A Context-Based Approach of Security Policies

Ghita Kouadri Mostéfaoui
Software Engineering Group
University of Fribourg, Switzerland
Ghita.KouadriMostefaoui@unifr.ch

Patrick Brézillon
LIP6
Université Paris 6, France
Patrick.Brezillon@lip6.fr

Abstract

The wide adoption of handled devices and remote services arises a number of security problems for users and services providers. The ubiquitous (pervasive) nature of such applications has brought new security vulnerabilities, and security in pervasive computing is thus, still a hot topic. Context-based security is an emerging approach for modeling adaptive security solutions based on the context of use of the system. Our contribution aims at presenting a new model for specifying context-based security levels. This approach is based on contextual graphs and relies on a set of contextual information collected from the system and user's environments.

Introduction

The pervasive computing paradigm allows the emergence of various applications in different domains, due to the new capabilities provided for such applications, such as invisibility (Weiser 1991) and that aims to the complete disappearance of pervasive computing technology from a user's consciousness (Satyanarayanan 2001), mobility of users and services, and users' devices and services heterogeneity, and adaptability. These new capabilities brought with them new types of security vulnerabilities whose frequency is dramatically increasing. Thus, new security solutions are required for which adaptation and reconfiguration in real time must be taken into account without darkening the network. The two features, reconfiguration and adaptation, are guided by the situation that surrounds both the requested service environment and the user's environment. The design of more adaptive systems tries to address context-awareness, but it is rather new in the domain of security. Context interests security by two observed phenomena. First, pervasive computing paradigm provides a plug-and-play facility that allows heterogeneous devices and resources to join or leave a network with a minimal effort. From a security perspective, this heterogeneity along with the different types of networks bring with it the non support of all required security mechanisms (as cryptographic protocols) and users' preferences. Future systems should be able to adapt their security mechanisms according to the type of devices and to the state of the network, and more generally the context of their use. Second, security systems developed now support a given and static framework, when attacks generally try to bypass these static contexts of effectiveness of security systems. A more secure infrastructure must be able to reconfigure itself at run-time in order to offer fine-grained control and to quickly detect and face new types of threats. Thus, the system now has difficulty to react at changes in context (not directly changes in actions) and suffers from a lack of remembering of past attacks in a CBR (Case-Based Reasoning) spirit. This paper describes a new approach called ”context-based security” that aims at designing adaptive security solutions and present the use of contextual graphs in determining the most appropriate security level for a pervasive application. The remainder of the paper is organized as follow. The next section presents our view of the concept of context-based security. The next section gives a brief overview of previous works on context-based security. The section after describes our approach, including a brief presentation of contextual graphs. The final section gives an evaluation of the proposed approach.

A Definition of Context-Based Security

As its name suggests, context-based security aims at introducing ”context” explicitly in the specification of security solutions (access control models, cryptographic protocols, etc). Context-based security emerged recently as a new approach to cope with the new types of security problems introduced by the high mobility of pervasive systems and the heterogeneity of devices used in these types of environments (Kouadri Mostéfaoui 2003). The idea behind context-based security is the following. The pervasive environment is initially controlled with a particular configuration of the security policy in an initial context. This context is continually changing in request to triggers (dynamic changes in the environment). The security policy must then adapt itself to the new context (security context) in order to fill the breaches that may be opened by the new context. By a security policy, we mean a specification that expresses clearly and concisely what security level to apply in each situation. A security level refers to the type of used cryptographic protocol, the way the user is authenticated, etc. This is what we call a security context.

Security Context

Kouadri and Brézillon (Kouadri Mostéfaoui & Brézillon 2003) made a preliminary attempt to define a security context. They state that a security context is a set of information collected from the user's environment and the application environment and that is relevant to the security infrastructure of both the user and the application. Thus, a security context corresponds to a set of information that requires taking a specific security decision for adapting the cryptographic protocol used in the communication, requiring a strongest authentication method or automatically denying access to a service when intrusion detection is triggered. This definition suggests the use of contextual information cues, such as user's identity, its interaction history with the service, his/her location, his/her preferences, type of requested service, time/date of request, exchanged information sensitivity, set of supported cryptographic protocols by the user's application and the service, along with a set of contextual information deduced by computation as the state of the network. For instance, if it is able to handle more user/service interactions by estimating the remaining network resources (CPU, bandwidth, etc) according to the number of already connected users.

