Guide

Architectures for Identity Management

Prepared by:
Christopher J. Harding,
Roger K. Mizumori,
& Ronald B. Williams

THE Open GROUP
Making standards work®

The views expressed in this Guide are not necessarily those of any particular member of The Open Group.

Guide: Architectures for Identity Management

ISBN: 1-931624-76-3

Document No.: G072

Published by The Open Group, March 2007.

Any comments relating to the material contained in this document may be submitted to:

ogpubs@opengroup.org

contents

About The Open Group

The Open Group is a vendor-neutral and technology-neutral consortium, whose vision of Boundaryless Information Flow™ will enable access to integrated information within and between enterprises based on open standards and global interoperability. The Open Group works with customers, suppliers, consortia, and other standards bodies. Its role is to capture, understand, and address current and emerging requirements, establish policies, and share best practices; to facilitate interoperability, develop consensus, and evolve and integrate specifications and Open Source technologies; to offer a comprehensive set of services to enhance the operational efficiency of consortia; and to operate the industry's premier certification service.

Further information on The Open Group is available at www.opengroup.org.

The Open Group has over 15 years' experience in developing and operating certification programs and has extensive experience developing and facilitating industry adoption of test suites used to validate conformance to an open standard or specification.

The Open Group publishes a wide range of technical documentation, the main part of which is focused on development of Technical and Product Standards and Guides, but which also includes White Papers, Technical Studies, and Business Titles.

A catalog is available at www.opengroup.org/bookstore.

Trademarks

Boundaryless Information Flow™ and TOGAF™ are trademarks and Making Standards Work®, The Open Group®, and UNIX® are registered trademarks of The Open Group in the United States and other countries.

All other brand, company, and product names are used for identification purposes only and may be trademarks that are the sole property of their respective owners.

Acknowledgements

The Open Group gratefully acknowledges the leadership of Ron Williams (IBM-Tivoli), Roger Mizumori (Waterforest Consulting), and Chris Harding (The Open Group) in writing the initial drafts of this Guide; Bob Blakley (was IBM-Tivoli, now The Burton Group) in restructuring it and providing additional material; and other members of The Open Group Security Forum and Identity Management Forum in reviewing successive drafts.

Chapter 1

Introduction

The purpose of this document is to guide the enterprise architect in the process of developing an enterprise identity management architecture.

An *identity* may be thought of as a set of information that characterizes a person or thing. It may include a name, an address, or descriptive information such as size or weight.

There may be information associated with an identity that is significant because of the context within which the person or thing is considered. This may include status, privileges, or role within the context. There may also be a cachet associated with some of these traits, based on their value to the context. These describe a *profile* of the person or thing that has a certain relevance to that context.

A *context* in this sense can be a country, a company, or a private association such as a golf club. What it means to belong to a community can be defined strictly: as citizenship of a country, employment by a company, or membership of a golf club. Or it can be defined loosely: as people that speak a language, customers and potential customers of a product supplier, or fans of a sports team.

A context may also be a business process, a configuration, or a network of relationships as they relate to objects as well as people. Examples might be participants in a transaction or type of transaction, an airplane, or a communications network.

Perspectives

There are numerous perspectives from which to view identity and its management. For this Guide, the Business perspective is the most important, but the following other perspectives are relevant and are addressed also: Individual, Social, Governmental, and Economic.

Individual Perspective

There are situations where it is advantageous to an individual and the organization for the organization to maintain a profile of the individual. This may, for example, be where the individual is a customer of the organization. Such a profile would be quite helpful in customizing the

products and/or services that the organization provides. This does not obviate the organization's responsibility for protecting the information provided by the individual, and the individual's perspective may be strongly influenced by the measures the organization takes to protect information about individuals and to use that information only in ways that are compatible with the individual's interests and dignity.

Social Perspective

To maximize personal freedom while maintaining the social good, there is a need to enable free flow of information without compromising personal privacy. This may seem a lofty goal, but it is an expectation of citizens and customers. Many organizations are realizing their own responsibilities in social and community citizenship. This could result in another separate profile between the individual and the organization. An example of this perspective might be a Golf Club where member information is kept. This may vary considerably from basic contact information, to keeping handicaps, to even tracking outside interests that the members might wish to share.

