Towards Security Auto-Configuration for Smart Appliances

Jean-Marc Seigneur, Christian Damsgaard Jensen, Stephen Farrell, Elizabeth Gray, Yong Chen

Distributed Systems Group
Department of Computer Science
Trinity College, Dublin 2, Ireland.

secure-tcd@cs.tcd.ie

Abstract

Now that smart home appliances are easily plugged into smart home networks, the question of how to simplify security management, especially of access rights, for such appliances arises. The problem is aggravated by the fact that home users cannot be considered as "skilled" administrators, but are instead often technology-unaware users.

Establishing trustworthiness when it comes to secure smart appliances has been considered as a "holy grail" not met by any current technology. The SECURE project aims to develop security mechanisms based on human notions of trust, which may prove part of the solution. Trust-based security mechanisms allow access rights to evolve among previously unknown principals, thus minimizing security configuration.

This paper outlines the process of applying the SECURE project's security technology to smart home appliances with minimal user intervention.

1. Introduction

Weiser's vision of ubiquitous computing [1] will become true when computing capabilities will be woven into the fabric of everyday life. Currently, major companies in the household appliances market are getting increasingly involved in smart home appliances – appliances with communications capability [2]. Soon, a digital heartbeat will be embedded in many everyday appliances. These appliances will be used within home networks, which, in turn, will be permanently connected to the Internet.

The fact that the environment is the home is important, because the home environment differs fundamentally from the corporate environment. In a corporate environment, there is the assumption of the presence of an administrator. In home environments, no "skilled" administrator is present and most of the users are less than "fully" knowledgeable. Thus, the general requirement is that smart appliances should strive for auto-configuration wherever possible. Technologies have emerged in the past few years, which ease the installation and use of home networks and their attached smart devices, e.g. Universal Plug and Play (UPnP) [3] or the Open Service Gateway initiative (OSGi) [4]. However, once all appliances in a household are automated and connected through a network, it becomes essential to consider issues of security, especially access control. The issue of simplifying the management of smart appliances reappears, this time due to security management. In the smart home, access control has to be configured and managed by technology-unaware and busy householders, without the presence of a full-time dedicated system administrator.

Current security management solutions for smart appliance middleware lack such required auto-configuration. For example, UPnP products available in 2002 "tend to have been designed based on requirements of industry rather than of the home, making their administration difficult and sometimes assuming the existence of both physical security and a group of on-call support professionals" [5]. In this paper, we present the technology developed within the SECURE project [6] and show how it achieves implicit access control management, thus minimizing user intervention thanks to security mechanisms based on human notions of trust and entity recognition.

Section 2 describes the SECURE project and its applicability for smart home appliances is described in section 3 along with some preliminary results. Related work is presented in section 4. Finally, section 5 describes our conclusions and outlines future work in this area.

2. The SECURE project

The SECURE (Secure Environments for Collaboration among Ubiquitous Roaming Entities) project is investigating dynamic and self-configuring security mechanisms for global computing based on human notions of trust. This subsection gives an overview of SECURE. Others [7] present SECURE in greater details.

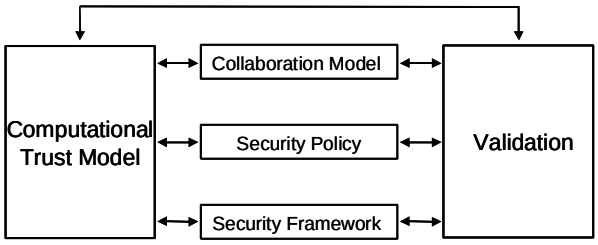

Figure 1: SECURE project overview.

The project has five different operational goals:

- Define a formal computational trust model
- Define a collaboration model (dynamic aspects of the trust model)
- Define means to specify security policies based on trust
- Develop a framework for trust management
- Validate the approach in the context of the formal model

