SPAPI: A Security and Protection Architecture for Physical Infrastructures and Its Deployment Strategy Using Wireless Sensor Networks

Hafiz Abdur Rahman
Department of ECE
University of British Columbia
Vancouver, Canada
rahmanha@ece.ubc.ca

Konstantin Beznosov
Department of ECE
University of British Columbia
Vancouver, Canada
beznosov@ece.ubc.ca

Abstract

In recent years, concerns about the safety and security of critical infrastructures have increased enormously. These infrastructures can easily become subjects of physical and cyber attacks. In this paper, we propose a software architecture named Security and Protection Architecture for Physical Infrastructures (SPAPI) for the protection of these critical infrastructures and for other non-military uses. SPAPI has hierarchical, loosely coupled, autonomous management modules for authentication, monitoring and the policy-based control of their respective domains. Due to their autonomous design, each management module works independently according to their predefined policies. In this paper we discuss the design and application of SPAPI in the context of a hypothetical chemical process facility.

1. Introduction

People's lives in a modern state depend on the smooth functioning of critical infrastructures such as important government and corporate buildings and manufacturing facilities; water supply, electricity, telecommunication, gas and petroleum distribution networks; interstate roads, etc. Until recently, safety and security were not prime concerns for infrastructure management, but a volatile world situation has changed this mindset. There are many valid concerns about the security of national critical infrastructures [1]. For example, several chemical industries belong to these critical infrastructures. The safety of these chemical industries is critical because many of them use hazardous production materials (HPMs) whose release can cause severe harm to plant operators and the general public. The traditional approach of Security Vulnerability Assessment (SVA) for chemical industries considers such potential releases as accidental risks and ignores the possibility of intentional risks [2]. However, a recent study shows that intentional risk is high, especially from terrorists and disgruntled employees [3].

Given this situation, Air Products and Chemicals Inc. (APCI) conducted SVA on a large number of chemical plants and came up with a set of recommendations [4]. Two of these recommendations are very important and require computer-aided technology for their implementation. The first is to divide the plant premises into well-defined and protected functional domains where access to protected areas is controlled through authentication mechanisms. The second recommendation is to install improved infrastructure monitoring and safety technology such as closed-circuit television, motion detectors, passive infrared sensors and alarms.

A typical chemical plant can be divided into a few well-defined functional domains, such as HPM Storage Facilities, Truck Unloading Section, and Production Equipment and Property Storage Areas [3]. Access to these functional blocks can be controlled based on operator roles within the plant. A mature access control technology named Role Based Access Control (RBAC) [5] appears to easily support this requirement of controlled access to the protected functional domains (details in Section 4).

Infrastructure monitoring usually entails different types of sensor-based systems (details in Section 2). To ensure infrastructure security and safety, we need sensor-based systems that will work consistently with an organization's long-term security and safety policies, and provide support for complex authentication needs. These systems should have the ability to accept any such policy and role definition at the root level in a high-level language and propagate those to sub-domains. Systems should also have the ability to monitor safety devices (sensors) and act upon any policy violations. These systems should work consistently in case of communication failures or the partial damage of any of its parts due to accidents or malicious attacks. To the authors' knowledge, there is no such architecture that satisfies all the above criteria. Sensor-based techniques developed so far for the defense community [6] are related to the monitoring and controlling of short-lived real-time events. The techniques developed for building automation (details in Section 2) focus mainly on adapting a building's "behavior" to its inhabitants and nature. Neither of these approaches is policy driven. Their architectures are mostly centralized and are subject to a single point of failure.

Based on these observations, we propose a new software architecture named Security and Protection Architecture for Physical Infrastructures (SPAPI, pronounced “spapee”). SPAPI is based on hierarchical autonomous architecture and supports authentication, physical event monitoring and security policy enforcement. SPAPI management modules are responsible for their respective domains. Due to their decentralized and autonomous design, they work independently according to their predefined policies, even if communication is disrupted with the central management module. We have designed SPAPI on two solid foundations. One is the Role Based Access Control (RBAC) model [5] and another is the Policy Based Management model [7]. The policy-based approach makes SPAPI-based systems highly configurable. To make the architecture expressive and portable, we propose SPAPI policy implementation architecture in XACML language [8]. In this paper, we provide background, rationales and design, and suggest different implementation and deployment strategies for SPAPI architecture.

The rest of the paper is organized as follows. Section 2 discusses related work. A brief introduction to the Policy Based Management and Role Based Access Control models is given in Section 3. Section 4 explains how the SPAPI-based system could be used to secure a chemical process facility. SPAPI architecture, implementation strategies and deployment techniques are discussed in Sections 5 through 7. Section 8 discusses future work. Conclusions are drawn in Section 9.

