Chapter 42
Leveraging UML for Access Control Engineering in a Collaboration on Duty and Adaptive Workflow Model that Extends NIST RBAC

Solomon Berhe
University of Connecticut, USA

Steven A. Demurjian
University of Connecticut, USA

Jaime Pavlich-Mariscal
Pontificia Universidad Javeriana, Colombia

Rishi Kanth Saripalle
University of Connecticut, USA

Alberto De la Rosa Algarín
University of Connecticut, USA

ABSTRACT

To facilitate collaboration in emerging domains such as the Patient-Centered Medical Home (PCMH), the authors' prior work extended the NIST Role-Based Access Control (RBAC) model to yield a formal Collaboration on Duty and Adaptive Workflow (CoD/AWF) model. The next logical step is to place this work into the context of an integrated software process for security engineering from design through enforcement. Towards this goal, the authors promote a secure software engineering process that leverages an extended Unified Modeling Language (UML) to visualize CoD/AWF policies to achieve a solution that separates concerns while still providing the means to securely engineer dynamic collaborations for applications such as the PCMH.

DOI: 10.4018/978-1-7998-3016-0.ch042

INTRODUCTION

With the increase usage of information technology in organizations and businesses during the past two decades, one of the main concerns was the scalable protection of systems and data against unauthorized user access. This led to the development of many access control models, such as the mostly adapted Role-Based Access Control Model (RBAC), formalized in 1992 and standardized by the National Institute of Standards and Technology (NIST) in 2000 (Sandhu, Ferraiolo, & Kuhn, 2000). Many information technology companies (IBM, Sybase, Secure Computing, Siemens, Microsoft, etc.) since then integrate NIST RBAC into their software for access control. In 1992, when RBAC was formulated the authors were mainly having standalone, offline, or local area network systems and software in mind, which are operated by many systems and users. Over the past decade software and systems are connected with each other at scale. Since 2007 with the usage of mobile internet capable devices the number of connected software and systems has even more multiplied. Since 2010 with the increase of Bluetooth connected devices, the Internet of Things (IoT) is projected, in which not only users and computers, but anything can be connected to everything. In many domains and industries (health care, logistics, sale, scheduling, etc.), this has an impact towards how tasks are performed, how workflows are re-designed, and how more and more tasks are completed by software and systems.

This trend leads to the hypothesize that traditional access control models, that focus on prohibiting access to systems, software, and data do not match requirements that emerge through the increased connectivity. We hypothesize that traditional access control models must be extended with collaboration models that obligate team-based access to systems and data in a coordinated manner. In our previous work we refer to this as Collaboration on Duty and Adaptive Workflow (CoD/AWF) model (Berhe, Demurjian, & Agresta, 2009). In the health care domain for example, using non-CoD/AWF based software may lead to forgetting or skipping important tasks, performing tasks without notifying or checking in with related users, teams, systems, and regulations, and non-conforming to health care standards. This may ultimately may increase both, the likelihood of unsound patient care and increased costs. In particular, to facilitate collaboration in the patient-centered medical home (PCMH), our prior work extended the NIST role-based access control (RBAC) model to yield a formal collaboration on duty and adaptive workflow (CoD/AWF) model. As a result, domains that require software which is accessed by users, software, and systems (across organizations), has standards and regulations, workflows, and team-based collaboration should consider incorporating CoD/AWF from an early stage on in the software engineering lifecycle. The next logical step is to place this work into the context of an integrated software development process for security engineering from design through enforcement. Towards this goal, we promote a secure software engineering process that leverages an extended unified modeling language (UML) to visualize CoD/AWF policies to achieve a solution that separates concerns while still providing the means to securely engineer dynamic collaborations for applications such as the PCMH.

Towards this goal, in this chapter we present in Section 2 background work on CoD/AWF, previous work on integrating NIST RBAC into UML, and a health care domain use case scenario. Section 3 will present the newly formulated CoD/AWF UML slice diagrams for teams, access control, obligation, and coordination. Section 4 will have a short discussion with concluding remarks.