Governmental Perspective

To protect the rights of the individual and the integrity of the business environment, governments feel obligated to ensure that fraud and abuse or misrepresentation of information is minimized. This has resulted in significant legislation – such as the US Sarbanes-Oxley Act – which requires accountability for business transactions. That need for accountability demands that those involved in any transaction or recording thereof must have their identity verified. The US financial Know Your Customer (KYC) regulation is the due diligence and bank regulation that financial institutions and other regulated companies must perform to identify their clients and ascertain relevant information pertinent to doing financial business with them. Typically, KYC is a policy implemented to conform to a customer identification program mandated under the US Bank Secrecy Act and USA Patriot Act. KYC policies have become increasingly important globally to the prevention of identity theft fraud, money laundering, and terrorist financing. In a simple form, these rules may equate to answering twelve questions, but this is the tip of the iceberg and regulators now expect much more. KYC

should not be thought of as a format to be completed – it is a process to be undergone from the start of a customer relationship to the end.

Economic Perspective

The evolution toward a true global economy mandates an infrastructure that can determine who is responsible for payments and delivery of goods and services for the purposes of conducting transactions. The scope of such an infrastructure ultimately must encompass all consumers and businesses on the planet. This requirement is not immediate, but it is inevitable.

Business Perspective

Companies are increasingly becoming participants in economic ecosystems. The factors contributing to this are strategic partnerships, business process transformation, mergers and acquisitions, and outsourcing. The goal is to achieve real-time process visibility across enterprise boundaries.

By participating in a dynamic economic ecosystem, companies are more able to respond to market demand, eliminate process misalignments, and manage change. They also find it easier to customize their offerings to the needs of their customers. Companies that have developed or participate in economic ecosystems are realizing that the real-time visibility gained can be a strategic asset.

However, becoming an economic ecosystem and achieving process visibility is no small feat. In an enterprise, a community of identities to be managed extends over the whole of the enterprise and the organizations that it deals with, both customers and suppliers. People are no longer confined within rigid internal boundaries. The external boundary between the enterprise and other organizations is breaking down. Inter-company security requires that all such access be trustworthy and implicitly mandates the need to assure identity of the people and systems doing the accessing, especially across external boundaries.

Organizational governance and policy

Inherent in the enterprise are the contracts, covenants, and regulations by which it is bound, in order to fulfill its mission. Each spells out the accountabilities and the offices in which they reside, and ultimately form the basis for its governance. From governance flows the authority to act, embodied in the policies that support the enterprise mission.

Governance, and the policy by which it communicates its decisions to the enterprise, reside within the framework of authority and imply assignment or delegation to the agents of the enterprise in order to fulfill its objectives. For example, the Chief Operations Office (COO) delegates its authority to hire and fire staff to line managers responsible for the day-to-day details of operations management. While the authority resides in the COO, it is a matter of practical necessity to delegate their authority and thereby multiply their capabilities.

Identity management is the infrastructure that enables the mechanization of the process of authority delegation; the assignment of privilege within the framework of enterprise policy. Identity management may be thought of as the bridge between enterprise governance, its officers and agents, and the assignment of system or application privilege supporting the business processes that are the enterprise mission.

It is therefore necessary to align the capabilities of the identity management architecture with the business processes that represent enterprise governance in action. The key is to identify existing agents of enterprise policy, and to duplicate and/or optimize the business processes they represent. For example, the human resource department is often the authoritative source for worker status and information, responsible for maintaining accurate records with respect to any given worker's job role, location, department, etc., and adequate personal information to be able to support the administration of worker benefits. Its authority flows from the Chief Executive Officer (CEO), responsible for the conduct of the enterprise at large, through executive personnel directors and their staff. Its information is regulated by industry best practice, and business and regulatory requirements. In many enterprises it may provide a source of authoritative information that may be used to establish system identifiers used within the applications by which business is conducted.

An enterprise has existing mechanisms by which user accounts are established, modified, suspended, terminated, archived, and deleted. These presumably fulfill enterprise policy and represent an appropriate flow of authority resulting in the granting of authority to perform enterprise business. They may be called *security administration* or the *help desk*. The lines of authority are neither rigid nor identical from enterprise to enterprise, but they must in all cases exist and be identifiable if assurance – the ability to audit enterprise activity – is to succeed.

