# A Security Framework for Smart Ubiquitous Industrial Resources

Anton Naumenko, Artem Katasonov and Vagan Terziyan
Department of Mathematical Information Technology
P.O.Box 35, FIN-40014, University of Jyväskylä, Finland
annaumen@cc.jyu.fi
artem.katasonov@jyu.fi
vagan@it.jyu.fi

**Abstract.** Conventional approaches to manage and control security seem to have reached their limits in new complex environments. These environments are open, dynamic, heterogeneous, distributed, self-managing, collaborative, international, nomadic, and ubiquitous. We are currently working on a middleware platform focused on the industrial needs, UBIWARE. UBIWARE integrates Ubiquitous Computing with Semantic Web, Distributed AI, Security and Privacy, and Enterprise Application Integration. In this paper, we describe our long-term vision for the security and privacy management in complex multi-agent systems like UBIWARE, SURPAS. The security infrastructure has to become pervasive, interoperable and intelligent enough to naturally fit UBIWARE. SURPAS aims at policy-based optimal collecting, composing, configuring and provisioning of security measures. Particularly, we analyze the security implications of UBIWARE, present the SURPAS research framework, and the SURPAS abstract architecture.

## 1 Introduction

Globalization of the economy, global and intercultural value chains, large-scale industrial environments, cooperative systems for the international production, logistic and marketing could hardly be imagined without the rapid evolution of information and communication technologies (ICTs). Moreover, continuous advances of ICTs and their adoption in the industrial world have been guaranteeing improvement and efficiency of industrial technologies in the last decades. Recent advances in networking, sensor and RFID technologies, etc allow connecting various physical world objects to the IT infrastructure, which could, ultimately, enable realization of the “Internet of Things” and the ubiquitous computing

visions. However, the adoption of new ICTs in the traditional production industries, e.g. the process industry, the machinery industry, etc, is relatively slow. It is mainly because of the growing complexity of emerging ICTs, inadequate security infrastructures, and the fact that the research in ICTs usually focuses on the industries with a short cycle of innovations deployment, such as health care or banking, largely overlooking the needs of the production industries.

In response to these problems, we are currently working on a new generation middleware platform focused on the industrial needs, Smart Semantic Middleware for Ubiquitous Computing (UBIWARE). UBIWARE integrates the Ubiquitous Computing domain with such domains as Semantic Web, Proactive Computing, Autonomic Computing, Human-Centric Computing, Distributed AI, Service-Oriented Architecture, Security, and Enterprise Application Integration.

This paper focuses on the security challenges in UBIWARE. It analyses the security threats, requirements, implications and measures needed for UBIWARE in the context of its industrial adoption. An industrial case, outlined in the paper, aligns our research results on UBIWARE, as such, with the real world needs and serves as a trigger and source of requirements for the research on security, particularly. We describe our long-term vision for the security and privacy management in emerging new types of environments, which we refer to as Smart Ubiquitous Resource Privacy and Security (SURPAS). SURPAS is mainly based on the advances in the Semantic Web, Multi-Agent Systems, and Ubiquitous Computing domains. Particularly, this paper presents the SURPAS research framework which guides our research towards SURPAS. It is a consolidated formal system of research ideas and prototypes for the interoperable pro-active context-aware self-protecting security management. The main components of the SURPAS research framework are the conceptual semantics of security policies, functionality of security mechanisms, including functional semantics, algorithms, abstract architecture, and reference implementation, and adopting applications in different business domains (e.g. industrial maintenance, subcontracting management, smart house, etc).

The rest of the paper is organized as follows. Section 2 briefly discusses the concept of UBIWARE. Section 3 addresses the security implications and concerns regarding industrial adoption of UBIWARE. Section 4 presents the motivating industrial case. Sections 5 and 6 give a detailed description of the SURPAS research framework and of the abstract architecture of secure agent, respectively. Section 7 concludes the paper.

## 2 The UBIWARE Concept

It is widely acknowledged that as the networks, systems and services of modern IT and communication infrastructures become increasingly complex, traditional solutions to manage and control them seem to have reached their limits. The IBM vision of autonomic computing (e.g. [1]) proclaims the need for self-managed computing systems able of self-configuration, self-optimization, self-protection, and self-healing. We believe that such self-manageability of a complex system requires its components to be to a certain degree autonomous themselves. In other

words, we envision that agent technologies will play an important part in building such complex systems. Agent-based approach to software engineering is also considered to be facilitating the design of complex systems [2][3].

