Access Control in Safety Critical Environments

Christoph Ruland
Faculty of Science and Technology
University of Siegen
Siegen/Germany
Email: Christoph.Ruland@uni-siegen.de

Jochen Sassmannshausen
Faculty of Science and Technology
University of Siegen
Siegen/Germany
Email: Jochen.Sassmannshausen@uni-siegen.de

***Abstract*—This paper describes an access control system for industrial automation and control systems (IACS) and similar automation systems for smart energy grids. The intended area of applications of the proposed system are control/station and substation networks to protect connected devices and associated safety relevant settings from unauthorized access. The proposed solution for access control introduces a two-stage access control schema. The first stage evaluates policies based on the eXtensible access control markup language (XACML) and the second stage uses knowledge about the system's behavior to prevent malicious or accidental operations that have negative impact on the systems stability. The access control system uses RFC 5755 attribute certificates to store properties of subjects, resources, rsp. objects and system information. The design and implementation of the system considers safety requirements such as timing requirements or availability in order to enable an integration in safety-critical environments.**

I. Introduction

The terms "Safety" and "Security" describe different aspects of cyber-physical systems. The term "safety" usually refers to functional safety, which includes correct system behaviour, fault handling and protection functions in order to protect humans, environment and equipment. The term "security" usually refers to information security which includes correct system behavior, fault handling and protection functions in order to protect humans, environment and equipment. The expression "security" is used for information of stored data and communication security of transmitted data, which include services and mechanisms for authentication, detection of manipulations and encryption to support confidentiality. Access control relies on communication security and authentication of origin and prevents unauthorized access to data. The authors of [1] state that both safety and security deal with risks: They differentiate between accidental risks that originate from the system itself (safety) and malicious risks that originate from attackers (security). However, the consequences of both types of risks are similar, they can include financial loss or damage to humans, environment and equipment. This leads to the conclusion that safety and security have several commodities and must not be treated separately. However, Cheminod et. al. state that many approaches focus on introducing security to existing systems that have been designed without security considerations [2].

The sometimes quoted phrase "There is no safety without security" states that safety is dependent on security measures that prevent manipulations and unauthorized access to the system. For example, the origin of commands must be authenticated in order to prevent manipulation of exchanged data and commands which could lead to system misbehavior. Safety systems have to rely on secure data exchange. However, security can also have a negative impact on safety properties. Some communication scenarios require very low latencies and jitter, which makes the application of certain cryptography-based security measures difficult or even impossible. Additionally, safety becomes dependent on the correct execution of the implemented security mechanisms.

Historically, industrial automation and control systems (IACS) were isolated systems. Therefore, security was not critical and limited resources of embedded devices were not capable to perform cryptographic operations. Security was restricted to physical access control to systems [3]. Nowadays, an IACS is no longer an isolated system. There are connections to the corporate network and the wide area network. The field devices are potentially accessible by a growing number of instances. This requires security measures against external and internal attackers. Device settings and processes must be protected from unauthorized manipulation to ensure a sufficient safety level. Cardenas et. al. [4] give several examples where lack of security enabled attacks which lead to damage to equipment/environment.

The rest of this paper is organized as follows: Section II will give a short overview of safety and security in IACS and related systems, section III describes a certain use case for power systems, which gives motivation to develop the access control system. Section IV describes the proposed system and section V discusses the results and gives a conclusion.

II. Background

A. Relationship between Safety and Security

Kriaa et. al. give an overview of different approaches that combine safety and security aspects for IACS applications [1]. Their overview includes a definition of safety and security, similarities, standardization initiatives and a detailed overview and comparison of approaches of other authors. Cheminod et. al. focus on security issues in industrial networks [2]. They give a present a review of security issues in IACS and also mention that safety and security must be treated jointly. The approach presented in [4] proposes a development scheme that integrates security considerations into the system development process. Reference [5] points out four different relationships

between safety and security requirements. This approach is also picked up and explained in detail by [1]. Basically, there are four different relations between safety and security measures: *Conditional dependency*, *Mutual reinforcement*, *Antagonism* and *Independence*. This means that some safety/security measures can be mandatory for other safety/security measures, but there may also be conflicts between safety and security. On one hand, sensor data must be protected from manipulation in order to ensure correct behavior of safety mechanisms or control loops that rely on authentic sensor values. On the other hand, cryptographic measures can cause a certain overhead (especially asymmetric algorithms like digital signatures) that conflicts with requirements for low latencies. Kriaa et. al. state that the system design must include these considerations and give examples for design schemes. Figure 1 shows a simplified design scheme. It is important that the system