Others have detailed how trust can be formalized as a computational concept [8-10]. The SECURE project provides a trust engine which can dynamically assess the trustworthiness of an entity based on the three sources of trust: observation, recommendation and reputation [7]. Another component of the SECURE framework is the entity recognition module, which allows recognition of previously observed/encountered entities. Usually, authentication is the first step to ensure security in computing environments but other work [11] discusses why traditional authentication should be revised for pervasive computing. The entity recognition process and end-to-end trust model [12] address this problem by recognition, which is a more general concept

than authentication, i.e. entity recognition encompasses authentication. We believe that the ability to recognise an entity is sufficient to establish the trustworthiness of that entity, i.e. recognition establishes a reference to the basis of trust, which is maintained by other components in the SECURE framework, e.g. the trust box maintains the current level of trust in the recognised entity and the trust formation component collects evidence, from the records of prior interactions with the entity, to support this level of trust. On the other hand, there is no belief that the name or authentication alone is sufficient to infer the proper property of trustworthiness. As soon as an entity has been through one cycle of the entire recognition process, interactions with this entity can happen and be taken into account by the security mechanisms based on human notions of trust engineered within the SECURE project. Based on interaction after interaction, trust will be associated with this entity, which is recognized thanks to the recognition module. Other sources of trust can also play a role in the evolution of the associated trust: recommendations or reputations.

3. SECURE applied to smart devices

Commonly, smart home middleware provides discovery services because it simplifies the installation of new smart appliances. These discovery schemes can be seen as recognition schemes which could be plugged into the SECURE Pluggable Recognition Module (PRM) [12], currently under implementation based on Java PAM [13, 14]. Most of these discovery schemes cannot be considered as "strong" recognition schemes. However, we can still use them in our recognition process since the PRM provides meta-data on the technical trust in the recognition scheme used. Indeed, the level of confidence can be taken into account in the final access control decision.

In fact, the main idea is to integrate a trust-formation element into the smart appliance access control mechanism to manage interaction between previously unknown users or smart appliances. Rather than having to set up an access control list for individual entities – other users or devices – or group of entities, the owner of a device only has to set up the level of trust required before interaction with an encountered entity can take place. The SECURE trust engine takes care of trust management – evolution of trust based on new observations, recommendations and reputations – on behalf of the user. In doing so, the access control at the entity level, which may be overwhelming for home users if done manually, is implicitly managed. It may be possible to ship the appliance with default trust configuration, e.g. based on common sense or trust brokering systems [15]. This would be a further step towards complete auto-configuration since no user intervention is required at all. The following diagram illustrates the use of SECURE inside a smart appliance. In this example, an access control matrix (ACM) is associated with every device. The ACM contains a list of assets, which is associated with the level of trust required to access each asset. Other components must be available in order to use SECURE: the trust store where each principal (appliance or user) is listed along with their trust information; the trust engine; and the PRM. All these components are considered local, inside the security perimeter of the appliance. However, they may have to be distributed due to resource constraints of the device. Although secure association with such distributed components is feasible, describing how this distributed architecture would be achieved is beyond the scope of this paper. Nevertheless, even locally, some components can be tailored for resource constrained devices: the PRM may provide only one recognition scheme rather than many; the set of entities kept in the trust store as well as the scope of their trust information could be limited by selective forgetting based on some algorithms under-development (e.g. the notion of forgetting in the recognition process [12]).

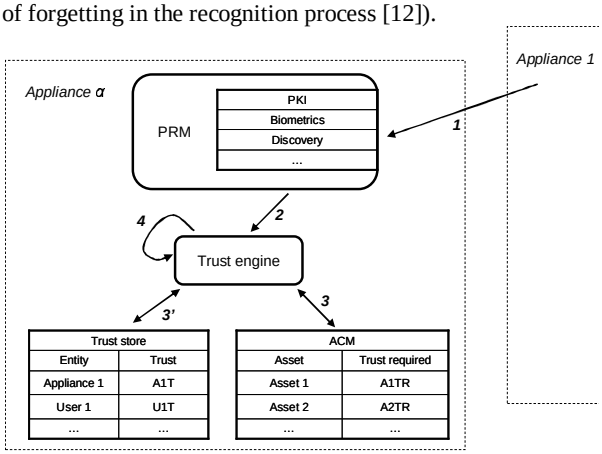

Figure 2: Proposed appliance with SECURE inside.