Another problem is inherent heterogeneity in ubiquitous computing systems, with respect to the nature of components, standards, data formats, protocols, etc, which creates significant obstacles for interoperability among the components of such systems. Semantic Web technologies [4] are viewed today as a key technology to resolve the problems of interoperability and integration within heterogeneous world of ubiquitously interconnected objects and systems. Our work subscribes to this view. Moreover, we believe that Semantic Web technologies can facilitate not only the discovery of heterogeneous components and data integration, but also the behavioral coordination of those components (see [5]).

UBIWARE aims at providing support in creation of self-managed interoperable complex industrial systems consisting of mobile, distributed, heterogeneous, shared and reusable resources of different nature, such as smart machines and devices, sensors, actuators, RFIDs, web-services, software, information systems, humans, organizations, etc. Such middleware enables various components to automatically discover each other and to configure a system with complex functionality based on the atomic functionalities of the components. UBIWARE relies on results from the SmartResource project (Proactive Self-Maintained Resources in Semantic Web, see http://www.cs.jyu.fi/ai/OntoGroup/SmartResource_details.htm), i.e. the "Smart Resource Technology" for designing complex interoperable software systems [6]. This technology gives every resource in an industrial system a possibility to be smart (by connecting a software agent to it), in a sense that it would be able to proactively sense, monitor and control own state, communicate with other components, compose and utilize own and external experiences and functionality for self-diagnostics and self-maintenance. The interoperability among the resources is assured by using metadata and ontologies. Fig. 1 provides example of resources that can be integrated using UBIWARE.

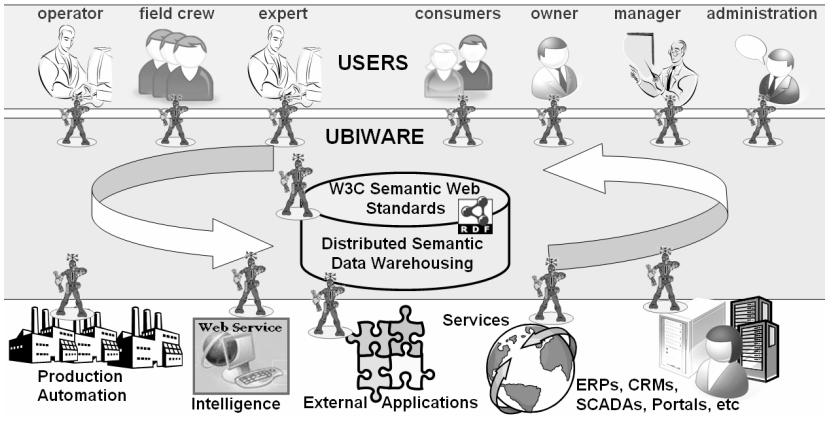


**Fig. 1.** UBIWARE as integrator of industrial resources

## 3 UBIWARE Security Implications

UBIWARE advances existing technologies to a qualitatively new level and brings to life new complex industrial environments, where traditional approaches to manage security fall short. Also, existing security measures for the technologies on which UBIWARE relies, e.g. multi-agent, are not in a mature stage and still require significant elaboration to mitigate associated risks. The security cannot be added to the UBIWARE platform later but the design decisions regarding security have to be thoroughly correlated with the requirements and design of the platform, due to the mutual impact on resulting features of UBIWARE. Thus, the analysis of interrelations between the major characteristics of UBIWARE security is an important task. It has to be conducted throughout the development of UBIWARE.

**Openness** of environments has several dimensions and refers to a range of features. The UBIWARE-supported industrial environments are open in a sense that they create, and are created, by business networks. Every partner of such a network can both use the environment (participate in different roles) and contribute some resources developing the environment further. UBIWARE is built on the top of open standards and technologies developed by open communities. Open industrial environments introduce more challenging security problems due to a greater amount of risks and threats. For example, open-source agent platforms and agents are easier to compromise and alternate for malicious behavior. Also, open environments require complex reputation-based trust management solutions.

**Dynamics** of the environments considered leads to unpredictable changes of their states, due to complexity and ad hoc nature of relations between entities in these environments. This challenges application of traditional techniques for achieving common security goals such as availability, reliability and integrity. In addition, the dynamics complicates introducing adequate techniques to ensure manageability and accountability.