Three types of Context

Context has an infinite dimension, but a modeling is however possible as shown in the approach followed in contextual graphs ((Brézillon & Pomerol 2000), (Brézillon 2002), (Brézillon 2003)), where the contextual elements are acquired incrementally when needed. Context has static and dynamic aspects that must be considered as intertwined and, thus, treated jointly. From a security point of view, an action undertaken by a user may leads to a change in the system and implies, as a consequence, a change in the security context. New security mechanisms must be enforced according to the new context. For dealing with the large number of contextual elements, (Pomerol & Brézillon 1999) distinguish between three types of context (Figure 1) for a given focus of attention, namely, external knowledge, contextual knowledge, and proceduralized context. The external knowledge is the knowledge that has nothing to do with the current focus. Conversely, the contextual knowledge is the knowledge potentially mobilizable at the current focus. The proceduralized context is the part of the contextual knowledge that is explicitly considered in the focus.

Context-Based Security Policies

Security policies are impossible to circumvent in specifying security requirements even for small or more complicated systems in terms of services they provide. According to (SANS 2001), a security policy establishes what must be done to protect information stored on computers. A well-written policy contains sufficient definition of ”what” to do so that the ”how” can be identified and measured or evaluated. A security policy imposes a set of requirements about the security infrastructure and defines which kind of mechanisms need to be implemented. Context-based security policies aim at considering context explicitly as a guide to deduce which security level to apply in a given situation. In the next section, we summarize the main works that consider context in security.

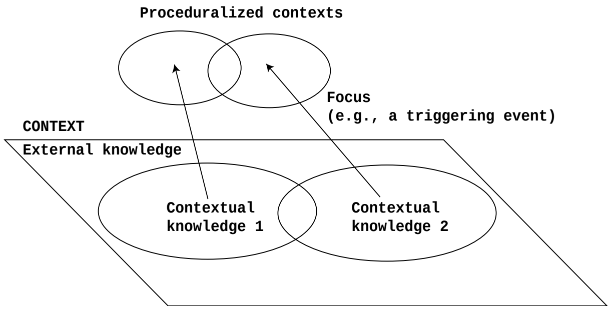

Figure 1: Relationships between the three types of context

Related Works and Motivation

Considering context in security is a recent research direction. Most of the efforts are directed towards securing context-aware applications. Covington's team explores new access control models and security policies to secure both information and resources in an intelligent home-environment ((Covington & Srinivasan 2001), (Covington & Ahamad 2002), (Covington & Abowd 2001). Their framework makes use of environment roles. In the same direction, (Masone 2002) designed and implemented RDL (Role-Definition Language), a simple programming language to describe roles in terms of context information. There have been similar initiatives in (Shankar & Balfanz 2002) and (Osbakk & Ryan 2002). It is interesting to observe that all previous work on combining security and context-aware computing follow the same pattern: using contextual information to enrich the access control model in order to secure context-aware applications with a focus on specific applications. However, the unique contextual elements considered in this approach are collected through sensors (e.g. location by GPS), but other contextual elements as the user's preferences are not really considered. The second main observation is that security decisions follow an old-fashioned rule-based formalism which does not consider systems and networks dynamics. The main problem here concerns the non consideration for the dynamic aspects of context. Kouadri Mostéfaoui and Brézillon (Kouadri Mostéfaoui & Brézillon 2003) propose a generic model for managing authorizations in a distributed environment. Their model offers a clear separation between the context handling process, the formalism that models the context-based policy, and the system to protect. In practice, this is equivalent to three main modules; the context bucket, the context engine and the distributed system to protect. This design choice allows security administrators to update the logic that secures the system with minimal efforts. The present work concerns an extension of this framework by using contextual graphs as a modeling tool which fits into the context engine.