2. Related Work

Over the last several years, there has been significant research performed on infrastructure monitoring and management, namely by the structural health monitoring, building automation and video surveillance communities. In this section, we discuss typical hardware and software architectures used by these groups to address different aspects of monitoring and control problems. Though this section is not a survey, it will introduce the typical architectural approach taken by different groups to achieve certain objectives.

Research by the structural health monitoring community focuses mainly on ensuring the safety and reliability of civil or mechanical engineering structures over their life spans through monitoring systems. The technology uses different kinds of sensors to detect and to remotely address any compromise in material structural integrity. An important study from the structural health monitoring community was done by Kottapalli et al. [12]. This research began with the observation that cable based monitoring systems have high installation costs and are vulnerable to ambient signal noise corruption. The research focused on finding an optimum wireless network topology and an appropriate protocol stack. In that study, they proposed a two-tiered wireless architecture in which sensor units (SU) were clustered as in a cellular network. A local site master (LSM) was assigned to each cluster to coordinate SUs and collect their data. The SU cluster formed the network’s lower tier, whereas LSM and Central Site Master (CSM) formed the upper tier. The lower tier used a 915 MHz frequency band for SU to LSM communication using the spread spectrum frequency hopping technique, which allowed up to 52 sensors per cluster. The upper tier used a 2.4 GHz frequency band for LSM to LSM and LSM to CSM communication. This band was represented by the IEEE 802.11b based standard. A special purpose Time Division Multiple Access (TDMA) protocol was used for all communications. Micro-Electro-Mechanical Systems (MEMS) based sensors were used for data acquisition. SUs were battery–powered, and LSM and CSM were connected to a regular power supply.

Building automation research focuses on developing sensor-based techniques to increase inhabitants’ comfort and to minimize building power consumption [13]. The objective of one of the most recent studies in this area was to increase user comfort by minimizing user interactions [14]. A building knowledge-base was developed from the asynchronous interest-based messaging between autonomous agents. Agents collected data from different sensors such as motion sensors and wall switches, and used an unsupervised learning algorithm to build a knowledge repository based on fuzzy logic. The knowledge repository controls the effectors (hearers, lights, electronically operated windows, etc.) and improves the building’s performance. All the devices are wired to LonWorks building network [13]. A similar approach was employed in [15] using hierarchical fuzzy-agent architecture.

The work of the video surveillance community involves using optical (video camera) and non-optical sensors (e.g., electrical, thermal, and biological) for security surveillance and environment monitoring. Typical research challenges include video content analysis, multi-camera calibration, motion detection and tracking, object detection recognition, etc. Fidaleo et al. [16] developed an architecture for the secure capturing and sharing of surveillance video data. This architecture, named Network Sensor Tapestry (NeST), has a centralized server, a client interface library, a layered XML messaging scheme and a data visualization and control interface. The system supported different types of clients, including PDA-based client hardware and remote sensor interface devices, that were connected to NeST using a TinyOS based microcontroller.

In this paper, we propose a SPAPI deployment strategy based on a two-tiered wireless architecture [12]. This strategy will be discussed in Section 7.

3. Policy Based Management Model and Role Based Access Control Model

Policies are rules governing a system's choices in behavior [17]. In the policy-based management approach, a set of explicit decision-making technologies are embedded into the management component of computing systems (Figure 1). There are two types of policies: authorization policies and obligation policies. An authorization policy defines what a manager is allowed to do, and an obligation policy defines what a manager must do. The manager has to interpret policies to achieve the overall goal of system management. The important benefit of such an approach is that it simplifies the management of large and complex systems by allowing administrators to specify management objectives in terms of declarative rules that need to be maintained. Otherwise, detailed instructions would be needed to achieve this same functionality. A policy-based system provides an automated feedback loop (Figure 1) so that a policy violation reported by monitoring agents automatically triggers corrective action as specified in the policies. Since system behaviours are governed by rules, such systems are easy to customize at a higher level of abstraction and are more adaptive to environmental changes and changes in application needs. Since its inception the policy-based management model [7] has gained popularity for managing services in large and complex environments, such as quality-of-service in communication networks, the provisioning of a virtual private network, the deployment of distributed applications and the security management of network services. In recent years, such an approach has been studied for Cluster Computing [18] and 3G Wireless Networks [19].