Assuming governance and policy have been identified and their requirements gathered, entity identifiers are the system representation of the actors authorized to use or participate in a given system or application. Like the human actors they represent, their authority, status, role, function, and responsibilities govern their relationship to the enterprise.

There are also system actors used to support processes not easily associated with an individual. These "system" or "virtual" identities require, as do their human counterparts, lifecycle management and the ability to trace accountability within the context of policy that governs their actions and management.

How to use this Guide

This Guide is for the enterprise architect undertaking the design of an information infrastructure to support internal and external user-based collaboration and commerce. This infrastructure may be comprised of Commercial Off-the-Shelf (COTS) products and existing IT assets.

This Guide is not intended to be a new approach to enterprise architecture nor a comprehensive methodology or exhaustive set of tasks to be performed. Our hope is that it will seem to be familiar, represent common practice, and focus on the essential activities of architecture design and deployment as applied to this cornerstone infrastructure. It is intended as a tool by which the architecture practitioner can steer an effective course to the delivery of demonstrable and measurable business benefit.

While this Guide does not describe a methodology, methodology is crucial to delivering measurable value. The use of a consistent methodology is more important than the specific methodology employed. The key is a traceable and repeatable process by which the goals, objectives, and progress to their attainment can be tracked and communicated. Architecture is the primary communication device by which a business process can be framed in a technical solution. There are a number of IT architecture methodologies and frameworks.

The Open Group Architecture Framework (TOGAF) is one of the most functional and comprehensive of these. For a full description of TOGAF, see the current TOGAF documentation, which is freely available from www.opengroup.org/bookstore/catalog/t_ar.htm.

This Guide:

- Describes and categorizes common requirements for identity management systems
- Describes fundamental business patterns for identity management
- Discusses identity information architecture
- Describes an identity management Technical Reference Model (TRM) that lists a set of identity management architectural building blocks and explains how they are inter-related
- Discusses the design of identity management within a Service Oriented Architecture (SOA)

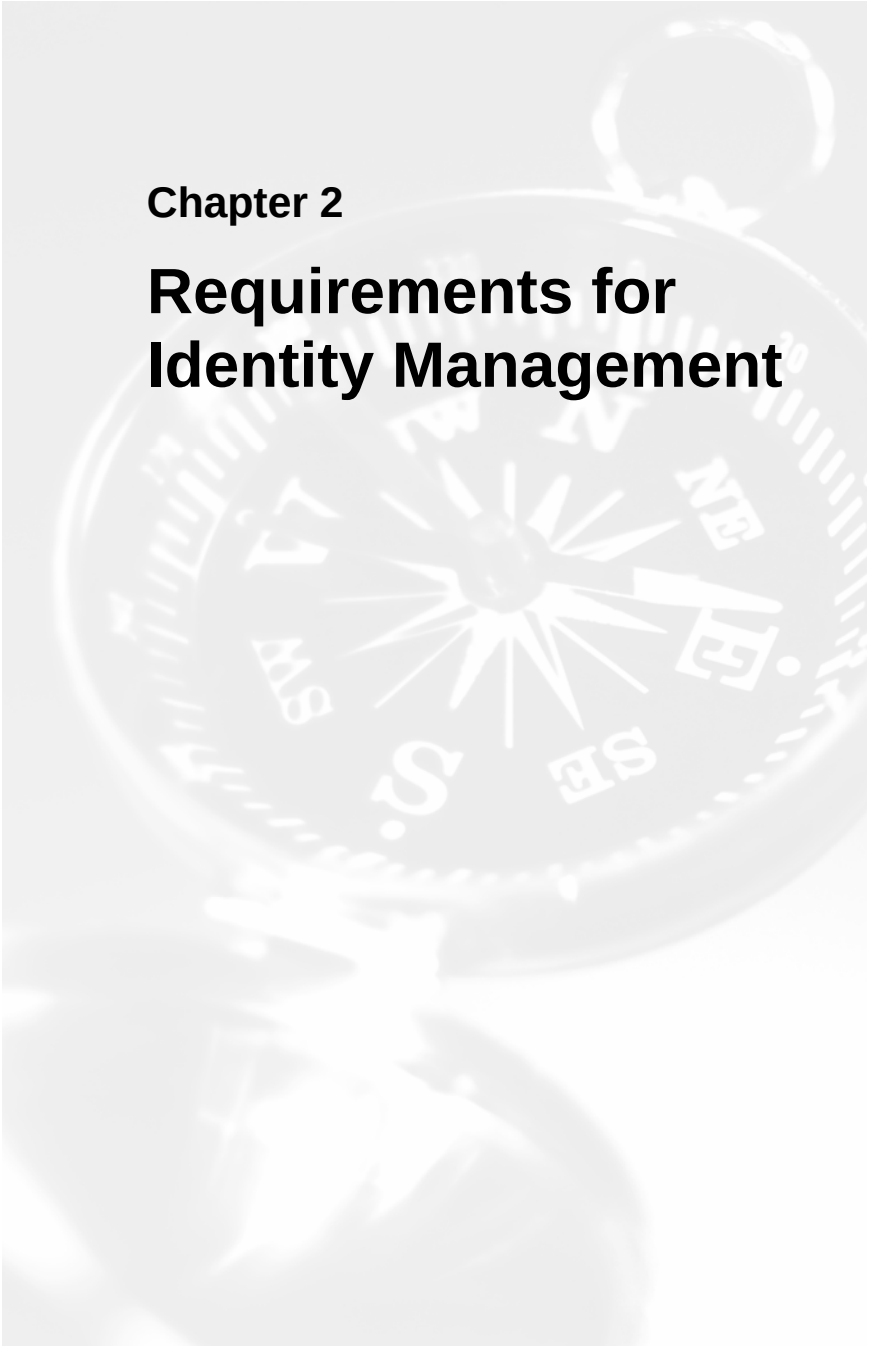
Using the information in this Guide with the TOGAF Architecture Development Method (ADM) or with another standard methodology will help the architect to develop a target identity management architecture and describe it in terms that are consistent with those in common use elsewhere.