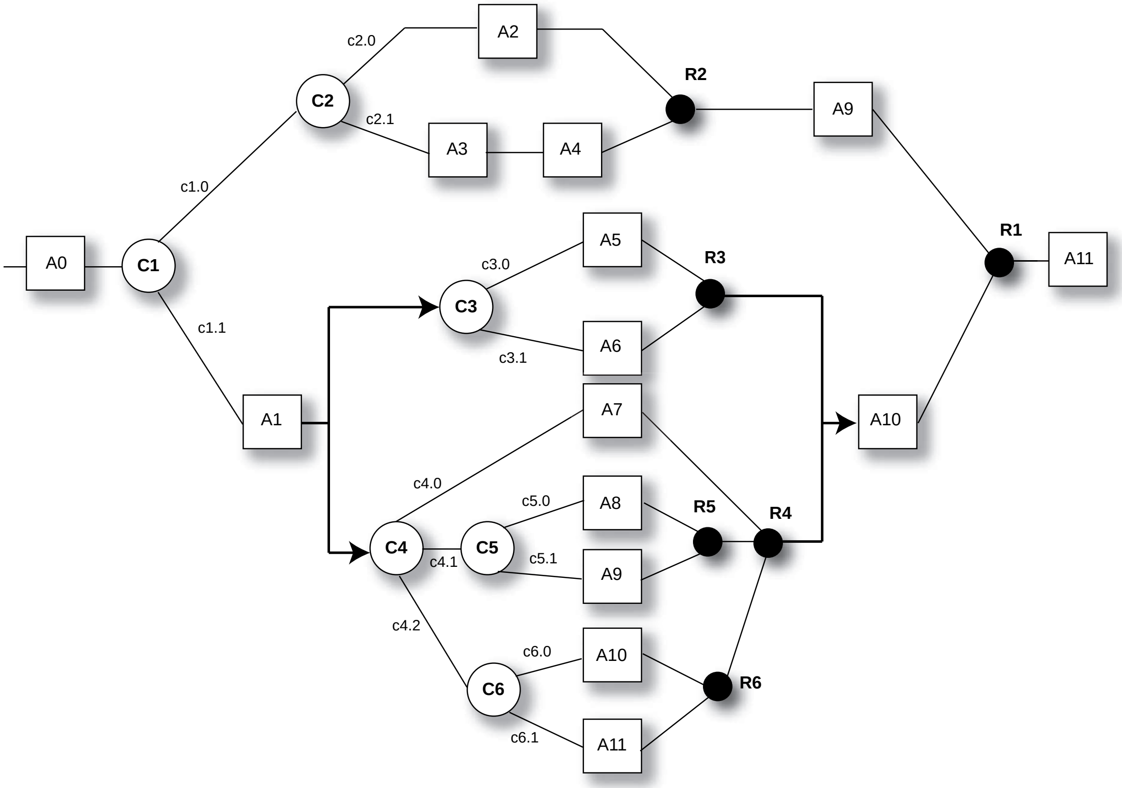

Figure 2: An example contextual graph

Modeling Security in Contextual Graphs

Even, if the rule-based representation is often adopted as an intuitive solution such as in (Covington & Ahamad 2002) and (Masone 2002), it suffers from three main limitations. The first one is the difficulty to maintain such formalisms in case of complex systems to secure. The second limitation is the difficulty to identify all the needed contextual information from the rule-based formalism which awkward the context management task. The third main limitation is that it does not provide a convenient way for understanding the followed strategy of the policy and makes the security management task cumbersome for security administrators. Decision trees are another way to structure the rules. However, the fine-grained nature of context leads to a combinatorial explosion of the trees size (Pasquier 2002). In order to get round these limitations, a new approach known at ”contextual graphs” has been explored (Brézillon 2003). This formalism would help in specifying context-based security policies and are used as a management tool that eases security administration for complex environments with many heterogeneous services and devices. Contextual graphs are inspired from decision trees, with two main differences. First, they have no decision node, only ”chance” nodes where a contextual element is analyzed in order to determine its value and to select the corresponding path. Second, there are no probabilities because each path corresponds to a well identified practice applied by (at least) one user. Contextual graphs have been initially designed for an application for incident solving on a subway line (Brézillon & Pomerol 2000), (SART 2004)). It is now the object of studies by its own (Brézillon 2003). A contextual graph is an acyclic directed graph with a unique input, a unique output, and a serial-parallel organization of nodes connected by oriented arcs (the whole CxG having then a structure of spindles). A node in a contextual graph can be an action, an activity (a particular sub-graph), a contextual node, a recombination node, or a parallel action grouping. Figure 2 presents a sample contextual graph where the Ai are the actions, Ci, the contextual nodes and Ri the recombination nodes. Parallel grouping are highlighted using thick arrowed links. A contextual graph (CxG for short) allows a context-based representation of a given problem solving for operational processes by taking into account the working environment (Brézillon 2003). In our case, they allow to treat security requirements as a problem solving process that allows only safe actions to be undertaken by the user as long as he interacts with the environment.