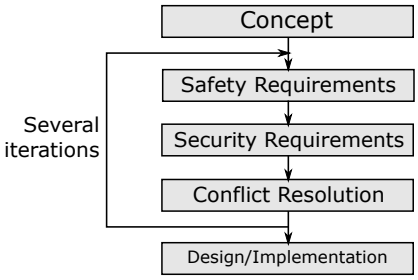

Fig. 1. Iterative system design with safety/security conflict resolution.

design includes security and safety considerations as well as conflict resolution. The process of defining safety and security requirements can include the mentioned four different relations between safety and security requirements. This phase of system design is iterative and requirements can be adjusted in order to resolve conflicts.

B. IACS System Architecture

Figure 2 shows a common architecture of an typical IACS that consists of different networks and attached devices. The

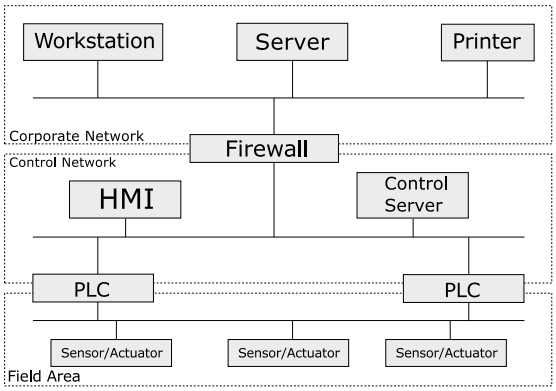

Fig. 2. Architecture of IACS with different connected networks

architecture is very similar to Smart Grid station architectures that will be the target of the later described access control solution. An IACS comprises different networks that have different properties and also different safety and security requirements. On the lowest level there are networks for fast exchange of data such as sampled measured values or event triggered data. These networks are designed to meet hard timing requirements and security mechanisms are difficult to implement. Automation devices are also accessible from a control network, which can be seen as local area network. Typical devices in this network usually are human-machine-interfaces, control servers, etc. The control network is connected to the corporate network and secured by a firewall. For more detailed IACS architecture descriptions, also see [6]. Timing requirements in the control network are not as hard as on the process bus resp. field areas. This enables the implementation of cryptographic security mechanisms such as access control that prevents unauthorized access from the control network. Guidelines and overviews such as [6], [7] and [2] state that there are different security requirements for corporate networks and control networks. Security requirements include *confidentiality*, *integrity (including authenticity)* and *availability*, but the priority for data exchange in control networks is set on integrity and availability, whereas confidentiality has highest priority in ”traditional” corporate networks. This requirements have to be taken into consideration during the design phase of security solutions for IACS.

C. Communication Security and Access Control

Communication security protects exchanged data and commands against manipulation by external attackers. Communication security is a mandatory requirement for access control, which protects assets from unauthorized access by both internal and external entities. The observation of [2], that security often is developed independent from the rest of the system also is partly true for the Smart Grid area. Standards like IEC 61850 introduce data models and communication protocols, but security is introduced separately by standards like IEC 62351-3, IEC 62351-4 and IEC 62351-6. The OPC Unified Architecture is very popular for IACS. It introduces a communication security model, which is similar to the approach taken by IEC 62351-3 and IEC 62351-4: A secure connection (which protects the data exchange) is established and end-to-end peer authentication/data origin authentication is performed at application level. Access Control has to be implemented on top of the communication security. The later presented access control solution will operate on application level. There can be additional access control at lower communication layers in the OSI stack, such as layer 2 (filtering dependent on MAC addresses) or layer 3 (filtering dependent on IP addresses). However, access control on application level is highly important because of the end-to-end property and the possibility to perform access control with knowledge about exchanged data and command. For example, a firewall that performs access control based on IP and Port does not ”know”, which data is contained by the protocol data units. Threat reports such as [8] or [9] show that there is a huge threat potential by internal attackers or corrupted entities that have

access to the system. Cardenas et. al. [4] mentions this threat and states that access control and principles like *separation of duty* or *least privilege* are essential to reduce impact of corrupted entities.