With such modules in smart home appliances, when a new appliance, appliance 1, comes into contact with appliance α, the PRM allows for dynamic enrollment, if home appliance discovery is used. Initial trust formation is associated with this appliance, for instance by means of recommendations or reputation. After that, if appliance 1 asks for access to asset 1, the following process described in figure 2 would be done:

1. Appliance 1 requests to use asset 1.
2. The PRM forwards the request to the trust engine along with meta-data on the recognition confidence.
3. The trust engine would get the trust required to access asset 1, i.e. A1TR; 3' is when the trust engine gets the trust in appliance 1, i.e. A1T.
4. Access is granted if the end-to-end trust [12] is higher than the trust required. This would mean: $f(A1T) > A1TR$ where $f(A1T) \leq A1T$ depending on the confidence in recognition. Calculation of trust must consider the trust in technical infrastructure (e.g. based on the confidence in recognition) and trust in the requesting entity, i.e. appliance 1.

To make this more concrete, consider the following scenario. A new smart lock is unpacked and positioned on the front door of a smart home by the owner. The smart lock, when unpacked, immediately recognizes the owner thanks to biometrics schemes along with the fact that this is its first installation on a door (i.e., this is a variant of the resurrecting duckling secure association process [16]). After a few minutes, the owner is recognized as the owner and the lock sets the trust to high. In the mean time, the smart home video monitoring system acknowledges that the owner has just installed a new lock on the front door. The smart home discovery mechanism advertises the new identifier of the new lock and, after data triangulation, the video sets high trust in the new lock because the owner has installed it. The lock on the kitchen door discovers this new smart lock. The kitchen door lock was set up with middle trust. After seeing that the smart home video monitoring system, with which it has built high trust over time, recommends that it can get

recommendations from this new smart lock, the kitchen door will highly consider recommendations from this new lock on the front door although it maintains autonomy over its final decision. So, the kitchen sets initially high trust in the new lock. Later on, the owner leaves the home and, while away, gives a key to one of his old friends who is in town for the first time. This old friend arrives alone in front of the front door because the owner has to work late. The smart lock on the front door, even if it does not recognize the old friend, accepts to let him in because she has the key. First, she tries to get into the kitchen searching for refreshments. The kitchen door lock does not recognize her since it is their first encounter. The kitchen door lock starts building initial trust for the friend and asks for a recommendation from the smart lock on the front door. The reply says that the friend (recognized by biometrics) has the key to open the front door and should be considered trustworthy. Nevertheless, this is not enough: the kitchen lock's trust in the friend is set below middle trust but above low trust. Thus, the kitchen lock chooses not to open. Then, the friend tries to switch on the smart TV, which is set to low trust. The trust formation process for the TV is the same as for the kitchen lock. So, after contacting the front door lock for a recommendation, the TV allows the friend to watch TV. The TV's trust in the friend is set above low trust for the same reasons as for the kitchen lock, but low trust is enough to switch on the TV. While watching TV, the phone rings. For the same reasons as for the TV, the friend is allowed to answer the phone: it is the owner and they discuss briefly. Thanks to voice recognition, the smart phone infers that the owner discussed with the friend. After watching TV for two hours, the friend is really thirsty so she decides to try to get into the kitchen again. The kitchen lock has, over time, built high trust in the TV and the phone in the same way as with the video monitoring system. This time the kitchen lock agrees to let the friend in since the smart TV and the smart phone recommend the friend based on two hours of interaction with the TV and more importantly one discussion with the owner over the phone. This is enough to increase the kitchen lock's trust in the friend above middle trust. Indeed, several good recommendations from other highly trusted peers increase the previous trust.

The approach, known as trust-based admission control has been assessed in the context of collaborative ad hoc applications. A distributed blackjack card game, where each player in a home may use different tablet computers, implements the trust-based admission control system to assign roles to users according to their trust-based admission rights [17]. It shows that trust evolves similarly to human trust and permits access to resources using a range of human like restrictions. Indeed, the bad guys can be identified and lose their rights without users intervention. The results are consistent with human intuitions about trust.

4. Related work

As mentioned in the introduction, different smart home technologies exist, which eases the federation and management of smart appliances. Some of these technologies, such as Jini [18] or UPnP, did not consider the home network to be connected permanently to the Internet, at least in the first versions. Others took straight into the account the presence of a gateway to the Internet at the edge of the home network, such as OSGi or JXTA [19].