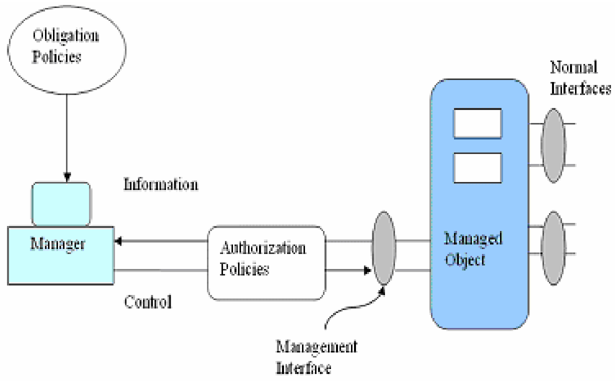

Figure 1. Policy-based management model.

Role-Based Access Control (RBAC) models [5] have received extensive attention from industry and academia as a generalized approach to controlling access to computerized resources. The basic RBAC model defines a set of elements such as users, roles and permissions (Figure 2). This model's key concept is that permissions are associated with roles, rather than users. Users gain access rights due to their role membership and they can be dynamically assigned and removed from a specific role without changing the permissions they have. RBAC has been recognized as an effective means to simplify authorization management in large organizations, which allows addressing the confidentiality, integrity and availability aspects of a large scale system, such as multi-domain digital government and complex web-based applications [20].

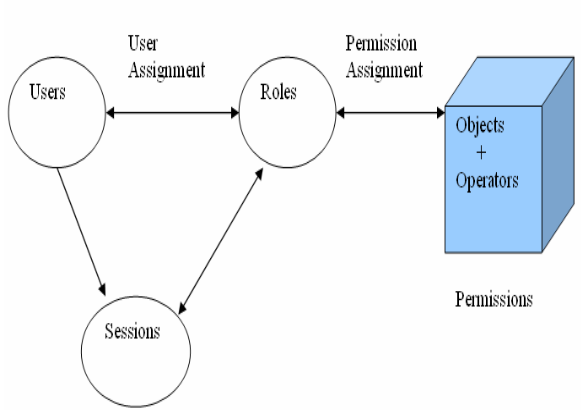

Figure 2. RBAC core components.

Section 5 presents a detailed implementation strategy for Policy Based approach and RBAC in SPAPI architecture.

4. SPAPI-based Approach to Securing a Chemical Process Facility

To show the usefulness of RBAC and Policy-Based Management in critical infrastructure safety and security management, we analyze the security issues of a chemical process facility. A similar building access problem has been discussed in [21, 22]. Both of these papers considered the access control problem as a multi-sensor data fusion problem. Biometric data from multiple sensors are processed through a Bayesian Network to make access control decisions, where the objective is to reduce the false rejection rate. Our approach to this problem is at a higher level of abstraction, where we assume users each have a valid ID and their access is restricted according to the access control policy and safety considerations of the chemical process facility. We believe our approach is more realistic and comprehensive than those reported in [21, 22]. In this context, we consider the following examples:

Example 1:

We consider a large chemical process facility, where people from different departments work together. There are people from the HPM (hazardous production materials) Storage Facilities, the Truck Unloading Section, the Production Equipment department, the Property Storage department, etc. Only people from the HPM Storage facility can access the HPM Storage facility; only those in the Property Storage department can access the Property Storage depot, and so on. However, if someone belongs to a particular department's management group, she might get access to other department facilities including her own. This access control problem cannot be solved with the widely used Discretionary Access Control (DAC) model [23], as any plant operator may inadvertently transfer his right to someone who should not get any such privilege. Mandatory Access Control (MAC) can satisfy this security requirement, but whenever any user or object is added or removed from the system the security label in the authorization and authentication database will need updating. This implies a significant administrative overhead for any large system. We propose to use RBAC as the most suitable model for dealing with this problem. Since access permissions are associated only with organizational roles rather than individual users, the administering of users and resources is largely simplified.

Example 2:

The plant management wants to invite a few university guests to the production facility. These people would visit the plant's Production Equipment department. Instead of creating static user IDs and assigning guest roles to each of them, they (plant management) would rather tell the guests to collect digital certificates from a trusted third party through the Internet. Once these invited people present the certificates (from their smart card or PDA) upon their arrival, a temporary guest role (RBAC) will be assigned to each of them. This role will give them access to the plant and to the production equipment department. To add a digital certificate to our authorization model, we propose to use a certificate validation architecture such as SPKI/SDSI [24]. Our strategy of assigning a RBAC guest role to a valid digital certificate holder will ensure the smooth integration of SPKI/SDSI in the RBAC-based authentication model.