**Heterogeneity** should be considered in the contexts of industrial environments and security infrastructure itself. As to industrial environments, the heterogeneity of resources poses a great variety of security requirements which UBIWARE has to meet. This variety of requirements and the variety of available security solutions related to the technologies, on which UBIWARE relies, complicate the construction of a consolidated security infrastructure. Therefore, the interoperability becomes one of the most important factors.

**Distributed nature** of UBIWARE-supported industrial environments reduces privacy concerns of partnering organizations because of local management of historical data associated with the owned resources. The distribution of control also enhances survivability of the whole system and is known to reduce network traffic and to overcome network latencies [7]. However, the distribution of components complicates the management of security, especially the logging for audit activities.

**Collaborative** or **social nature** of industrial environments correlates with several other characteristics of UBIWARE. The major impact on security is that the communication has to be secured for an efficient and trusted collaboration. This area of research in security has traditionally been addressed. However, the unique features of agent technologies and high demands of industrial applications still keep a place for elaboration.

**Internationality** of today's industrial world requires that the policy languages have to flexible and expressive enough to handle the diversity of cultures, legislations, and traditions in international cooperation.

**Self-management** in terms of security is self-protection that is a vision of pro-active context-aware autonomic security mechanisms to detect, identify and protect against various types of threats [8][1].

**Mobility** of agents and resulting security implications are well addressed in the literature [9][10][11]. However, the UBIWARE security is impacted by the mobility of both resources and agents, and also by the limitations of mobile devices and technologies. The mobility directly affects the solutions for all the security goals and requires some tradeoffs between mobility, performance and security.

**Ambient intelligence**, **ubiquity**, and **pervasiveness** of information technologies have tighten the digital and physical worlds to the extent when security becomes the ultimate issue. The major implication of penetrating ICTs on security is that the risks and negative consequences of security threats become higher than ever. On the other hand, the security infrastructure itself has to become pervasive, interoperable and intelligent enough to naturally fit UBIWARE.

## 4 Motivating Industrial Case

In this section, we exemplify industrial impact, business benefits and security issues of UBIWARE using a case study in the domain of distributed power network management [5] that we performed in collaboration with ABB company (Distribution Automation unit). We present four scenarios of potential new applications that could be created based on UBIWARE and discuss the security implications. ABB is a global vendor of hardware and software for power networks. The power networks themselves are owned, controlled and maintained by some local companies. It is noticeable that the control systems of different companies are not integrated.

However, the information exchange between sub-networks may be very important for fault localization, network reconfiguration, and network restoration when a fault happens on the border of sub-networks. This is our first scenario: introducing an inter-organizational smart middleware solution like UBIWARE could solve this issue. Existence of adequate security mechanisms are a prerequisite though. A challenging research question is how to elaborate flexible and expressive framework for the distributed, collaborative and policy-based management of security.

The second scenario in our vision is related to a new business model that ABB could implement. At present, all ABB expertise gets embedded into hardware or software systems and sold to the customers as it is. A new business model would be to start own Web-services providing implementation of certain algorithms, so the ABB customers will utilize those algorithms online when needed. ABB will be always able to update algorithms, add new, and so on. Noticeable that, if semantically defined, such Web-service can potentially be utilized across the globe even by the customers who never purchased any of ABB hardware or software. Regarding security, this means that UBIWARE must handle secure provisioning of

(semantic) web services, which is still an open research question. We have already targeted issues related to the access control policies in (semantic) Service-Oriented Architecture [12] and in provisioning of Web services in mobile infrastructures [13]. However, there are still problems related to the secure communication, privacy and trust management.

The third scenario in our vision is integration of contextual data with the currently used data such as the network structure and configuration, feeder relay readings, etc (Fig. 2). Such integration can be used for:

- Information about weather conditions, ongoing forest works, or forest fires can be used for evaluating existing threats for the power network. This may be used to trigger an alert state for the maintenance team, or even to do a precautionary reconfiguration of the network to minimize possible damage.
- Facilitation of fault localization. The output of fault localization algorithms is not always certain. The information about threats for the power network that existed at the time when the fault occurred may greatly improve the precision of algorithms. In some situations, contextual information alone may even be sufficient for the localization.
- Contextual information may be also used just to extend the operators' view of the power network. For example, satellite imagery can be used for geographic view (instead of locally stored bitmaps as it is in current systems); also, dynamically-changing information can be accessed and represented on the interface.