D. Access Control in IACS and Smart Grids

IEC 62351-8 introduces Role-based Access Control (RBAC) for Power Systems [10]. However, there are newer access control models such as attribute-based access control (ABAC) that also include environment conditions (including the current system state) and allow the definition of arbitrary object properties in form of attributes. This technology is more flexible and the consideration of environmental attributes enables the definition of access control policies that consider the current system state. Descriptions of RBAC, ABAC and Hybrid-RBAC-ABAC solutions can be found in [11], [12] and [13]. Several authors describe implementation of access control in different scenarios. The Authors of [14] and [15] describe attribute-based access control in industrial systems and [16] focuses on a RBAC implementation in power systems that use IEC 61850. These descriptions of access control do not focus on safety in detail. There are other publications such as [17] that focus on access control and emergency situations, but not on automation system specific implementations. Ferreria et. al. [17] focus on medical applications, but a similar "break the glass" - algorithm can be used in industrial systems. Ref. [18] focuses on challenges of safety-critical environments and also points out that there is a need to introduce security in order to achieve safety requirements. Attribute-based access control can be implemented using the OASIS-standard XACML [19]. Both [16] and [14] use XACML to realize access control, but do not consider safety requirements. Availability is highly important and has to be considered during system and policy design. A comparison and performance evaluation of XACML implementations in different scenarios that include access control scenarios with huge policy- and rule sets can be found in [20]. There are different systems that include external system parameters in access control policies. These systems are often called *Risk-adaptive*, *Situation-aware* or *Context-aware*. Examples can be found in [21], [22] or [23]. Information about situations and risk is provided by additional modules and can dynamically affect security parameters such as role-right associations.

III. Motivation

The presented access control system was developed for the Smart Grid and IEC 61850 environments, but it can also be integrated into similar scenarios in IACS. The main components of the security solution are not IEC 61850 specific but generic and could be used with industry standards like OPC UA as well. The German Association of Energy and Water Industries (BDEW) published a whitepaper that describes a "Traffic Light Concept" that represents different states of the power grid [24]. The electric grid can have different states such as "green", "yellow" and "red". State "green" means that there are no problems, whereas "red" indicates that the grid is in emergency state. There may be even more fine-grained system states or additional states such as *Maintenance*, but the three mentioned states are common states in power systems. The green state is the normal state. If the system is in a critical state, there are restrictions to certain actions. The current state of the grid is provided externally. Involved entities, such as the station operator, the Transmission System Operator (TSO) or the Distribution System Operator (DSO) have certain rights dependent on the current system state. For this purpose, an access control system is required. The access control system must know the current system state in order to choose the correct policies for access control. A change of the system state can be triggered by local events that are monitored by a monitoring system. The second possibility is an external change of the system state with a pre-defined command. This possibility allows the external change of the system state with a "break-the-glass" mechanism. The Access Control System must support different system states and the system state must be provided by a local system state monitor. Access Control technologies such as XACML supports the definition of policies that include environment conditions such as date, time or arbitrary conditions like the mentioned system state. The scenario described in [14] also includes environment conditions such into access control rules. However, these boundary conditions cannot be arbitrary complex, they are restricted by the policies and [20] shows that a huge set of policies and rules may have negative impact on performance. This motivates the introduction of a second stage that is independent of XACML. This second stage is responsible for checking of boundary conditions that have to be met within the system. Boundary conditions contribute to safety properties as they prevent arbitrary changes of safety-critical settings. This is motivated by the given threat reports that also state that critical manipulation of systems are not always malicious, they can also be result of accidental changes by authorized entities. Both stages are independent. The secondary conditions can still be evaluated if the Policy Decision Point (PDP) fails.

IV. Access Control Solution

A. Architecture

Figure 3 shows the overall architecture of the proposed system. The Policy Enforcement Point (PEP) is the component of a system where access control decisions are actually enforced. The access control system can be integrated either as part of end devices or as an intermediate access control firewall. [15] considers bump-in-the-wire devices that can be used to introduce access control for legacy devices. The *Secure Context Handler* is inspired by the system architecture introduced in the XACML 3.0 standard [19], where the (abstract) component between PEP and other components such as Policy Decision Point (PDP) or Policy Information Point (PIP) is referred to as *Context Handler*. The proposed system calls this part *Secure Context Handler*, because the origin of all information that are used to evaluate requests (e.g. attributes, policies, secondary conditions) have been authenticated.