To get started, you will need your requirements in hand and the organizational objectives firmly in mind. The goal of an architecture deployment is fast delivery of essential capability. That does not mean implementing an entire architecture in one go; but rather prioritizing the deliverable components in a way that reflects minimal interdependency between desired capabilities.

A full-featured enterprise and identity management infrastructure is not required in order to deliver every set of business capabilities. The purpose of the architecture is to provide a framework in which the rational planning and execution of new capability can be realized, and within the context of business objectives and fiscal constraint. It will enable an organization to identify the essential capabilities required and component interdependencies necessary to optimize delivery, reduce project risk, increase data quality, and increase overall success factors.

Chapter 2

Requirements for Identity Management



This Guide does not deal with the definition of the requirements management process. Rather, it provides help with the requirements gathering activity itself, by describing frequently encountered requirements for identity management. It provides a set of reference source material that the architect can use.

Comparing the requirements of the target enterprise with those of other enterprises helps the architect to organize and describe the requirements of the target enterprise in a logical and structured way, similar to that used by other architects. This makes it easier to communicate those requirements to developers and product suppliers. Also, it may enable the architect to identify requirements that have not been expressed but are nevertheless important.

The requirements described in this Guide do not all apply to all enterprises, and they do not include all possible requirements that may apply. They cannot simply be copied into the enterprise requirements database as "the requirements for identity management". Judgment is needed to select and interpret the material that is applicable. Some requirements may be specific to the industry of the enterprise. Additionally, the enterprise is likely to have other requirements that are not described, and these must also be investigated.

How requirements arise

Identity management lends itself to architectural consideration because it serves as a common or unifying role amongst a collection of functions. As a result, it is critical to capture all uses of identity and to ensure that all representations are understood. The challenge for the architect is to establish a means to serve all the varied requirements for identity.

Nevertheless, requirements tend to arise not from someone in an organization saying: "I think we need an identity management system!" but rather from an executive who says: "we need a cost appropriate means to collaborate electronically with our suppliers" or: "we're spending too much for user administration".

Other examples may look like this:

- A compliance committee's review of internal audit findings may

determine that its organization has deficient means to audit the administration of its information system users.