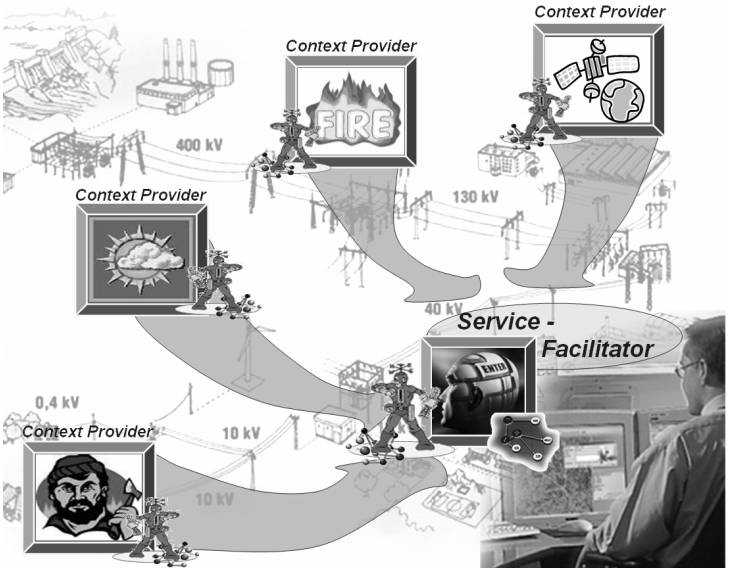


**Fig. 2.** Integration with contextual data (from [5])

Basically, integration of contextual information from external sources requires different approaches to trust management depending on the used techniques and

purposes of integration. Thus, the major question related to security is how to formally compute reputation and trust for the external contextual services because these issues influence the confidence in predicted risks, fault locations, etc.

The last scenario is transferring the knowledge of human experts to automated systems, by means of various data mining tools. For example, now it is always a decision of a human expert which of the existing fault localization algorithms will perform the best in the context of the current configuration of the power network and the nature of the fault. Such decisions made by an expert, along with the input data, could be forwarded to a learning Web-service. After a sufficient learning sample, this Web-service could start to be used in some situations instead of the human expert, e.g. in situations when a faster decision is needed or when the expert is unavailable. Considering humans as objects of access control in machine-to-human interactions is an interesting research question. The data mining algorithms perform better on larger sample sets when they are collected for all equipment of power networks. A question is then how to treat the privacy concerns of the owners of different sub-networks.

## 5 SURPAS Research Framework

Traditional security goals like confidentiality, availability, reliability, integrity, manageability, accountability, responsibility etc, together with conventional measures and mechanisms that support security, do not cover all the needs and threats of new emerging computing environments. UBIWARE poses challenges for the research and development towards more pervasive and intelligent countermeasures. Such countermeasures have to provide the high level of user privacy, effective trust management, built-in self-security, context-awareness, and pro-activity. Moreover, protection of multi-agent systems like UBIWARE is still immature area, where the adoption of conventional security measures and elaboration of new techniques are promising [10][11].

The elaboration of a consolidated security infrastructure following the SURPAS research framework will lead to more innovative and intelligent industrial tools and will transform security from an obstacle to a driver of large-scale industrial collaboration. SURPAS follows the general UBIWARE vision – configuring and adding new functionality to the underlying industrial environment on-the-fly by changing high level declarative descriptions. Regarding security, this means that SURPAS is able of smoothly including new, and reconfiguring existing, security mechanisms, for the optimal and secure state of the UBIWARE-based system, in response to the dynamically changing environment.

Fig. 3 illustrates the SURPAS research framework. The SURPAS semantics [15] is the main conceptual part of the research. The formal explicit specification of semantics is an input for the critical analysis of characteristics of suggested features and for further elaborations of other components of the framework. In a nutshell, the use of ontologies, instead of mathematical security and domain models, is the main characteristic of the SURPAS research.

SURPAS functionality consists of two parts: the enforcement function and the administrative function. The SURPAS enforcement function defines SURPAS run-

time policy-enforcement mechanisms. The SURPAS administration function defines mechanisms for managing SURPAS data like semantic annotations of resources and operations, domain ontologies, ontology-based policies, configuration settings for the enforcement function, etc. Each function is further decomposed into the functional semantics, algorithms for specific tasks, the abstract architecture based on ontologies and abstract use cases, and the reference implementation.

