DOI: 10.7250/eunis.2013.028

Event Driven Identity Management Systems

Tero Hakkarainen[1], Lauri Viitanen[2]

[1]Software Engineer, Helsinki Metropolia University of Applied Sciences,
tero.hakkarainen@metropolia.fi
[2]Software Engineer, Helsinki Metropolia University of Applied Sciences, laurivii@metropolia.fi

Keywords
Identity, Entity, Role, Event based, IAM, IDM.

1. INTRODUCTION

Businesses typically have several services where employees may log in using some centrally provided credential(s). However, those credentials are usually system specific and stored separately in different records - one for each system. When one of the attributes (e.g. name or title) common to some credentials changes, it should be updated everywhere to match. Doing this in batch mode e.g. once a day has at least the following drawbacks:

- Creates delay (credentials are out of sync until batch is run)
- Unnecessary processing of identities that didn't change
- High system load while identities are being synchronized

Event driven identity management trades all these flaws to increased complexity. In Helsinki Metropolia University of Applied Sciences such system, an in-house developed software named Amme (translates literally into "basin, tub"), is being used.

The purpose of this paper is to present an overview of the software architecture and behaviour of Amme as an example of an event driven identity management system. This paper also studies other parts of IAM (Identity and Access Management) architecture and support processes in Metropolia e.g. software systems where identity information is originated, directories where data is populated, custom interfaces for software systems and single-sign-on - architecture.

2. SYSTEM ARCHITECTURE FOR IAM

The IAM system architecture in Metropolia consists of two main information sources for identities: Atbusiness HRM is primary data source for employee identities. CGI Winha is primary data source for student identities. Identities which are out of employee or student role scope are managed completely in Amme. User logins in Microsoft Windows desktop environment are handled with Microsoft Server Active Directory. LDAP directory can also be used for user authentication. Single-sign-on sessions for web-environments are managed with Jasiq CAS.

Amme is integrated with several custom built interfaces with different software systems. Interfaces can be made for example as database view based, using web services or using traditional interface files. The best possible connection method is chosen per interface basis.

3. ARCHITECTURE OF AMME

Amme is a central hub in a star model. It receives or queries information from source systems (data sources) and builds a single identity from the received “partial identities” also known as “roles”. The resulting identity of a real person is then sent to every destination system (data target), some of which were source systems as well. The most important data sources and targets are listed in the previous chapter.

Each of them is represented by a unique Java bean that knows how to receive or gather (usually by polling) roles from that system. Data targets can be security and access control systems, external records or any of the data sources. These are beans that know how to send the identity of a person to that system specifically. The active beans are selected and their properties set via Spring XML

DOI: 10.7250/eunis.2013.028

configuration file(s). This architecture causes Amme to be very adaptable to changes in the data topology.

4. PROCESS FOR CREATING IDENTITIES

The identity management process in Metropolia is contract based. For any right of use there should be some form of contract connecting the user and user's right of using a software system. Contracts can be created automatically based on user's role in Metropolia. Some contracts are created by request and some of them also require acceptance from the person responsible for the software system.

The identity management process in Metropolia can be simplified in the following process model:

1. Person makes contract with Metropolia. A contract can be:
    a. Contract of Employment
    b. Acceptance Letter for a Student
    c. Other form of connection with Metropolia e.g. partnership.
2. User is created in the master data system:
    a. Atbusiness HRM for employees
    b. CGI Winha for students
    c. Amme for other roles
3. If user's dataset is valid, user information is automatically imported to Amme through data import interface.
4. Amme runs the required calculations and
    a. returns generated information to the master data systems
    b. populates calculated data to Amme database
    c. synchronizes identity data with directories and data system targets

Changes in contracts in the master data systems are handled with similar data update process.

Contract based identity management requires also life cycle management of identities: when a contract has ended, Amme has to take care of proper deprovisioning on user rights in every connected software system.

5. EVENT BASED IDENTITY MANAGEMENT

An identity management event is triggered when a student or employee entry is modified or created in their respective management systems. The interfaces to those systems are being polled by Amme and once the new/updated identity fragment i.e. role is noticed, it is pulled into Amme.

The role is either associated with its existing full identity or if none exists, a new identity is created. The fields of the full identity are refreshed to match with the data in the received role. Finally the new/updated identity is pushed into every target system, which will update their respective information about the person in question to match that of the identity.

6. AUTHOR BIOGRAPHIES

Tero Hakkarainen

Tero Hakkarainen has Bachelor of Business Administration degree in Information Technology from Laurea University of Applied Sciences (2004) and Bachelor of Business Administration degree in Business Administration from Metropolia University of Applied Sciences (1996). He has worked in Metropolia since 2009 first as IT Systems Designer and from 2012 as Software Engineer. His work is divided between working in projects as a specialist and working as a developer in software development projects. His work is focused in Financial Accounting and Personnel Management.

Before Metropolia Tero worked as an IT Consultant in Oy Esdata Ab (2001-2008). In Esdata he worked with customers to solve their problems in everyday computing and process problems. He developed software solutions from planning to actual code.

DOI: 10.7250/eunis.2013.028

Lauri Viitanen

Lauri Viitanen graduated from Helsinki Metropolia University of Applied Sciences in spring of 2012 as Bachelor of Applied Science. He specialized in software design and development of operating systems and drivers.

During his studies in 2010-2011 he worked as a trainee programmer in Envault Corporation. Work involved testing and developing company's automated flash drive (de)crypting kernel module and associated software. After graduating he has been the lead Java EE programmer in the IT administration department of Metropolia University of Applied Sciences.