- A retail office supply outlet anticipates a significant reduction of transaction and administrative costs for itself and those of its business clients that adopt federated sign-on.
- The three or four new banks acquired per year by one global conglomerate now exceed the conglomerate's ability to consolidate its existing and newly acquired user populations. This is driving up administrative cost and raising internal compliance issues.
- Budget cuts at a university have eliminated funding of administrative staff required to manage partner universities' students who use its online course system. It needs to find a way to continue providing service without having to administer accounts for each of the other university's students.

Requirements gathering and analysis

When analyzing identity management requirements for an enterprise, it is often desirable to start by describing the business drivers that make that enterprise want identity management.

The next step is to work down from the business drivers to identify strategic goals and objectives within those goals. Figure 1 illustrates four common goals, and example objectives within each of them.

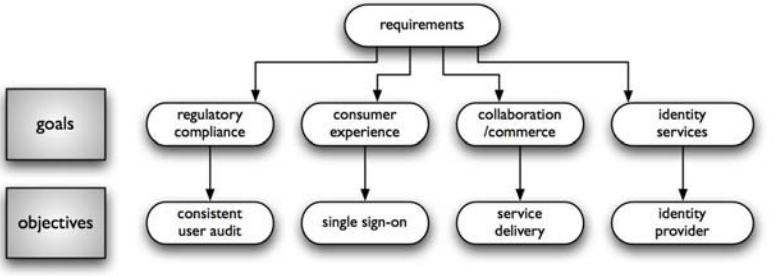


Figure 1: Identity Management Goals and Objectives

The final stage is to translate the objectives into measurable properties of the resulting infrastructure. These would include things like the ability to automate user creation and privilege assignment, or the production of accountability reports that fulfills specific regulatory or audit capability.

Common goals and objectives for enterprise identity management include regulatory compliance and accountability reporting. Measurable properties of an infrastructure supporting this requirement might be the ability to deliver timely reports detailing the systems to which particular employees have access, or a report that provides a detailed history of administrative changes to an executive's system accounts.

Another example of a measurable deliverable might be the so-called "day-zero" processing, whereby a new or transferred employee is provisioned to a standard set of resources like LAN servers, email, employee portal, etc. Demonstrated reduction in both the zero-day provisioning, the amount of time before the employee has full productivity with base systems, and reduction in the level of effort (how many people involved in provisioning and approval and how long each spends) represent achievable and measurable goals for an enterprise identity management system.

The point is that while an architecture *design* should be as comprehensive as possible, its *deployment* as an *operational IT infrastructure* should be focused on the delivery of measurable business benefit, and the components required to achieve it.

The remaining subsections of this section describe common business drivers and common categories of requirement, each of which includes goals, objectives, and measurable deliverables.

Business drivers

Why do enterprises use identity management? There are several important reasons:

- Compliance with regulations and laws (e.g., Sarbanes-Oxley, HIPAA)

- Improved operational efficiency through greater ability to communicate within the enterprise and its business partners, lower administrative overhead for identity information, and user-profiled applications
- Improved security through better control of access to information and services, and reduction of security problems that arise due to poor identity and administration
- Improved risk management due to better knowledge of system users
- Improved customer experience due to profiled services
- Reduced fraud due to better knowledge of system users
- Lower IT costs through simplified development and more efficient administration

Identities to be managed

The processes adopted and capabilities deployed to support identity management need to support identities of people and things within the same context of governance.

The people whose identities are managed often include system users, whose identities are used for authorization and access control. But the identities of other people may be managed also; for example, Customer Relationship Management (CRM) systems manage identity information about people who are customers but not system users.

Things can be security principals, just as people can, and their identities may be managed for the purposes of authorization and access control. The identities of other things beyond the scope of the enterprise may be managed also.

The enterprise at a minimum must effectively manage the information concerning its members (employees, typically, in the case of a business enterprise). But services provided by the IT infrastructure are expected to become increasingly personalized as individuals take on functions across organizational boundaries and, as boundaries across organizations become increasingly open, identity services will be extended outside the

enterprise to partners, suppliers, and customers, making the identity management task more complex.

Competition for customers is becoming ever more intense as we further engage in a global economy. When Henry Ford began manufacturing cars, he could succeed in selling cars whose only available color was black. Today, the manufacturing processes are efficient and flexible and product designs so sophisticated as to provide for many interchangeable options. This enables an organization to easily personalize products and services to the preferences of the customer.