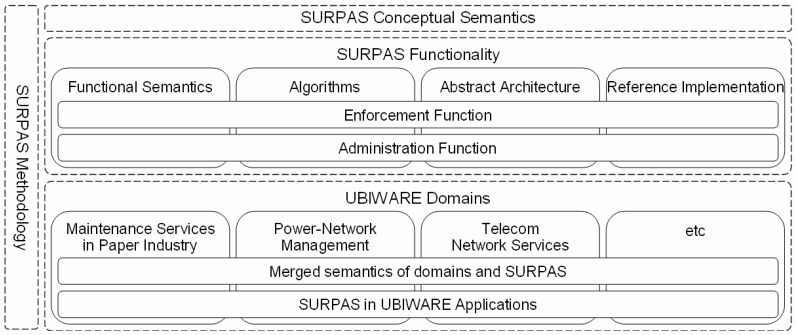


**Fig. 3.** The SURPAS Research Framework

The functional semantics is an abstract specification of functionality. It precisely defines semantics of enforcement and administrative functions that change the state of the UBIWARE-based system to keep it consistent and secure.

The algorithms define explicit step-by-step procedures to perform or compute enforcement and administrative functions, mathematically specified in the functional specification, that are complex and for which the solutions are not obvious from their abstract specifications. Envisioned algorithms for rigorous study include: semantic annotating of requests; retrieving relevant domain ontologies; taxonomic and faceted classification of subjects, operations, and objects of access; retrieving relevant policy statements; resolving conflicts in relevant policies; and access control decision making.

The abstract architecture is an upper view on architectural components of SURPAS and interactions between them. This abstract description captures and reveals only fundamental elements and relations. Basically, the abstract architecture is a bridge between theoretical findings and adoption of SURPAS into practice.

The reference implementation takes the form of a set of software components. It serves for research and development purposes to prototype and test characteristics of proposed ideas, to generate feedback for upper components of SURPAS for further refinement, to make SURPAS tangible for better understanding and evaluating of proposed research ideas, and to implement components for possible reuse in industrial domains.

The UBIWARE adoption domains define industrial application areas of SURPAS. For example, a business of remote maintenance services in the pulp&paper industry may serve as a good domain for adopting UBIWARE. In this

case, SURPAS will be important for managing access to control systems of maintained machinery equipment [14] in the business network of customers. Another promising application area is a business of decentralized management of power-networks owned by different business actors (see section 4). Use of SURPAS in UBIWARE applications aligns SURPAS with the real world needs and issues. Merged semantics of domains and of SURPAS will be the result of merging or/and mapping of SURPAS ontologies with domain ontologies.

The SURPAS methodology is the formally described system of principles, practices and procedures that guides applying the SURPAS in concrete industrial cases of UBIWARE.

Another perspective on the SURPAS research framework would divide the research results according to the main areas of information and system security, namely access control, secure communication, privacy and trust. It is useful to consider along these dimensions all of the conceptual and functional semantics, algorithms, components of the abstract architecture and the reference implementation. In addition, architectural components of UBIWARE define the third, architectural, perspective on the SURPAS framework. In the next section, we further focus on the issues related to the policy-based security management and its adoption according to this architectural perspective.

## 6 SURPAS Abstract Architecture

The central in UBIWARE is the architecture of a secure SmartResource agent depicted in Fig. 4. This architecture of an agent extends the one from [5] by adding the security components. It can be seen as consisting of four layers: reusable atomic behaviors (RABs), behavior models corresponding to different roles the agent plays, SURPAS security policies, and the behavior engine.

A reusable atomic behavior (RAB) is a piece of code implementing a reasonably atomic function. As the name implies, RABs are assumed to be reusable across different applications, different agents, different roles and different interaction scenarios.

The behavior of an agent is defined by the roles it plays in one or several organizations. Some examples of the possible roles for the power-networks domain: operator's agent, feeder agent, agent of the feeder N3056, fault localization service agent, ABB fault localization service agent, etc. Obviously, a general role can be played by several agents. On the other hand, one agent can (and usually does) play several roles, potentially coming from different organizations. A role consists of a set of beliefs representing the knowledge needed for playing the role and a set of behavior rules. Roughly speaking, a behavior rule specifies conditions of (and parameters for) execution of various RABs. Obviously, RABs need to be parameterizable. Notice that, in UBIWARE, if a role specifies the need of interaction with another agent, that agent is always specified by its role, not name or another unique identifier of a particular agent.