Example 3:

Chemical plant officials want to ensure safety for everyone. Therefore, they monitor events (normal state change) and alarms (persistent exception that has to be taken care of) from different kinds of sensors for fire, smoke, hazardous gas, humidity, vibration (earthquake) etc. In case of any safety concern, they would deny access to any part of or the whole plant, even to people with valid access credentials. Depending on the severity of the safety risk, they might want everyone to evacuate the plant. For this level of control, they need to implement different policies to activate actions based on events received from the monitoring network. Unfortunately, RBAC, like any other access control model, does not support the event-condition-action (ECA) paradigm to ensure the safety, reliability and control of network resources. This aspect of safety administration can be addressed by Policy Based management [7] in addition to RBAC. The use of RBAC and the Policy-Based model within SPAPI are discussed in the following section.

5. SPAPI Architecture

In this section, we discuss the components of the Security and Protection Architecture for Physical Infrastructures (SPAPI).

5.1 SPAPI Components

SPAPI consists of three levels of hierarchical autonomous management modules (Figure 3). These are Global Manager (GM), Domain Manager (DM) and Local Manager (LM). There are one or more managed sensors/devices at the leaf of the tree per LM (Figure 3). Each LM represents a single management point in physical space. LM collects multiple types of data and/or controls its object of interest. For example, a LM could collect both temperature and vibration data from two sensors pointing to a single physical object. Domain represents a continuous and non-overlapping physical space that contains one or more LMs. For instance, a DM could control a room (domain) using one or more LMs. GM is the overall supervisor of the system and manages one or more DM.

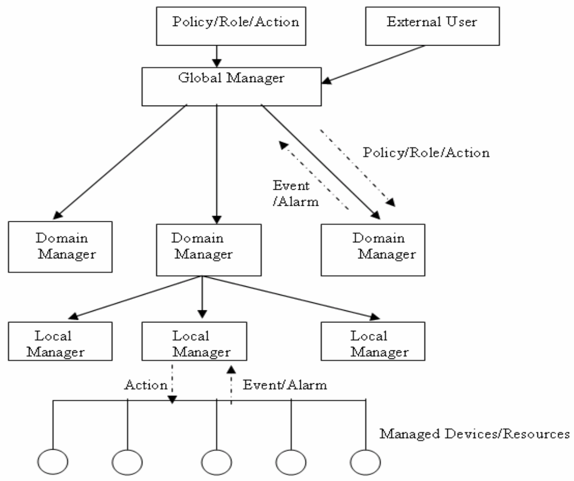

Figure 3. SPAPI System Architecture.

In an abstract sense, SPAPI uses these management entities to control information flow based on two considerations: authentication (role) and policy constrains. There are subtle differences in their design and implementation, which are discussed in the following sections (Figures 4 to 6).

Global Manager: The Global Manager is the system's central administrative unit (Figure 4). It receives Policy, Role Definition and Action specifications from administrative tools, keeps central Event/Alarm logs, authenticates digital certificates and assigns guest roles to temporary users. Global Manager is the central coordination point for its Domain Managers. It has a centrally managed Policy, Role and Action Repository. These repositories interact closely with the Policy Processor. Policies, roles and actions are defined centrally at the Global Manager level, but due to the autonomous design principle of SPAPI, they are propagated to the local levels through a distribution service. All system-wide updates also propagate in regular intervals. Policy Processor for Global Manager is responsible only for those actions that have global impact, for instance, a shutdown of the entire plant due to a major accident.

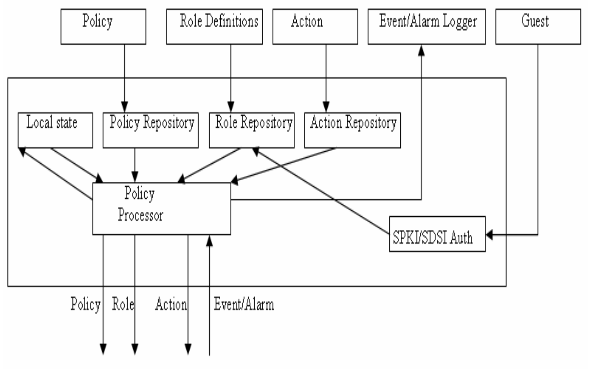

Figure 4. Global Manager.