Since customer profiles are a key integrating component for CRM systems and applications, they require the same high levels of data integrity for identity. While the preferences of each customer may reside in the CRM, there is still the need to link that to other systems such as accounts receivable, shipping, websites, etc. Thus, any identity management architecture must, at minimum, also support the context requirements for CRM.

Basic identity management functionality

The focus of identity management is the administration of attributes associated either directly or indirectly with people and things relevant to the enterprise. The values of these attributes are usually set by administrators, automated processes, and the people themselves.

The core identity management functions are creation, modification, archive, and deletion of attribute values associated with specific entities, and making those values available to users, applications, equipment (including components), and access decision points.

By establishing a single reliable source for identity information, identity management enables improved data quality, increased transactional integrity, increased efficiency, greater business integration opportunities, and new opportunities for identity-dependent services.

Additional identity management capabilities

In addition to basic identity management functionality, an enterprise will often require particular capabilities in one or more of the following areas:

- Compliance with particular legislation
- Support for legacy system integration
- User self-service
- Support for policy definition and enforcement
- Support for management by individuals of identity information for themselves and their circles of contacts

Implementation parameters

As with any kind of IT system, there are a number of parameters or qualities for which an enterprise may require particular levels to be met by the implementation. They include the following:

- Information quality
- Performance
- Availability
- Confidentiality
- Security
- Ease-of-management
- Ease-of-use

Federation

Business capabilities, such as partner collaboration and commerce, often require loose association bound by a level of trust and integrity. This is often referred to as *federated identity management*. This allows the users of one enterprise environment the ability to access the services of

another, without registering each within the user registries of the other. The measurable property is the demonstration of users obtaining (or being provided) service, with full audit and accountability, without individual registration in the service provider.

Lifecycle requirements

Enterprises manage different types of identity – such as identities of people and identities of things – with different associated lifecycles for each instance of them. The policies that govern each type and their sources of authority will likely differ. For example, an employee record may not end at termination when benefits are involved.

Regardless of lifecycle type, their characteristics at the time of an action and the management requirements for authorization need to be specified in order to provide a complete architectural solution. Increasingly, regulatory and business requirements demand an equal attention to an identifier's suspension and deletion as with its creation and modification. End-to-end lifecycle management of entity identifiers is therefore not an option, but a fundamental business requirement.

Governance and policy are applied within the context of the management of the identifier. They are enabled via approval processes – automated or (human) process-based – that capture the appropriate authorization at the appropriate points in the lifecycle, and simultaneously creating audit trails by which their activity may be scrutinized.

Lifecycle management of the entity identifier is therefore a specification, based on policy requirements, of the mechanisms, human and automated, by which entity identifiers are created, modified, terminated, and archived.

Identifier lifecycle management is comprised of the activities by which entities are identified, modified, used, discarded, or set aside. It is based on the notion that within enterprise systems, entity identifiers exist in one of a series of states, beginning with creation and ending with deletion.

An entity identifier is created when needed to audit accountability. It is no longer needed when its accountability ceases to be an issue. An employee's identifier may be managed from hire to end of life, a machine's from deployment to decommissioning.

In general, lifecycle management pertains to the assignment and removal of permissions during an entity's existence. This includes the removal of old and granting of new privilege as an entity's role changes within an organization, up to and including the defined removal of all permissions and access upon eventual retirement.

The capabilities of lifecycle management are those that help to define entity lifecycles in terms of a set of capabilities granted, and the policies that govern their granting or removal. These sets of capabilities may be associated with aggregate identifiers (groups, roles, profiles) whose names provide an abbreviated way of defining lifecycles for different entities.

Rudimentary identity management architecture will provide for creation, modification, and elimination of identifiers according to defined and specified policy guidelines. More sophisticated lifecycle management will provide for the comprehensive definition of capability sets and the ability to organize them into appropriate series or "lifecycles" for automated processing.

Administrative Workflow and Approval Processing

In a complex enterprise, administrative workflow and approval processing recognizes that within the various divisions and departments of a given enterprise, different authorities interact, each having a different scope of control. The "user administrator" may have control over the attributes of a particular identity that in turn may be responsible for granting access to particular resources. Yet, the user administrator *may not* have authority over the resources themselves. Workflow and approval processing provides the mechanism by which these two spheres may interact, under the control of enterprise policy.