The behavior engine is the same for all the SmartResource agents. The behavior engine consists of the agent core, and the two core activities that we named "assign role" and "live". The AssignRole activity is responsible for parsing roles into the

beliefs and behavior rules storages. The Live activity implements the run-time loop of an agent. Introducing SURPAS embeds the policy enforcement mechanism (see details below) into it. The Live activity has also to be protected by some built-in security measures. The Live activity iterates through all the behavior rules, checks them against current beliefs, goals and security policy constraints. After that, it executes RABs together with security mechanisms corresponding to roles and policies, respectively.

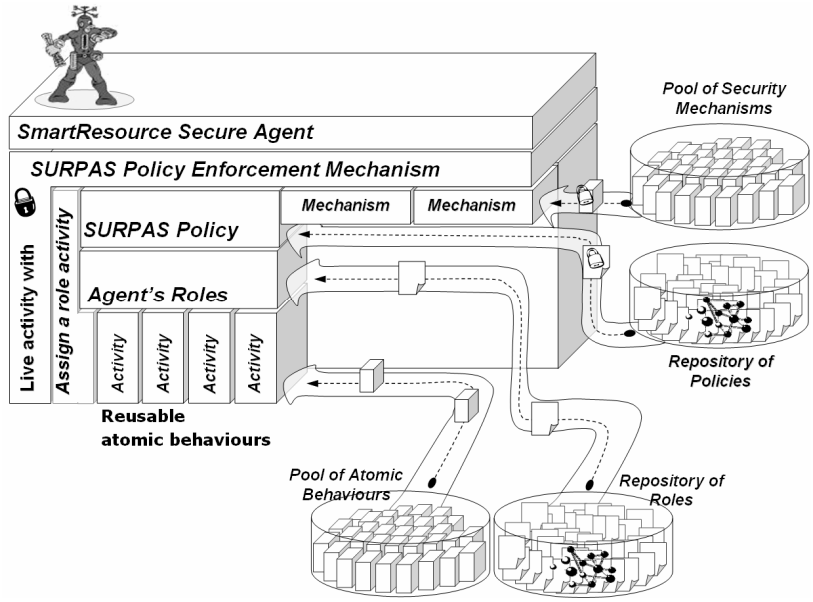


**Fig. 4.** Architecture of a secure SmartResource agent

The SURPAS policy enforcement mechanism manages security policies and security mechanisms. Its main task is to enforce security policies by interweaving with the Live activity. SURPAS policies are declarative descriptions using expressive and machine-interpretable data formats of Semantic Web. They are reusable over different agents, processes and organizations. Usually, SURPAS policies restrict actions prescribed by roles and enforce use of security mechanisms in addition to normal activities.

Agents access the roles, policies, security mechanisms, and RABs from external repositories, which are assumed to be managed by the organizations which own or hire the agents, or trusted authorities. It is done either upon startup of an agent, or if the organization requests an update to be made. Externalization of roles, policies, security mechanisms and RABs has several advantages:

- Increased flexibility for control and coordination. Namely, the organization can remotely affect the behavior of the agents through modifying the behavior roles and security policies.
- The roles, policies, security controls, RABs can always be kept up-to-date.

- There is a possibility to create self-configuring and self-protecting agents.
- Agents may 'learn' in run-time how to play a new role and how to follow a new security policy.
- Organizations are able to provide not only instructions what to do (declarative descriptions of roles and policies), but also the tools enabling doing that (RABs and security mechanisms).
- Agents may have a "light start" with on-demand extension of functionality.
- Inter-agent behavior and security awareness. The agents can make some use of the information about some roles and policies, even if they do not follow them. One reason is to understand how to interact with, or what to expect from, an agent playing those roles and following those policies.

In summary, the security components, which SURPAS introduces into the architecture of the SmartResource agent, are the policy enforcement mechanism that is built-in into the behavior engine, and security measures and security policies which can be either provided upon agent's startup or retrieved on demand.

In UBIWARE, we also envision some additional security related services, e.g. verifying and signing of roles, policies, security mechanisms and RABs by external trusted authorities to guarantee defect-free and proper behavior of agents. Regarding security, the beliefs storage of an agent has to supports following important activities: semantics-based logging and audit for proactive context-aware intrusion detection and non-repudiation, computing reputation for the management of trust relations between agents and security services, persistent storing of security contexts, and other.