One important feature of Policy Processor is that it re-examines the authorization of those users who are already authorized by RBAC before giving access to any resource (Figures 4 to 6). One key benefit of this approach is that access to resources can be controlled independent of users' roles in the organization. For instance, if the "two-person rule" is required in some sensitive areas, it can be implemented by this mechanism. This makes resource management easier and more flexible. Section 5.2 presents details of the policy processor.

Domain Manager: The Domain Manager (DM) is a domain's local administration point. User authentication and most of the policy evaluations are done at this level. It has functional blocks similar to Global Manager, except that it lacks an event logging service and support for digital signature validation. DM's Policy, Role, and Action repository get updated from Global Manager on a regular basis. However, one key element that makes DM important in SPAPI architecture is the event consolidation point from the Local Managers. All local level policy-based decision-making is done at this level. Issues related to such a decision-making approach are discussed in Policy Processor design. DM forwards only those alarm/events to GM for further processing that might have a significant impact on the other domains or that are important otherwise, such as spreading fire or smoke. Such events and information flows are specified in the policy specification.

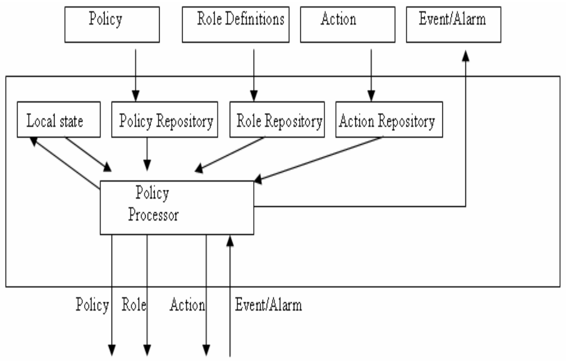

Figure 5. Domain Manager.

Local Manager: The Local Managers (LMs) are connected to the actual event sources (sensors), act on them and monitor their status at regular intervals. LMs are low-power computing devices. To conserve computation power and memory, they run Policy and Role Proxies. These proxies "point" to the Domain Manager for their actual definition. Therefore, computations related to decision-making in the local node are minimum. The proxies collect local events and alarms, and pre-process them according to the required data collection format specified in the following section. After this pre-processing, they forward these events/alarms to the Domain Manager for decision and control. The Domain Manager forwards them to the Global Manager if required by the policy. An individual user may get access to LM or managed objects (sensors) based on her role and policy constraints.

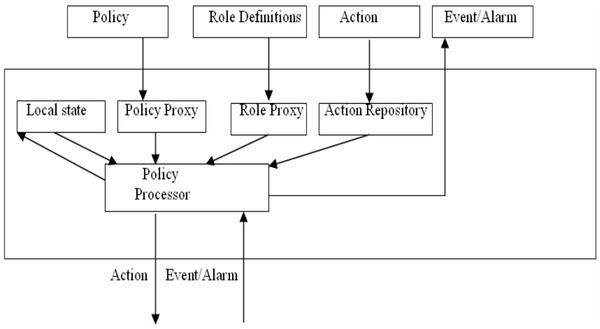

Figure 6. Local Manager.

5.2. The Policy Processor

The Policy Processor (PP) is the core of the management modules. It interprets policies supplied by human users and controls the system's operation according to those policies. The policy description is a logical description of approved system states that satisfies strategic goals. The implementation of these policies requires a mechanism to separate approved (normal) system states (events) from the unapproved system states (alarms). A simple statistical approach, such as computing the deviation from the mean, is too naive to separate any potential alert from normal events. This approach requires a significant volume of data (large time window) before making any such decisions. More sophisticated statistical techniques can help in such cases. Rahman [25] studied one such technique, named threshold crossing statistics, in the context of irregular network traffic characterization. The idea was to use the non-linear property of the M/G/1 queue to set up triggering rules for abnormal traffic patterns. Thottan and Ji [26] studied a similar approach by using the Generalized Likelihood Ratio (GLR) to detect network traffic anomalies. This approach required the knowledge of normal traffic patterns (a possible policy definition) and its likely changes. The computation of the likelihood ratio was based on the autoregressive traffic characteristic over non-overlapping window in which the traffic flow was assumed stationary. However, defining system characteristics (policy description) can be application-specific and as such are beyond the scope of this paper. In this section, we discuss how such a policy description can be written for Policy Processor.