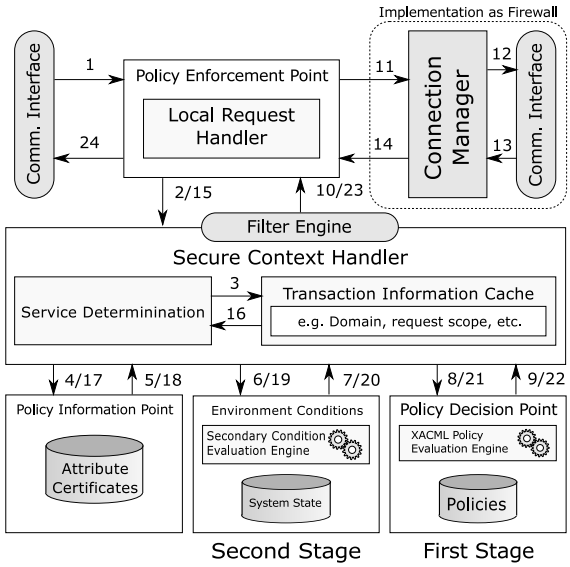

Fig. 3. The architecture of the security solution and the request handling procedure

B. Request Handling

Figure 3 shows the system architecture and the process of request handling. Incoming requests (1) are passed to the secure context handler (SCH) (2) where request PDUs are analyzed by the service determination unit and the service (e.g. read/write) is determined. This system component and the services are standard specific (e.g. OPC UA, IEC 61850). The SCH can store certain information about the request for later access (3). The SCH retrieves attributes about subject and resource by the PIP (4,5). The SCH then requests the Secondary Condition Evaluation Engine to check the boundary conditions according to the current active profile and the information contained by the request (6), the response contains information about whether the conditions are met or not (7). The primary conditions are evaluated by the PDP and the SCH sends an evaluation request to the PDP (8) and receives a response (Permit/Deny) (9). The PDP uses XACML-based policies and rules (more details in section IV-D). The steps (6,8) and (7,9) are independent and can be performed in parallel, if possible. The filter engine (10) is a module that transforms request/response messages. For example, a request can contain multiple write requests and some of them may be allowed and others may be denied. The filter engine filters out parts of the request that may not be performed. A similar system is demonstrated in [25] where parts of an XML document are filtered. If the access control system is implemented as firewall, the filtered request is forwarded to the end device (11,12). The connection manager manages connections to the end device and retrieves responses from end devices (13,14). If the system is not implemented as firewall, the response (ie. the processing result) is obtained by the local request handler. Responses are handled just like requests (15-24). The response evaluation can also include information about the associated request (16). This is required for certain IEC 61850 services where the content of response messages has to be viewed in the context of the request message. For some requests, critical data can be part of the response. Examples of such services are requests to obtain information about the data model. IEC 62351-8 refers to this as the "VIEW" right. Some entities are not allowed to see the whole data of a device. This is related to the least-privilege principle or the need-to-know-principle.

C. Attribute Definition

RFC 5755 defines *"An Internet Attribute Certificate Profile for Authorization"* [26]. The attribute certificates defined by RFC 5755 are very similar to X.509 subject certificates. However, an attribute certificate (AC) does not contain a public key but it contains an element *"Holder"*, which identifies an entity to which the AC belongs to. This can either be a distinguished identifier of a subject with an associated X.509 certificate or a unique identifier of a resource. An AC is used to associate additional information (attributes) to the holder of an AC. The advantage of AC is the independence of subject certificates. Therefore, the subject certificate can remain the same even if there is a new attribute certificate that invalidates previous attribute certificates. Attribute certificates usually have much shorter validity periods than subject certificates. Another advantage of AC are security aspects, which are the main motivation for the usage of AC for this security solution. All information of an AC is secured by a digital signature. The Secure Context Handler and the PIP are able to verify the information using the corresponding CA certificate. The solution uses AC for both subjects and resources.

D. First Stage: XACML Policy Evaluation

1) XACML performance: XACML stands for *eXtensible Access Control Markup Language* and defines a syntax for access control policy definition and a procedure for policy evaluation. Some performance evaluations such as [20] show that evaluation of XACML performance causes overhead if the number of policies and included rules is large. The evaluated policies contain up to 1000 rules. This has heavy impact on policy evaluation times. The main goal of policy design is a small set of policies and rules in order to maintain fast response times.