## 7 Conclusions

Conventional approaches to manage and control security seem to have reached their limits in new complex environments. These environments are open, dynamic, heterogeneous, distributed, self-managing, collaborative, international, nomadic, ambient, and ubiquitous. New generation middleware such as UBIWARE will significantly advance the industrial automation towards automatic discovery, composition, orchestration, integration, invocation, execution monitoring, and coordination of industrial resources. These advanced automation techniques target physical world objects and thus put security as the core need-to-be-addressed issue. In this paper, we described our long-term vision for the security and privacy management in such complex environments, SURPAS. It aims at policy-based optimal collecting, composing, configuring and provisioning of security measures in multi-agent systems like UBIWARE. Particularly, we analyzed the security implications of UBIWARE, presented the SURPAS research framework which guides our research towards SURPAS, and the SURPAS abstract architecture.

There are an enormous number of targets for further work. They include ontology engineering for fundamental elements of security, elaborating architectures, designing new specific algorithms for the intelligent security policy management, developing reference implementations and, finally, adopting research ideas into practice in real-world industrial settings.

**Acknowledgements**. We are grateful to Tekes (National Technology Agency of Finland), Agora Center of the University of Jyväskylä, and cooperating companies (ABB, Metso Automation, TeliaSonera, TietoEnator, and Jyväskylä Science Park) for supporting activities of the SmartResource project. We are aslo grateful for the financial support to the Rector and to the Department of Mathematical Information Technology, University of Jyväskylä.

## References

[1] O. Kephart and D. M. Chess, (2003) The vision of autonomic computing, Computer, vol. 36, no. 1, pp. 41--50.

[2] Jennings, N.R. (2001) An agent-based approach for building complex software systems. Communications of the ACM 44(4): 35-41

[3] Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., and Mylopoulos, J. (2004) Tropos: An agent-oriented software development methodology. Autonomous Agents and Multi-Agent Systems 8(3): 203-236

[4] Berners-Lee, T., Hendler, J., and Lassila, O. (2001) The Semantic Web, Scientific American, Vol. 284, No. 5, pp. 34-43.

[5] Terziyan V., Katasonov A., Global Understanding Environment: Applying Semantic Web to Industrial Automation, In: J. Cardoso, M. Hepp, M. Lytras (eds.), Real-world Applications of Semantic Web Technology and Ontologies, Springer, 2007 (in press).

[6] Kaykova O., Khriyenko O., Kovtun D., Naumenko A., Terziyan V., Zharko A., (2005) General Adaption Framework: Enabling Interoperability for Industrial Web Resources, Int. Journal on Semantic Web and Information Systems, Idea Group, Vol. 1, No. 3, pp.31-63.

[7] Colin G. Harrison, David M. Chess, and Aaron Kershenbaum, Mobile Agents: Are they a good idea?, technical report, 1995, IBM Research Division.

[8] P. Horn, (2001) Autonomic computing: IBM's perspective on the state of information technology, IBM Corporation, Tech. Rep., 15 Oct. 2001. Available: http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf

[9] Jansen, W., (2000) Countermeasures for Mobile Agent Security, Computer Communications, Special Issue on Advanced Security Techniques for Network Protection, Elsevier Science BV.

[10] Jansen, W., and Karygiannis, T., (1999) Mobile Agent Security, National Institute of Standards and Technology, Special Publication 800-19, August 1999.

[11] Borselius, N., (2002) Mobile agent security, Electronics & Communication Engineering Journal, Volume 14, no 5, IEE, London, UK, pp 211-218.

[12] Naumenko, A. and Luostarinen, K., (2006). Access Control Policies in (Semantic) Service-Oriented Architecture, In Semantic Systems From Visions to Applications, Proc. of the SEMANTICS 2006, OCG, Vienna, Austria, pages 49-62.

[13] Satish N. Srirama and Anton Naumenko, (2007) Secure Communication and Access Control for Mobile Web Service Provisioning, (work in progress)

[14] Luostarinen, K., Naumenko, A., Pulkkinen, M., (2006), Identity and Access Management for Remote Maintenance Services in Business Networks, in IFIP International Federation for Information Processing, Volume 226, Project E-Society: Building Bricks, Springer, Boston, pp. 1-12.

[15] Naumenko, A., (2006) Contextual rules-based access control model with trust, In Shoniregan C. A. and Logvynovskiy A. (Eds.), Proceedings of the International Conference for Internet Technology and Secured Transactions, ICITST 2006, ISBN 0-9546628-2-2, e-Centre for Infonomics, pages 68-75.