Policy Processor interprets policy text written in a policy description language. Several studies were performed in both industry and academia to formally specify security and management policies [27]. The selection of a policy language has a significant impact on system design, implementation and execution. The following factors are important in this consideration:

- Expressive power
- Ease of use
- Ability to capture RBAC constraints
- Declarative way to specify event-condition-action rules
- Ability to specify complex workflow tasks
- Wide support from industry for interoperability, portability and cross-platform development

As discussed in [27], there is no single language that captures all these requirements. However, with careful design choice most of these requirements can be met. In Section 6, we discuss one such implementation architecture using OASIS eXtensible Access Control Markup Language (XACML) 2.0 [8] for SPAPI. XACML is XML-based and flexible enough to express complex policies. Version 2.0 of XACML provides good support for RBAC. Besides, it has the notion of "obligation" that defines requirement associated with policy. However, XACML has no native support for the event-condition-action based paradigm for physical event monitoring and control. Our proposed implementation idea uses WBEM architecture [30] which has an event provider and subscriber mechanism and a declarative event-processing language named WQL (WBEM Query Language) to circumvent this shortcoming.

Another candidate for SPAPI implementation is Policy Description Language (PDL) [27]. PDL was developed at Lucent Bell Laboratories and has been used in Lucent switching products. It is based on the event-condition-action paradigm. However, PDL does not support access control policies, which needs to be extended to support RBAC.

5.3 Events and Alarms

In this section, we discuss events and alarms, the last important components in our architecture.

5.3.1 Events

Actual data collected from the environment, events are generated by different sensor and network elements. Whenever an event occurs, a corresponding condition clause is evaluated. If the evaluation result is true, an action will be executed. These data include temperature, humidity, smoke and gas measurements, vibration, audio and video. To make decisions based on these data in the policy processor, we propose a data collection format that pre-processes these data at the Local Manager level by adding additional attributes. Up to now, we have identified the following attributes. Additional attributes will be added depending on the data processing requirement.

Source Type: The data source can be a scalar type, such as a temperature source, or a vector type, such as a moving object detected by a video camera. The source type will determine how the data will be interpreted and if any additional sources are required for detection and estimation.

Source Location: Source location is essential to building a spatial correlation among these data and very important for decision-making.

Source Priority: A priority number will be given to the source. Based on the priority value, the data may be forwarded to a specific location (say Global Manager) and action will be taken. Prioritizing events has proved very useful for BACNet [13].

Time stamp: Time stamp will enable the data synchronization and interpolation of the missing data points. It will also enable temporal searches and queries.

Scale: Different types of data are collected in different formats: the temperature in Celsius, the sound in decibels, etc. Scale information is essential for data conversion and evaluation.

Data Type: The collected data may have different formats (integer, double, array, etc). These need to be mentioned for consistent data processing.

5.3.2 Alarms

Alarms are a persistent presence of unexpected events and signify potential problems. They are collected from regular sources. However, any irregular state for a sustained period of time implies an alarm. Alarms can be represented according to the RFC 3877 guideline [28]. This standard-based representation will allow the policy designer to process any alarms using a unified framework.

6. SPAPI Implementation Architecture

This section outlines our plans for implementing SPAPI. The Distributed Management Task Force (DMTF) [29] and IETF Policy Framework Working Group jointly defined a policy information model as an extension to the Common Information Model (CIM), which is known as Policy CIM (PCIM) [9]. CIM is a platform-independent abstract representation of the managed objects and their inter-relations. At the macro level, CIM is a part of another management architecture named Web-Based Enterprise Management (WBEM) [30], which was initiated by DMTF to unify the management of a distributed computing environment. To date, WBEM has been implemented in Windows, Linux, Solaris, VMX, Novell and many other platforms [31]. Based on the wide availability of WBEM and its platform-neutral design, we propose an implementation strategy for SPAPI based on the PCIM model using WBEM architecture. An implementation of an XACML-based policy interpreter for the PCIM model was discussed in [32].

We consider implementing SPAPI on a typical MS Windows platform for a number of reasons. First, WBEM, which we plan to use in SPAPI, is implemented in MS Windows as a system service called Windows Management Instrumentation (WMI). All physical and logical devices have virtual representation called provider in WMI managed object repository. Second, SPAPI policy modules will be implemented as PCIM policy provider objects [9] in the WMI provider repository. Similar to that, all sensor devices will have their specific type of data providers in WMI repository. Policy provider will interact with all these sensor data providers using the DCOM communication protocol. In a Windows environment, RBAC [33] is implemented as a COM+ service. As such, RBAC information will also be available using the corresponding provider type. According to the policy loaded into the policy provider (using a policy editor), the policy provider will guide/direct system states based on information received from the sensor data provider and the RBAC provider. WMI has an event provider and a subscriber mechanism that facilitates this event-driven interaction. Only the Global and Domain managers will use the WBEM-based implementation architecture (as these will be more powerful machines). Local managers are low-power computing devices and will use object proxy to communicate with the Domain Manager for actual policy and role definitions.