Security Management: An Example

As an example, we use contextual graphs to model the context-based security policy that manages access to a resource in a distributed environment. In our case, the input corresponds to the user entering into the environment. The output corresponds to the user leaving the environment with no security incidents on both the environment and the user as long as the user is connected. Figure 3 illustrates the

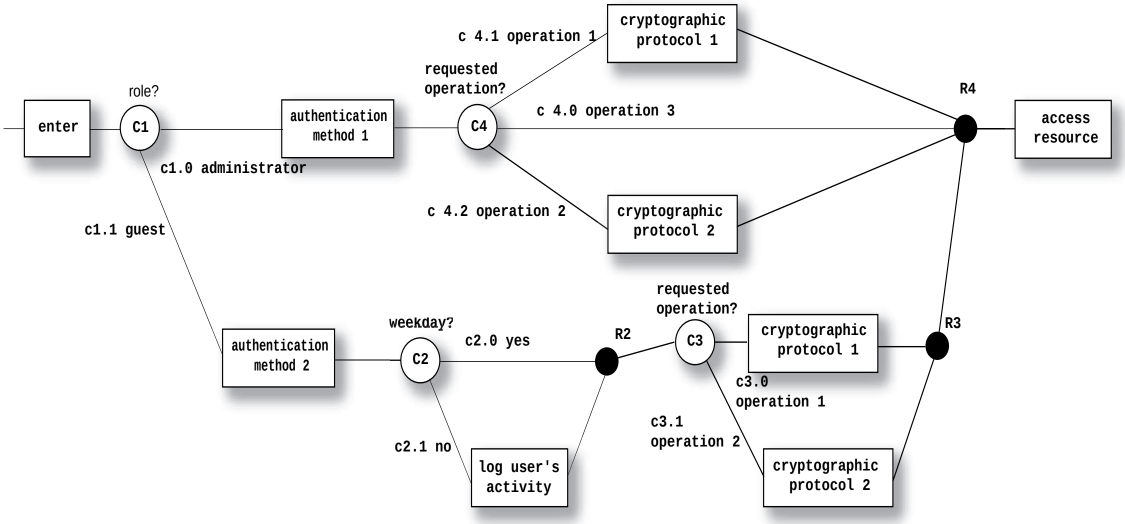

Figure 3: Contextual graphs-based security policy

context-based security policy that manages our distributed application. For sake of clarity, only a small part of the contextual graph is represented. In this example, a task corresponds to calling a method on a resource. The user wishing to access the resource first enters the distributed system that allows him to specify its role. Depending on this context (C1), a specific authentication method is applied. If the user is an administrator, the cryptographic protocol that will be set between the user application and the resource application relies on the type of the requested operation. If the user is a guest, an additional step is needed in order to know if its corresponding activity must be logged or not depending on the day of request (C2). As one can observe, for a guest user, only two operations are allowed for him to perform on the resource in contrast to the administrator. Not allowed operations are not shown on the contextual graph. This is to allow specifying only safe paths in order to perform a secure action and is commonly known as closed security policy ”which is not explicitly permitted is denied”. According to the user’s role, the time, and the type of requested operation, the contextual graph represents the decisions to undertake according to the current context. These decisions are security actions including authentication methods (code authentication, username/password, etc), cryptographic protocols and more specific operations such as to log user’s activity.

Elements of the Security Policy in CxGs

A detailed definition of each element of contextual graphs as a generic formalism is presented in (Brézillon & Pomerol 2000). In this paper, we describe each element according to our use. Namely, in order to specify the context-based security policy.

Security Actions. A security action is an executable method that aims at enforcing the policy at a given point of the CxG. In our example (Figure 3), authentication method1, and log user’s activity represent security actions.

Contextual Elements. A contextual element is represented by two types of nodes, namely a contextual node and a recombination node. A contextual node corresponds to the explicit instantiation of the contextual element. For example, a contextual element could correspond to the role of the requesting user; administrator or guest. For each contextual node Cn a set of exclusive branches corresponding to known practices emerges. The associated recombination node Rn corresponds to the abandon of the instantiation of the contextual element once the action on the branch is accomplished, for example R2. Then, there is a convergence of the different alternatives towards the same action sequence to execute after. Thus, at the contextual node, a piece of contextual knowledge becomes instantiated and enters the proceduralized context. At a recombination node, that last piece entered in the proceduralized context goes back to the contextual knowledge. Thus, a change in the context correspond to the movement of a piece of contextual knowledge into the proceduralized context, or conversely from the proceduralized context to the contextual knowledge.