For example, a user administrator may be authorized to assign a role of "development manager" to a particular entity. The assignment of such a role may grant access to a particular system; e.g., a Code Revision and

Control System (CRCS). However, a resource administrator – say the release manager of the code managed in the CRCS – alone has authority for granting access to the system to particular users. Workflow processing and approval provides a mechanism by which the resource administrator can “approve” or “deny” the request for access made by the user administrator.

Workflow and approval processing then provides a policy-based mechanism by which the actions of two separate but interacting authorities – the user administrator and resource administrator – are rationalized. It recognizes the complexity of these interactions and provides an automated mechanism to assure accountability of both management (administrative) and user action.

Accountability and audit

The ability to capture administrative activity in support of audit is fundamental to identity management’s requirements. Given the premise that all activity must be specifically authorized, all activities, functions, and processes of the identity management system should have the capability of generating machine-readable logs in order to support audit requirements. Specifically, the system should permit the capture of any identity management event that results in the generation, change, or deletion of identifier records. The system should also capture all system starts, stops, errors, and configuration changes. Event records should contain adequate information to identify the event initiator, time, object, and any other contextual information needed to support audit requirements.

Chapter 3

Business Considerations for Identity Federation

Most organizations, at least to start with, manage identities on a stand-alone basis. That is to say, they maintain their own identity information, which may include identities of people or things external to the organization (customers, suppliers, partners, etc.) as well as within it, and they do not attempt to relate their identity information to identity information held by other organizations. This may apply to divisions, even departments, within a single enterprise. However, there is a growing desire for organizations to relate each others' identity information through identity federation, and there is a growing body of standards and technology that support federation. This section analyzes the common business patterns that lead organizations to implement federated identity management.

The business drivers for federated identifier management divide across two primary use-cases. The first is reflected by consumer-to-business models (C2B) that are primarily user-centric. The second set of drivers is based on business-to-business (B2B) models, wherein two or more businesses want to share either users or services (or both) with another.

Consumer-centric federation

Consumer-centric models are characterized by mechanisms designed to enhance privacy through consumer control of attributes that identify and describe them to services with which they interact. They enhance user experience through reduced sign-on activity and enable the consumer to link various services or accounts to a primary identity provider.

The term "federation" in consumer-centric models refers to capabilities provided to a user that enable him or her to "link" various accounts together. The federation in this case is an *ad hoc* collection of service providers who are joined together, by the consumer, and who as a consequence of consumer control, share a single identity provider. These models, of course, assume the ability of the linked services to consume the credentials (assertions, tokens, etc.) provided by the consumer's preferred identity provider.

The key feature of consumer-centric models is that the federation exists for the convenience of the consumer. They imply a relationship between the consumer's identity provider and the service or account to which it is

linked. In this model the consumer's policy governs the federation. While a service provider's policy governs how and when to accept credentials from a consumer's particular identity provider, the federation in this case exists subject completely to the consumer's policy or desire.

Business-centric federation

Whereas consumer-centric federations exist completely for the convenience of the consumer, business-centric federation reflects a model whereby two or more businesses join and share services in support of their joint business aims.

The federation is therefore based upon relationships formed between business partners via contracts and covenants, and enabled by the underlying federation technology. The contracts and covenants express the terms and conditions by which business is conducted. The underlying technology simply (or perhaps not) expresses the technical realization of the relationships defined.

Two primary drivers for businesses entering into a federation are:

- Elimination or significant reduction of administration effort and cost
- Alignment of liability and business process

Two typical federation models (uni-directional and bi-directional) reflect these business drivers.

Uni-Directional Federation

This uni-directional business-to-business (B2B) case is based upon distinct business entities acting as either an identity provider or service provider. The purpose is to leverage the existing account or user administration of one business – the identity provider – by another – the service provider.

An example of this model is a corporation that contracts with a second to provide healthcare benefits administration to its workers. The healthcare benefits corporation is the service provider; the contracting

corporation is the identity provider. After the deployment of the supporting technology, users access their benefits service by authenticating first to their own or "home" organization, which in turn provides credentials to the service organization.

Administration of users and their attributes is performed at the home organization; the service organization as relying party accepts the credentials based on the agreements by which the federation was formed. The underlying technology enables secure transport of identifying information forming the technical infrastructure of the federation.