7. Physical Deployment

A possible SPAPI deployment strategy may use a two-tiered wireless architecture as in [12]. However, the performance of a hexagonal cellular cluster in our domain definition needs further investigation. Besides, since much recent research uses the TinyOS-based Mica 2 microcontroller to interact with the sensor devices, we think it would be a better choice than the Atmel microcontroller used in [12]. In addition, due to low overhead and the simplicity of the SPINS [34] sensor network security protocol, we propose its use in a lower tier (Local Manager to Domain Manager) of SPAPI architecture for secure data communication. For the upper tier (Domain Manager to Global Manager) IPv6-based security communication can be used. These wireless sensors [10] [11] make SPAPI-based systems quickly deployable in massive numbers close to the event source. However, SPAPI is a generic architecture that can be deployed using any other deployment technology.

Figure 7. Mica 2 Sensor Mote (www.xbow.com).

8. Future Work

In this paper we have presented design and implementation plans for SPAPI. Our future plan is to build a prototype based on this architecture for a chemical industry and test its performance. We would also like to extend SPAPI ideas for the security of non-physical infrastructures, such as data communication networks.

9. Conclusions

In this paper, we propose SPAPI, a new framework for the safety and security of physical infrastructures. SPAPI uses a new approach to integrate authentication with physical safety. The policy-based management approach is the key to this integration. This policy-based framework, with its automatic triggering mechanism, could significantly reduce administrative overhead.

SPAPI is based on hierarchical autonomous modules, which increase a system's dependability in a significant manner. We also propose an event format, which would be useful for security administration as well as for infrastructure usage pattern detection. Our design is standard–based, is portable and supported by the available set of development tools, which eventually imply low management costs. Our proposed implementation technique uses the new low-cost sensor network, which will ensure fast, easy and low-cost deployment.

10. Acknowledgement

We would like to thank the anonymous reviewers for providing helpful feedback and our LERSSE [35] colleagues for their constructive criticism and for reviewing the initial drafts of this paper. ICICS editorial assistant Ben D'Andrea was instrumental in making this paper more readable.

References

[1] Critical infrastructure protection links: http://www.ocipep.gc.ca/relatedlinks/cip_e.asp.

[2] Moore, David A., "The new risk paradigm for chemical process security and safety", *Journal of Hazardous Materials*, v 115, n 1-3, pp 175-180, SPEC. ISS., November 2004.

[3] J. R. Lemley, V. M. Fthenakis, P. D. Moskowitz, "Security risk analysis for chemical process facilities", *Process Safety Progress*, v 22, n 3, pp 153-160, September 2003.

[4] B. R. Dunbobbin, T. J. Medovich, M. C. Murphy, A. L. Ramsey, "Security vulnerability assessment in the chemical industry", *Process Safety Progress*, v 23, n 3, pp 214-220, September, 2004.

[5] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli. "Proposed NIST Standard for Role-Based Access Control." *ACM Transactions on Information and System Security (TISSEC)*, 4(3): pp 224-274, 2001.

[6] G. McIntyre and K. Hintz; "A Comprehensive Approach to Sensor Management, Part I: A Survey of Modern Sensor Management Systems," *IEEE Transactions on SMC*, April 1999.

[7] Morris Sloman, "Policy Driven Management for Distributed Systems", *Journal of Network and Systems Management*, Vol. 2, No. 4, pp 333-360, December 1994.

[8] Oasis XACML Version 2.0: http://docs.oasis-open.org/xacml/access_control-xacml-2_0-core-spec-cd-04.pdf

[9] Policy Core Information Model -Version 1 Specification, RFC 3060, February 2001, http://www.ietf.org/rfc/rfc3060.txt.

[10] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. "Wireless sensor networks: a survey.", *Computer Networks*, 38:393-422, 2002.

[11] Crossbow Technology, Wireless Sensor Networks: http://www.xbow.com/Products/products.htm

[12] Venkata A. Kottapalli, Anne S. Kiremidjian, Jerome P. Lynch, Ed Carryer, Thomas W. Kenny, Kincho H. Law, Ying Lei, "Two-Tiered Wireless Sensor Network Architecture for Structural Health Monitoring", *SPIE's (The International Society for Optical Engineering) 10th Annual International Symposium on Smart Structures and Materials*, San Diego, March 2003.