Evaluation

Contextual graphs have been successfully used for incident solving for RATP, the subway company in Paris. They also have been used in medicine in order to represent the diagnosis of ischemy, based on thoracic pain. Contextual graphs have also been used in different domains of reasoning (interpretation, decision making, pattern recognition, etc.), where context impacts strongly. In security, contextual graphs constitute a promising approach for the modeling of context-based policies. They provide an understandable representation of security mechanisms to implement in each situation. Contextual graphs support incremental knowledge acquisition. The security administrator may easily add/modify secure paths based on new detected breaches. Thus, security policy has the capacity of evolving by accommodation and assimilation of practices. A potential scenario would be to consider a contextual graph as the whole set of types of connections, identify the sensible path in the CxG, and block immediately the user that creates a dangerous context, before to execute his action and penetrate the server. That is, the acquisition of a new practice corresponds to the addition in the contextual graph of the minimum number of elements (generally one pair contextual node - recombination node and an action). With a contextual graph representing the set of all the ways to reach a remote service, the system will be able to identify the way chosen by a user according to his actions, as any security system, but also from the contextual choices (i.e., the instantiation chosen at each contextual node) initiated by the user. Such a system will thus, be able to stop an attack if the user selects a path that is dangerous for the system security.

Conclusion

Even if context has been used since a while in policies specification, it is rarely considered explicitly. As a consequence very few works benefit from the theories and tools already developed in the context-aware computing area in order to model the needed contextual information. Contextual graphs provide a convenient way for specifying security requirements in pervasive environments, and can be used as a security management tool that eases the task of understanding and modifying the security policy. We are actually developing a tool that allows building and modifying contextual graphs-based policies graphically. The resulting application is implemented in Java. It offers a set of default actions such as, logging user's activity, and support incremental acquisition of practices (i.e., secure paths on the graph).

References

Brézillon, P. Pasquier, L., and Pomerol, J.-C. 2000. Reasoning with contextual graphs. *European Journal of Operational Research* 136(2):290–298.

Brézillon, P. 2002. Modeling and using context: Past, present and future. Technical report, LIP6, University of Paris 6, France.

Brézillon, P. 2003. Using context for supporting users efficiently. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03)*, 127.

Covington, M. Long, W. S. S. D. A. A. M., and Abowd, G. 2001. Securing context-aware applications using environment roles. In *Proceedings of the 6th ACM Symposium on Access Control Models and Technologies (SACMAT '01), Chantilly, Virginia*.

Covington, M-J. Fogla, P. Z. Z., and Ahamad, M. 2002. Context-aware security architecture for emerging applications. In *Proceedings of the Annual Computer Security Applications Conference (ACSAC), Las Vegas, Nevada*, 249–260.

Covington, M-J. Ahamad, M., and Srinivasan, S. 2001. A security architecture for context-aware applications. Technical report git-cc-01-12, College of Computing, Georgia Institute of Technology.

Kouadri Mostéfaoui, G., and Brézillon, P. 2003. A generic framework for context-based distributed authorizations. In *Proceedings of the Fourth International and Interdisciplinary Conference on Modeling and Using Context (Context'03)*, volume LNAI 2680, 204–217. Berlin: Springer Verlag.

Kouadri Mostéfaoui, G. 2003. Security in pervasive environments, what's next? In *Proceedings of the International Conference on Security and Management (SAM'03), Las Vegas, Nevada*, 93–96.

Masone, C. 2002. Role definition language (rdl): A language to describe context-aware roles. Technical Report TR2002-426, Dartmouth College of Computer Science, Hanover, NH.

Osbakk, P., and Ryan, N. 2002. Context privacy, cc/pp, and p3p. In *Proceedings of UBICOMP2002 Workshop on Security in Ubiquitous Computing*.

Pasquier, L. 2002. *Modélisation de Raisonnements Tenus en Contexte. Application à la Gestion d'Incidents sur une Ligne de Métro*. Ph.d thesis, Université Paris 6, France.

Pomerol, J.-C., and Brézillon, P. 1999. Dynamics between contextual knowledge and proceduralized context. In *Modeling and Using Context (CONTEXT-99)*, volume LNAI 1688, 284–295. Berlin: Springer Verlag.

SANS. 2001. Giac basic security policy. version 1.4, page 3.

SART. 2004. Sart project homepage, http://www-poleia.lip6.fr/∼brezil/sart/index.html, accessed february 2nd, 2004.

Satyanarayanan, M. 2001. Pervasive computing: Vision and challenges. In *IEEE Personal Communications*, volume 8, 10–17.

Shankar, N., and Balfanz, D. 2002. Enabling secure adhoc communication using context-aware security services. In *Proceedings of UBICOMP2002 Workshop on Security in Ubiquitous Computing*.

Weiser, M. 1991. The computer for the 21st century. *Scientific American* 3(265):66–75.