R. Sandhu, RBAC Model for SCADA, Innovative Algorithms and Techniques in Automation, Industrial Electronics and Telecommunications, Springer Netherlands (2007), pp. 329-335.

[16] H. Cheung, A. Hamlyn, T. Mander, Y. Cungang, Role-based model security access control for smart power-grids computer networks, IEEE on Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century (2008), pp. 1-7.

[17] D. Rosic, U. Novak, S. Vukmirovic, Role-based access control model supporting regional division in Smart Grid system, in: Fifth International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN) (2013), pp. 197-201.

[18] N. Kuntze, C. Rudolph, I. Bente, J. Vieweg, J. V. Helden, Interoperable device identification in Smart-Grid environments, Power and Energy Society General Meeting, IEEE (2011), pp. 1-7.

[19] A. Veichtlbauer, D. Engel, J. Ressel, F. Knirsch, O. Langthaler, F. Moser, Advanced metering and data access infrastructures in Smart Grid environments, The Seventh International Conference on Sensor Technologies and Applications (SENSORCOMM) (2013), pp. 63-68.

[20] C. Lin. Structural controllability, IEEE Transactions on Automatic Control, 19(3) (1974) pp. 201208.

[21] T. Haynes, S. Hedsetniemi, S. Hedetniemi, M. Henning, Domination in graphs applied to electric power networks, SIAM Journal on Discrete Mathematics, 15(4) (2002), pp. 519-529.

[22] H. Samuel, W. Zhuang, and B. Preiss, Improving the dominating-set routing over delay-tolerant mobile ad-hoc networks via estimating node intermeeting times, EURASIP Journal on Wireless Communications and Networking, Hindawi Publishing Corporation, 2011 (2011), pp. 1–12.

[23] G. Pagani, M. Aiello, The power grid as a complex network: a survey, Physica A: Statistical Mechanics and its Applications, 392(11) (2013), pp. 2688-2700.

[24] C. Palmer, J. Steffan. Generating network topologies that obey power laws, in: GLOBECOM, 1 (2010), pp. 434438.

[25] R. Albert, A. Barabási. Statistical mechanics of complex networks, Reviews of Modern Physics, 74(1) (2002), pp. 4797.

[26] J. Cockayne, S. Hedetniemi, Towards a theory of domination in graphs, Networks, 7(3) (1977), pp. 247-261.

[27] R. E. Kalman, Mathematical description of linear dynamical systems, Journal of the Society of Industrial and Applied Mathematics Control Series A, 1 (1963), pp. 152192.

[28] S. Guha, K. Samir, Approximation algorithms for connected dominating sets, Algorithmica, 20(4) (1998), pp. 374-387.

[29] J. Yu, N. Wang, G. Wang, D. Yu, Connected dominating sets in wireless ad hoc and sensor networks A comprehensive survey, Computer Communications, 36(2) (2013), pp. 121-134.

[30] A. Bentaleb, A. Boubetra, S. Harous, Survey of clustering schemes in mobile ad hoc networks, Communications and Network, 5 (2013), pp. 8-14.

[31] J. Kneis, D. Mölle, S. Richter, P. Rossmanith, Parameterised power domination complexity, Information Processing Letters, 98(4) (2006), pp. 145149.

[32] C. Alcaraz, E. E. Miciolino, S. Wolthusen, Structural controllability of networks for non-interactive adversarial vertex removal, in: the Eighth International Conference on Critical Information Infrastructures Security, Springer, Critical Information Infrastructures Security, LNCS 8328 (2013), pp. 120-132.

[33] C. Alcaraz, E. E. Miciolino, S. Wolthusen, Multi-round attacks on structural controllability properties for non-complete random graphs, the 16th Information Security Conference, Springer (2014), In press.

[34] C. Alcaraz, S. Wolthusen, Recovery of structural controllability for control systems, the Eighth IFIP WG 11.10 International Conference on Critical Infrastructure Protection (2014), Springer-Heidelberg, LNCS 441, pp. 47-63.

[35] R. G. Downey, M. R. Fellows, Parameterised complexity, Monographs in Computer Science, Springer-Verlag, Heidelberg, Germany, 1999.

[36] IEC-62351-4, Power systems management and associated information exchange - Data and communications security - Part 4: Profiles including MMS, 2007, `http://www.iec.ch/smartgrid/standards/`, retrieved April 2015.

[37] IEC-62351-6, Power systems management and associated information exchange - Data and communications security - Part 6: Security for IEC-61850, 2007, `http://www.iec.ch/smartgrid/standards/`, retrieved April 2015.

[38] C. Alcaraz, S. Zeadally, Critical control system protection in the 21st century: threats and solutions, IEEE Computer, 46(10) (2013), pp. 74-83.

[39] C. Alcaraz, and S. Zeadally, Critical infrastructure protection: requirements and challenges for the 21st century, International Journal of Critical Infrastructure Protection, Elsevier Science, 8 (2015), pp. 53-66.

[40] S. Groat, M. Dunlop, W. Urbanksi, R. Marchany, and J. Tront, Using an IPv6 moving target defense to protect the Smart Grid, 2012 IEEE PES Innovative Smart Grid Technologies (ISGT) (2012), pp. 1-7.

[41] K. Zeilenga, COSINE LDAP/X.500 schema, RFC-4524, 2006, `http://tools.ietf.org/html/rfc4524`, , retrieved April 2015.

[42] Honeywell International, RBAC driven least privilege architecture for control systems, U.S. Department of Energy, office of electricity delivery and energy reliability (2012), pp. 1-2.

[43] IEC-62351-7, Power systems management and associated information exchange - Data and communications security - Part 7: Network and system management (NSM) data object models, 2007, http://www.iec.ch/smartgrid/standards/, retrieved April 2015.

[44] INCITS, For Information technology role based access control, American National Standard for Information Technology, ANSI INCITS 359-2012 (2012).

[45] T. Nguyen, W. Perkins, T. Laffery, and Dl Pecora, Knowledge base verification, AI Magazine, 8(2)(1987), pp. 65–79.

[46] R. O'Keefe, and D. O'Leary, Expert system verification and validation: a survey and tutorial, Artificial Intelligence Review (1993), pp. 3-42.

[47] Apache Directory Studio, 2006-2013,http://directory.apache.org/studio/, retrieved April 2015.

[48] M. Smith, Definition of the inetOrgPerson LDAP object class, RFC-2798, 2010, http://www.ietf.org/rfc/rfc2798.txt, retrieved April 2015.

[49] Sandia National Laboratories, the Java rule engine (JESS), 2007-2013,http://herzberg.ca.sandia.gov, retrieved April 2015.

[50] G. M. Burt, and J. R. McDonald, Potential advantages of a diagnostic expert system for assisting operator response to system events, Third International Conference on Power System Monitoring and Control (1991), pp. 222-224.

[51] Ulrich Lang, Rudolf Schreiner, Manageable Smart Grid security policy automation, Information Systems Security Association Journal (2012), pp. 10-36.

[52] NIST, NISTIR 7628: guidelines for Smart Grid cybersecurity - vol. 1, Smart Grid cybersecurity strategy, architecture, and high-level requirements, revision 1 (2013).