Java Intelligent Network Infrastructure (Jini) offers high-level services, such as service discovery and transactions. It has been intensively used in research projects for smart home management [20]. However, it requires improvement concerning security features, e.g. to provide a security policy for dynamically granting permissions at runtime or to support non-uniform trust relationships, varying by client and server. Its tight relation with Sun's RMI implementation makes other choices more appealing.

JuXTApose (JXTA) is a network programming and computing platform based on the peer-to-peer paradigm. The aim of the JXTA platform is to allow implementation on any smart device. "JXTA peers operate in a role-based trust model, in which an individual peer acts under the authority granted to it by another trusted peer to perform a particular task" [21]. Interestingly, the security group of the JXTA project has chosen to use the concept of trust to deal with security issues: Poblano [22] is a kind of decentralized trust model implemented on a peer-to-peer topology built on top of JXTA. The trust spectrum neither requires nor prohibits the presence of a PKI (Public Key Infrastructure). They have started two applications based on this model: a reputation guided searching tool and a recommendation system for security purpose. To calculate the cooperation threshold between two peers, they have chosen a user-centric approach. The user interaction is required to tune the software and to make choices. Access control based on direct observations is not the main aspect of the work, which is still under-development.

OSGi is quite far from our approach since it consists of a gateway between the Internet and the home network, which is accessed remotely by providers managing services and smart appliances in the home. This solution is centralized and the user is still involved in local configuration of devices even if the gateway itself needs no configuration. Another gateway-based solution added Access Control Lists (ACLs) to smart devices using XML configuration files [20]. In doing so, the granularity of configuration is increased; access rights can be controlled at the level of the device or its sublevels at the level of individual functions.

ACL is also the choice of the new UPnP Security [5]. However, "it has the disadvantage of requiring a great deal of ACL editing if there are a large number of ACLs or a large number of subjects" [5]. To be fair, it also provides ways to decrease user intervention by various means: authorization server, authorization certificate, group definition certificate and delegation. Roughly speaking, the security is provided by key distribution and PKI.

Different existing trust management systems [23-25] are based on key infrastructures. As seen above, they can be a part of a security solution for smart appliance technologies. Nevertheless, SECURE aims at building a more fine-grained and dynamic trust-based solution: "the implicit, coarse and static view of trust in current systems fails to model the notion of trust, as human intuition understands it. A dynamic model of trust will provide the ability to operate and make decisions autonomously" [26].

5. Conclusion and future work

Ellison, the architect of UPnP Security, claimed that "unfortunately we have no sure means of establishing trustworthiness when it comes to security" [5] before describing a work around for this problem, in order to ease the management of security in UPnP. Other smart appliances technologies such as JXTA or Jini are also looking at trust for solving this problem. The SECURE project is aiming to

establish trustworthiness when it comes to security. This paper describes the process of applying the SECURE technology to securing smart home environments with minimal user intervention.

After describing what SECURE is really about, i.e. a security framework where access control rights evolve automatically and according to the human notion of trust, an approach for integrating this technology into smart appliances is shown. The first evaluation of this integration into a simple application, which could be deployed on home tablet computers, shows how access rights can evolve without user intervention, according to the human notion of trust, and how "bad guys" are automatically detected and punished by diminished access rights.

The implementation of the SECURE formal trust model has to be evaluated in the same way as the trust engine used in the blackjack application and support for recommendations has to be added to the current prototype and evaluated. Currently, only personal observations are taken into account. Moreover, the PRM has to be integrated. The smart home environment is attractive for evaluating the integration of weak recognition schemes – potentially the different discovery schemes of the smart home technologies. This will also be a step forward for smart home technologies, working towards a solution for establishing trustworthiness.

This work is sponsored by the European Union, which funds the IST-2001-32486 SECURE project.

6. References

[1] M. Weiser, "The Computer for the 21st Century", Scientific American, 1991, http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html.

[2] A. Kung, "E-protection of appliances through secure and trusted access", White paper, E-pasta consortium, 2001, http://www.trialog.com/e-pasta/publications/Whitepaper.pdf.

[3] UPnP, "Universal Plug and Play", Website, www.upnp.org.

[4] "OSGi Service Platform Release 2 Specification", The Open Services Gateway Initiative, 2001, http://www.osgi.org/resources/docs/spr2book.pdf.