2) Implemented policies:

a) Area of Responsibility: This policy checks if the accessed device/data is part of the subject's Area of Responsibility (AoR). The AoR also includes the subjects working hours, i.e. start and end of the shift. This policy prevents subjects from accessing the system outside of their working hours and also prevents access to parts of the system the user is not responsible for. This policy is very coarse-grained and consists of two rules (one for the time-controlled access and one for the area).

b) Classification and Clearance: Classification and Clearance is a common access control principle that can be implemented using attribute-based access control systems such as XACML. Each data object of the system receives a certain classification. The classification is result of a previous

risk analysis that considers the impact to the system if an unauthorized entity gains access to this particular data object. The classification is dependent on the action that is actually performed. For example, a write operation on certain settings may have critical impact on safety functions, but the setting itself may not be confidential. The counterpart of a resource classification is the assignment of clearance information to system participants. A low clearance means only low impact on the whole system, if he concerned entity is corrupted. The Classification and Clearance system is required by standards like ISO 27002 [27] and ISO 27019, which is an extension of ISO 27002 for power systems [28]. The policy for classification and clearance contains only one rule that is applicable for every service that accesses or views classified data elements. The rule itself states that access will be denied if the accessing subject's clearance is lower than the data object's classification.

c) Least Privilege: The least privilege principle is an important mechanism to restrict influence of single entities to the absolute minimum required level. The proposed access control system uses the following approach: The security management defines a set of tasks that can be performed in a certain system state. A task T comprises a set of actions that have to be performed on different resources. For example, there may be a task *"Motor Control"*, which requires read access to sensor values like speed, heat, torque, etc. and write access to the settings of the motor (speed, direction). Tasks are both associated with subjects and objects. The association *Subject - Task* defines, which subject is allowed to perform which task and the association *Task - Object* defines, which objects and actions are required to perform a certain task. There are two sets of attributes, one set $Tasks_S$ of subject attributes that contains tasks of the subject and $Tasks_R$ of resource attributes that contains tasks associated with a resource. Let $Tasks_S(Subj, St)$ be the set of tasks associated with a subject $Subj$ and system state St. Let $Task_R(A, St, R)$ be the set of tasks that require action A on resource R when the system is in state St. If a user $Subj$ tries to perform action A on a resource R, the secure context handler determines the two sets $Tasks_S(Subj, St)$ and $Task_R(A, St, R)$ according to the current system state. The XACML policy for this access control principle contains only one rule that compares both sets and determines if there are common members in which case access will be granted. This approach that models permissions as part of association between groups of subjects and objects is very similar to the approach taken by "Next-Generation Access Control" [13].

d) Additional Policies: The presented access control system is not restricted to a certain number of policies and rules. Additional policies can be integrated into the system, but the impact on safety requirements has to be taken into account. The presented three policies are the basic setup that can be extended, if required.

3) Performance analysis: It is possible to optimize the given policy set and integrate all given policies and rules into one single policy. This results in a single policy with a total of four rules, which is very little compared to the evaluated scenarios in [20]. The access control system aims to adjust rights by assignment of new attributes to entities whereas the policy set remains static. The context handler is responsible to collect all required attributes according to the system state and to provide the information to the PDP within a XACML request. This section presents some evaluation results for the given PDP. The first two policies are static, and it can be expected that evaluation of both policies has constant time. The third policy compares two sets of attributes that do not have a constant size. It can be expected that huge sets of attributes can have a notable impact on the evaluation performance. As a consequence, the performance results are determined for two sizes of attribute sets (10 and 100 attributes with only one common member). Table I shows the evaluation of the described scenario. The used term "SIZE" in both right columns refers to the size of both sets $Tasks_S(Subj, St)$ and $Task_R(A, St, R)$. The test scenario measures the average times of 1000 evaluated (different) requests. The requests were created in such a way that every rule of the policy must be evaluated to obtain valid results. The table helps to choose adequate hardware setups for different safety requirements such as maximum processing times. The table only shows the PDP performance, the actual request handling (see figure 3) may take longer. The measured times refer to steps (8,9) and (21,22) of the request handling scheme shown in figure 3.

