Chapter 23
Developing Secure, Unified, Multi-Device, and Multi-Domain Platforms:
A Case Study from the Webinos Project

Andrea Atzeni
Politecnico di Torino, Italy

John Lyle
University of Oxford, UK

Shamal Faily
University of Oxford, UK

ABSTRACT

The need for integrated cross-platform systems is growing. Such systems can enrich the user experience, but also lead to greater security and privacy concerns than the sum of their existing components. To provide practical insights and suggest viable solutions for the development, implementation, and deployment of complex cross-domain systems, in this chapter, the authors analyse and critically discuss the security-relevant decisions made developing the Webinos security framework. Webinos is an EU-funded FP7 project, which aims to become a universal Web application platform for enabling development and usage of cross domain applications. Presently, Webinos runs on a number of different devices (e.g. mobile, tables, PC, in-car systems, etc.) and different Operating Systems (e.g. various Linux distributions, different Windows and MacOSx versions, Android 4.x, iOS). Thus, Webinos is a representative example of cross-platform framework, and even if yet at beta level, is presently one of the most mature, as a prototype has been publicly available since February 2012. Distilling the lessons learned in the development of the Webinos public specification and prototype, the authors describe how potential threats and risks are identified and mitigated, and how techniques from user-centred design are used to inform the usability of security decisions made while developing the alpha and beta versions of the platform.

DOI: 10.4018/978-1-5225-3422-8.ch023

INTRODUCTION

People use multiple devices with different form factors every day. These devices provide access to similar services but in different ways - native apps, Websites, mobile-specific Websites, etc. As such, these devices are interacting with each other more often, either to synchronize data or to provide cross-device user experiences, e.g., using a smart phone as a remote control for a smart TV, or having a companion application to a live TV programme. These new activities, scenarios and cross-domain user experiences require greater communication and increase the potential for misuse.

For example, *Gloria likes to personalize her online experience by setting application preferences, but also for privacy reasons she retains separate online identities. Gloria may be used to adopting a mobile device for one identity and a laptop for another, each of which covers two separate contexts. With smart systems and identity providers both available, Gloria's device may switch from one identity to another, but Gloria may be unaware of this switch if she set up her device to move between services without any intervention. In fact, she may not be aware which identity is exposed unless her activities are such that she would be conscious of an identity switch*.

Every different device may make a different trade-off considering authentication, authorization, and usability. For example, some devices may only infrequently ask the user to authenticate in order to minimize the use of a small keyboard or screen. However, when devices are used together, their different settings may conflict and either harm the user experience or reduce the system's security.

Security control can introduce usability problems (Schneier, 2009) as configuring and then using complex security features, like access control systems, can be difficult, time-consuming and fundamentally at odds with the primary goals of the end user. As each new platform may have a different system and interface for doing this, the access control problems in cross-device systems are magnified.

How security problems can be addressed in such a complex scenario without losing focus on the usability of the system is the topic of this chapter. The chapter describes a case study in multi and cross-device access control based on the *Webinos* project. The *Webinos* project has designed and implemented a cross-platform application environment which allows developers to create applications which can communicate seamlessly between each platform. This includes the development of a personal device network (Niemegeers and Heemstra de Groot, 2002) which attempts to solve many of the related problems. User-centred design techniques are one of the most important points in our approach. Users are personified as specific entities (like Gloria), with skills, attitudes and motivations, to avoid talking about generic users" who might become contradictory when based on solely on the imagination of developers.

This chapter is structured as follows: section 2 introduces the *Webinos* project and gives a high-level technical overview, as well as listing desired goals, implementation details and the most important concepts related to the *Webinos* architecture. Section 2 also introduces related work on similar architectures to inform our security framework. Section 3 explains in detail how we approached the key usability problems. Section 4 states the main threats we identified in the cross-domain *Webinos* platform. Section 5 introduces the security of the *Webinos* execution environment. Section 6 describes the different types of authentication mechanisms introduced in *Webinos*. Section 7 highlights how to secure a communication session to avoid confidentiality and integrity losses. Section 8 addresses the core of the access control system: the policy framework. Section 9 approaches another task performed as part of *Webinos*: the security analysis of the APIs introduced in *Webinos* so far. Section 10 briefly describes the need for a secure storage to keep confidential information in our cross-device system. Section 11 finally discuss our findings in a multi-platform system and draws conclusions.