[13] Deborah Snoonian, "Smart buildings", *IEEE Spectrum*, v 40, n 8, p 18-23, August 2003.

[14] U. Rutishauser, J. Joller, Josef, R. Douglas, "Control and learning of ambience by an intelligent building", *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans.*, v 35, n 1, pp 121-132, January 2005.

[15] Hani Hagras, Victor Callaghan, Martin Colley, Graham Clarke, "A hierarchical fuzzy-genetic multi-agent architecture for intelligent buildings online learning, adaptation and control", *Information Sciences*, v 150, n 1-2, pp 33-57, March 2003.

[16] Douglas A. Fidaleo, Hoang-Anh Nguyen, Mohan Trivedi, "The Networked Sensor Tapestry (NeST): A privacy enhanced software architecture for interactive analysis of data in video-sensor networks", *Proceedings of the ACM Second International Workshop on Video Surveillance and Sensor Networks*, pp 46-53, 2004.

[17] Seraphin Calo and Morris Sloman, "Policy-Based Management of Networks and Services", *Journal of Network and Systems Management*, Vol. 11, No. 3, pp 249-252 September 2003.

[18] Lionel Sacks, Ognjen Prnjat, Ioannis Liabotis, Temitope Olukemi, Adrian Ching, Mike Fisher, Paul McKee, Nektarios Georgalas, and Hideki Yoshii., "Active Robust Resource Management in Cluster Computing Using Policies", *Journal of Network and Systems Management*, Vol. 11, No. 3, pp 329-350 September 2003.

[19] Said Soulhi, "3G Wireless Networks Provisioning and Monitoring via Policy Based management", *International Conference on Communication Technology Proceedings*, ICCT, v 2, pp 1137-1143, 2003.

[20] J. Joshi, A. Ghafoor, W. G. Aref, and E. H. Spafford, "Digital government security infrastructure design challenges". *IEEE Computer* 33, 2, pp 66-72, February 2001.

[21] Lisa Ann Osadciw, Pramod K Varshney, Kalyan Veeramachaneni, "Improving Personal Identification Accuracy Using Multisensor Fusion for Building Access Control Applications", *International Conference on Information Fusion*, July 7-11, 2002, Annapolis, Maryland.

[22] Nuri Yilmazer and Lisa Ann Osadciw, "Sensor Management and Bayesian Networks", *SPIE's (The International Society for Optical Engineering) Aerosense (Defense and Security Symposium)*, April 12-16, 2004, Orlando, Florida.

[23] J. McLean, "Security models" in J. Marciniak, editor, *Encyclopaedia of Software Engineering. Wiley Press*, 1994.

[24] R. L. Rivest and B. Lampson., "SDSI A Simple Distributed Security Infrastructure". CRYPTO 1996.

[25] Hafiz Abdur Rahman, "Network Status Monitoring and Prediction using WBEM", *MSEE Thesis*, May 2000, Purdue University, West Lafayette, Indiana, USA. URL: http://min.ecn.purdue.edu/~rahmanha/thesis.pdf

[26] Marina Thottan, Chuanyi Ji, "Adaptive Thresholding for proactive Network Problem Detection", *IEEE Computer Society, International Workshop on Systems Management*, Newport, Rhode Island, April 1998.

[27] Morris Sloman and Emil Lupu, "Security and Management Policy Specification", *IEEE Network*, pp 10-19, March/April 2002.

[28] Alarm Management Information Base, September 2004: http://www.ietf.org/rfc/rfc3877.txt.

[29] Distributed Management Task Force, Inc (DMTF), "Common Information Model (CIM) Specification", v.2.9 (http://www.dmtf.org/standards/cim), January 5, 2005.

[30] Web-Based Enterprise Management (WBEM): http://www.dmtf.org/standards/wbem.

[31] OpenWBEM: http://www.openwbem.com.

[32] Emir Toktar, Edgard Jamhour, Carlos Maziero, "RSVP policy control using XACML", *Proceedings - Fifth IEEE International Workshop on Policies for Distributed Systems and Networks*, pp 87-96, 2004.

[33] Role-Based Security Administration: http://msdn.microsoft.com/library/default.asp?url=/library/enus/cossdk/htm/pgservices_security_9l.asp

[34] A. Perrig, R. Szewczyk, V. Wen, D. Cullar, and J. D. Tygar. "SPINS: Security protocols for sensor networks." *Proceedings of the Annual International Conference on Mobile Computing and Networking*, MOBICOM, pp 189-199, 2001

[35] LERSSE Homepage: http://lersse.ece.ubc.ca