TABLE I
XACML PERFORMANCE EVALUATION FOR BASIC RULE SET AND DIFFERENT PLATFORMS

Platform	SIZE = 10	SIZE = 100
700 MHz ARM11, single core, 256 MB RAM	27 ms	45 ms
900 MHz ARM Cortex-A7, 4 cores, 1 GB RAM	5.6 ms	7.2 ms
1.2 GHz ARM Cortex-A53, 4 cores, 1 GB RAM	3.1 ms	3.9 ms
3.2 GHz Intel Core-i5, 2 Cores, 8 GB RAM	1.1 ms	1.3 ms

E. Second Stage: Evaluation of Secondary Conditions

1) General: The last section described the first stage of the access control system that consists of a XACML-based PDP. Safety-critical systems differ from "classical" corporate networks. Wrong actions can lead to damage to equipment and may have impact on system safety. Cardenas et. al. [4] state that an access control for IACS or cyber-physical systems in general requires knowledge about the system it protects. This is the main motivation for the second stage of the proposed access control system. The second stage ensures that certain boundary conditions are met. This makes malicious system changes more difficult and helps to reduce the impact of accidental changes. Standards for role-based access control like IEC 62351-8 do not take this into account. It focuses only on services that can be either permitted or denied on certain data elements [10]. The second stage of the proposed security solution "looks inside" the content of requests and can negate the access control decision of the PDP.

2) Definition of Conditions for Data Elements: The access control system uses a dedicated format for the definition of boundary conditions. It supports the definition of conjunctive and disjunctive boolean expressions that can include arithmetic expressions, constants, parts of the data model, values contained by write requests and comparison functions. The definition of these expression is inspired by the expressions supported by programming languages such as C++ or Java. The evaluation of these expressions requires a dedicated parser. Secondary conditions have to be evaluated when entities try to access and change data elements. Secondary conditions are capable to ensure that certain actions are only allowed in certain system states (e.g. switching operations are performed in a certain order). More complex scenarios may include correlations between certain components of the system. For example, there could be an linear relation between generator speed and measured heat. The boundary conditions for generator speed settings can models this relationship to ensure that certain safety requirement (maximum temperature) will be ensured for future operation.

The security solution supports boundary conditions that are global or system state specific. Each defined system state can be associated with own secondary conditions that are applicable for actions that originate from this particular system state. Another type of secondary conditions are bound to a user and are only applicable for certain user. Figure 4 shows how secondary conditions are associated with users. Secondary

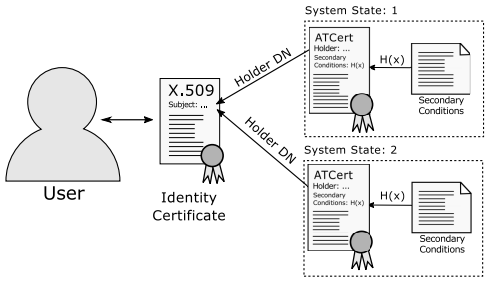

Fig. 4. Secondary conditions bound to users using attribute certificates and cryptographic one-way hash functions.

conditions are defined in a configuration file. The attribute certificate itself stores only a hash code of the configuration file. The hash code is calculated with a secure one-way-hash function (e.g. SHA-256) and is protected from manipulation by the attribute certificate's signature. Both the configuration file and the certificate can be verified by the secure context handler. An entity of the system may have several attribute certificates, each for a special pre-defined system state. The hash code of the secondary conditions is stored as special attribute in the attribute certificate. Figure 4 also shows the principle of how subject certificates are referenced by attribute certificates: The Holder Distinguished Name (DN) refers to the element "subject" of the X.509 identity certificate.

3) Evaluation of Secondary Conditions: The secondary condition evaluation engine (SCEE) (also see fig. 3, steps 6,7 and 19,20) receives an evaluation request that contains a command to change a certain part of the data model (e.g. toggle a switch). The SCEE collects all secondary conditions that include the particular data object. This collection is dependent on global secondary condition definitions, global definitions for the current system state and user-dependent secondary conditions. This collection of secondary conditions can be done in advance for every system state and data object. A rebuild of these collections is only required if new attribute certificates are pushed to the secure context handler or if global definitions of secondary conditions are changed. The evaluation of secondary conditions involves the dedicated parser and other values of the system (e.g. sensor values or state information that are included in secondary condition definitions). The SCEE has to obtain all required values in order to perform the evaluation. The retrieval of these values can be straightforward if the system is implemented directly as part of the end system. However, this process can be more complicated, if the system is implemented as access control firewall. In this case, the system has to request these data values from the end device or subscribe state change notifications if the communication system supports this possibility. There can be an overhead in processing time that has to be taken into account.