[5] C. M. Ellison, "Home Network Security", in *Interoperable Home Infrastructure*, vol. 6 (4), Intel Technology Journal, 2002, http://www.intel.com/technology/itj/2002/volume06issue04/vol6iss4_interoperable_home_infrastructure.pdf.

[6] SECURE, "Secure Environments for Collaboration among Ubiquitous Roaming Entities", Website, http://secure.dsg.cs.tcd.ie.

[7] E. Colin, P. Nixon, S. Terzis, A. McGettrick, and H. Lowe, "Security Models for Trusting Network Appliances", in *the 5th IEEE International Workshop on Networked Appliances*, 2002.

[8] S. Marsh, "Formalising Trust as a Computational Concept", PhD Thesis, Department of Mathematics and Computer Science, University of Stirling, 1994, http://citeseer.nj.nec.com/marsh94formalising.html.

[9] A. Jøsang, "The right type of trust for distributed systems", in *Proceedings of the 1996 New Security Paradigms Workshop*, ACM, 1996.

[10] A. Abdul-Rahman and S. Hailes, "A Distributed Trust Model", in *Proceedings of the New Security Paradigms Workshop*, ACM, 1997.

[11] S. Creese, M. Goldsmith, B. Roscoe, and I. Zakiuddin, "Authentication for Pervasive Computing", in *Proceedings of the First International Conference on Security in Pervasive Computing*, 2003.

[12] J.-M. Seigneur, S. Farrell, C. D. Jensen, E. Gray, and Y. Chen, "End-to-end Trust Starts with Recognition", in *Proceedings of the First International Conference on Security in Pervasive Computing*, 2003.

[13] C. Lai, L. Gong, L. Koved, A. Nadalin, and R. Schemers, "User Authentication and Authorization in the Java Platform", in *Proceedings of the 15th Annual Computer Security Application Conference, Phoenix*, 1999.

[14] V. Samar and C. Lai, "Making Login Services Independent of Authentication Technologies", Sun Microsystems, 1995, http://java.sun.com/security/jaas/doc/pam.html.

[15] J. M. Seigneur, J. Abendroth, and C. D. Jensen, "Bank Accounting and Ubiquitous Brokering of Trustos", in *7th Cabernet Radicals Workshop*, 2002, http://citeseer.nj.nec.com/seigneur02bank.html.

[16] F. Stajano and R. Anderson, "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks", in *Proceedings of the 7th International Security Protocols Workshop*, 1999.

[17] E. Gray, P. O'Connell, C. D. Jensen, S. Weber, J. M. Seigneur, and Y. Chen, "Towards a Framework for Assessing Trust-Based Admission Control in Collaborative Ad Hoc Applications", Technical Report, Trinity College Dublin, 2002, http://www.cs.tcd.ie/publications/tech-reports/reports.02/TCD-CS-2002-66.pdf.

[18] Jini, Sun Microsystems, Website, www.jini.org.

[19] JXTA, Website, www.jxta.org.

[20] J.-M. Seigneur, "HOUSe-KEEPER, a vendor-independent architecture for easy management of smart homes", Technical Report, Trinity College Dublin, 2001, http://citeseer.nj.nec.com/seigneur01housekeeper.html.

[21] JXTA, "Project JXTA: Java Programmer's Guide", Sun Microsystems, 2001, http://www.jxta.org/project/www/jxtaprogguide_final.pdf.

[22] R. Chen and W. Yeager, "Poblano, A Distributed Trust Model for Peer-to-Peer Networks", Sun Microsystems, 2001, http://www.jxta.org/docs/trust.pdf.

[23] M. Blaze, J. Feigenbaum, and A. D. Keromytis, "Keynote: Trust Management for Public-Key Infrastructures", in *Proceedings of the Security Protocols International Workshop*, 1998.

[24] C. M. Ellison, "SPKI requirements", in *RFC 2692*, IETF, 1999, ftp://ftp.isi.edu/in-notes/rfc2692.txt.

[25] P. R. Zimmermann, "The Official PGP User's Guide", ISBN 0-262-74017-6, MIT Press, 1995.

[26] E. Colin, P. Nixon, S. Terzis, A. McGettrick, and H. Lowe, "Dynamic Trust Models for Ubiquitous Computing Environments", in *Ubicomp'02 Security Workshop*, 2002, http://www.teco.edu/~philip/ubicomp2002ws/organize/paddy.pdf.