4) Monitoring of System State: The proposed system needs to have information about the current system state. There can be a local system state monitor that provides the required information. The system state can be monitored by a local state monitor that determines the system state according to predefined rules. Another possibility is the external change of the system state such as described by [24]. A station in the grid may not know that the connected grid is in an alert state and actions have to be performed by external operators. A mechanism that allows breaking access control rules in emergency situations is described in [17]. A similar "Break the glass" mechanism can be introduced in order to change system states that allow extended privileges to certain entities.

F. Consideration of safety requirements

The secondary conditions are also capable to perform basic access control for single users or groups, because these conditions can be bound to users trough attribute certificates. It is also possible to reduce the secondary conditions to a minimum and extend the XACML rules by additional rules that consider boundary conditions. There has to be a load balance between PDP and Secondary Condition Evaluation Engine (SCEE). It is also possible to define certain system states with a different policy and rule set and different secondary conditions. Special system states could include scenarios that cover SCEE failure or PDP failure. This contributes to a higher availability and reduces the dependencies on single system components such as SCEE and PDP. The design of the proposed system aims to achieve a high level of robustness and minimal impact on transaction times and availability. The presented basic setup achieves good performance that can be sufficient for many safety requirements. However, the iterative design schemes mentioned in section II-A (Relationship between Safety and

Security) have high relevance in the later policy design. The performance of the access control scheme is heavily dependent on the implemented policies. The basic system setup does not have huge impact on safety requirements. Huge sets of complicated policies, secondary conditions and rule sets can have notable impact on safety requirements. An iterative design scheme (see figure 1) must be applied during policy and secondary condition development. There has to be a conflict resolution to achieve a acceptable trade-off between safety and security requirements. Risk analysis is very important in policy design phase.

V. Conclusion

An important aspect of the proposed system is the definition of policy-independent boundary conditions that include system state as well as information about values that shall be written to parts of the data model. This is an essential difference to systems that only have a static set of policies and further to systems that are situation aware, but do not implement further checking of boundary conditions. For example, a dynamic role-right association does not protect that data objects to which access was granted from arbitrary manipulation. The proposed system can provide mechanisms that control the modifications on data elements to which access was granted. The definition of this second stage of the system allows the definition of conditions that have to be ensured in order to achieve safety requirements, which is an extension to other situation-aware systems that only perform an adaptive role-permission association or policies that include environment conditions like *Access to object X is allowed only from the local network*. Future extensions will extend the secondary condition evaluation engine with artificial intelligence and algorithms that try to predict the future system state depending on the current system state and requested actions.

References

[1] Siwar Kriaa, Ludovic Pietre-Cambacedes, Marc Bouissou, and Yoran Halgand. A survey of approaches combining safety and security for industrial control systems. *Reliability Engineering and System Safety*, 139:156 – 178, 2015.

[2] M. Cheminod, L. Durante, and A. Valenzano. Review of security issues in industrial networks. *IEEE Transactions on Industrial Informatics*, 9(1):277–293, Feb 2013.

[3] Andrew J. Kornecki and Janusz Zalewski. Safety and security in industrial control. In *Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research*, CSIIRW '10, pages 77:1–77:4, New York, NY, USA, 2010. ACM.

[4] Alvaro Cardenas, Saurabh Amin, Bruno Sinopoli, Annarita Giani, Adrian Perrig, and Shankar Sastry. Challenges for securing cyber physical systems. In *Workshop on Future Directions in Cyber-physical Systems Security*. DHS, July 2009.

[5] L. Pitre-Cambacds and M. Bouissou. Modeling safety and security interdependencies with bdmp (boolean logic driven markov processes). In *2010 IEEE International Conference on Systems, Man and Cybernetics*, pages 2852–2861, Oct 2010.

[6] Keith Stouffer, Victoria Pillitteri, Suzanne Lightman, Marshall Abrams, and Adam Hahn. Guide to industrial control systems (ICS) security. National Institute of Standards and Technology, NIST Special Publication 800-82r2, jun 2015.

[7] D. Dzung, M. Naedele, T. P. Von Hoff, and M. Crevatin. Security for Industrial Communication Systems. *Proceedings of the IEEE*, 93(6):1152–1177, June 2005.

[8] Industrial Control System Security - Top 10 Bedrohungen und Gegenmanahmen 2016. BSI-Verffentlichungen zur Cybersicherheit, 2016. https://www.allianz-fuer-cybersicherheit.de/ACS/DE/_/downloads/BSI-CS_005.pdf?__blob=publicationFile&v=4#download=1.

[9] Vormetric Insider Threat Report, 2015. http://enterprise-encryption.vormetric.com/rs/vormetric/images/CW_GlobalReport_2015_Insider_threat_Vormetric_Single_Pages_010915.pdf.

[10] IEC 62351-8: Power systems management and associated information exchange - Data and Communication Security - Part 8: Role-based Access control, August 2007.

[11] Ed Coyne and Timothy R. Weil. ABAC and RBAC: Scalable, Flexible, and Auditable Access Management. *IT Professional, Volume 15, Issue 3*, pages 14–16, May-June 2013.

[12] Edward J. Coyne Richard Kuhn and Timothy R. Weil. Adding Attributes to Role-Based Access Control. In *IEEE Computer, vol. 43, no. 6*, pages 79–81, June 2010.

[13] David Ferraiolo, Ramaswamy Chandramouli, Rick Kuhn, and Vincent Hu. Extensible Access Control Markup Language (XACML) and Next Generation Access Control (NGAC). In *Proceedings of the 2016 ACM International Workshop on Attribute Based Access Control*, ABAC '16, pages 13– 24, New York, NY, USA, 2016. ACM.

[14] Erkan Yalcinkaya, Antonio Maffei, and Mauro Onori. Application of Attribute Based Access Control Model for Industrial Control Systems. *International Journal of Computer Network and Information Security(IJCNIS), Vol.9, No.2*, pages 12–21, 2017. DOI: 10.5815/ijcnis.2017.02.02.

[15] J. H. Huh, R. B. Bobba, T. Markham, D. M. Nicol, J. Hull, A. Chernoguzov, H. Khurana, K. Staggs, and J. Huang. Next-Generation Access Control for Distributed Control Systems. *IEEE Internet Computing, Vol. 20, Issue 5*, September 2016.

[16] Byunghun Lee, Dae-Kyoo Kim, Hyosik Yang, and Hyuksoo Jang. Role-based access control for substation automation systems using xacml. *Information Systems*, 53:237 – 249, 2015.

[17] A. Ferreira, R. Cruz-Correia, L. Antunes, P. Farinha, E. Oliveira-Palhares, D. W. Chadwick, and A. Costa-Pereira. How to Break Access Control in a Controlled Manner. In *19th IEEE Symposium on Computer-Based Medical Systems (CBMS'06)*, pages 847–854, 2006.

[18] J. C. Knight. Safety critical systems: challenges and directions. In *Proceedings of the 24th International Conference on Software Engineering. ICSE 2002*, pages 547–550, May 2002.

[19] The eXtensible Access Control Markup Language (XACML), Version 3.0. OASIS Standard, January 2013. http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf.

[20] Fatih Turkmen and Bruno Crispo. Performance evaluation of xacml pdp implementations. In *Proceedings of the 2008 ACM Workshop on Secure Web Services*, SWS '08, pages 37–44, New York, NY, USA, 2008. ACM.

[21] Marc Hüffmeyer, Pascal Hirmer, Bernhard Mitschang, Ulf Schreier, and Matthias Wieland. Situation-aware access control for industrie 4.0. In Paolo Mori, Steven Furnell, and Olivier Camp, editors, *Information Systems Security and Privacy*, pages 59–83, Cham, 2018. Springer International Publishing.

[22] Kaiyu Wan and Vangalur Alagar. Context-aware security solutions for cyber-physical systems. *Mobile Networks and Applications*, 19(2):212–226, Apr 2014.

[23] S. Kandala, R. Sandhu, and V. Bhamidipati. An attribute based framework for risk-adaptive access control models. In *2011 Sixth International Conference on Availability, Reliability and Security*, pages 236–241, Aug 2011.

[24] Smart Grid Traffic Light Concept. German Association of Energy and Water Industries (BDEW), march 2015. https://www.bdew.de/media/documents/Stn_20150310_Smart-Grids-Traffic-Light-Concept_english.pdf.

[25] A. Khurat and J. Abendroth. A mechanism for requesting hierarchical documetns in xacml. In *2008 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, pages 202–207, Oct 2008.

[26] S. Farrell, R. Housley, and S. Turner. An internet attribute certificate profile for authorization. RFC 5755, RFC Editor, January 2010.

[27] ISO/IEC 27002: Information technology - Security techniques - Code of practice for information security controls, November 2016.

[28] ISO/IEC 27019: Information technology - Security techniques - Information security controls for the energy utility industry, October 2017.