The value of the federation in this case is realized by eliminating or markedly reducing user administration across the federation. User administration occurs only at the identity provider, with little or markedly reduced administration of user attributes at the service provider.

The second mechanism to reduce cost might be thought of as "just-in-time" provisioning. In this use-case, federation protocols are used to establish service provider accounts when a user makes its first contact with the service provider. Upon receiving authenticated credentials from an identity provider for the first time, exchange of policy between service and identity providers may then result in exchange of additional user information sufficient to establish an account at the service provider. This "just-in-time" capability eliminates the so-called "user forklift" problem, whereby an identity provider, the contracting corporation in this case, is required to extract *en masse* from its user registries sufficient information to enable the contracted service provider to establish initial account information.

Bi-Directional Federation

In the bi-directional federation model, organizations act as both identity and service providers. Two use-cases exhibit the benefits of this model.

The first represents an organization or company who through acquisition or merger expands both its user and service base in a non-linear fashion.

The second case reflects the more general business case where

businesses wish to provide services to their business partners (and their business partners' employees, users, etc.).

In the merger and acquisition (M&A) use-case, the volume and scale of the event precludes user-by-user, service-by-service introduction of the new "parts" of the organization. Forklift of users from one organization to the next presents a non-scaleable problem in that the administrative effort to join organizations is either too costly or too slow. Additional factors such as jurisdictional differences (inter-country M&A) may introduce additional risk that needs to be addressed on an organization-by-organization basis.

Establishing an intra-company organization in this case eliminates internal user forklift, and enables the uniform application of policy subject to the different jurisdictions that may be represented in a merger.

Chapter 4

Identity Management Information Architecture

This chapter addresses the architecture of the information with which identity management systems are concerned: its major types and sources.[1]

It is important to note that this effort is *not* concerned with database design. The goal is to define the data entities relevant to the enterprise; not to design logical or physical storage systems.

Identity management information

Identity management systems are concerned with two kinds of information:

- *Identity information*, comprising the attributes of the people and things whose identities are managed
- *Identity relationship information*, connecting different sets of identity information that relate to the same identity – particularly in federated systems

Enterprise identity management is focused on managing the attributes of entity identifiers within the context of corporate (organizational) governance and policy. Federated identity management is focused on managing the attributes of the relationships that form a "federation". In the first case, entity attributes are managed directly. In the second, entities are managed – recognized, granted, or denied access – in the context of relationships (federation) between organizations that have agreed to share either identities or services.

Identity Information

The semantics of identifiers and related attributes is a difficult area, but one in which there has been substantial standards activity. It is a frequent practice to reference the X.500 standard as a primary source of identifier and attribute semantics.

[1] In TOGAF Version 8, this corresponds to the Data Systems part of the Information Systems Architecture phase.

X.500 refers to a series of recommendations of the International Telecommunications Union (ITU), originally developed to define standard directory services for electronic mail, but appropriate to storing identity information of all kinds. They cover information models, access protocols, and inter-directory communications protocols.

The Internet Engineering Task Force (IETF) work on directories has resulted in a slightly different access protocol optimized for use over the Internet – the Lightweight Directory Access Protocol (LDAP) – but assumes the same model for identity information as X.500. The X.500 information model is based on object classes of entities whose identities are managed, and attributes associated with those object classes. For further information on this, refer to IETF RFC 2798, which includes example object classes and attributes, including *inetOrgPerson*.

X.500 assumes a hierarchical directory structure. Originally, common practice was to put country at the first level of the hierarchy, organization at the next level, organizational unit at the next levels (there could be multiple levels of organizational unit in the hierarchy), with the directory entries at the lowest level, the leaf nodes of the hierarchy. So, for example, for a person John Smith in the finance department of the San Francisco branch of The Open Group, which is headquartered in the USA, the successive nodes of the hierarchy would be:

country=*us*
organization=*opengroup*
organizational unit=*sanfrancisco*
organizational unit=*finance*
name=*John Smith*

A variation introduced by the IETF organizes the hierarchy by Internet domain name, with top-level domain (“com”, “org”, “edu”, etc.) at the first level, registered domains at the next level, subdomains of those domains at the next levels, and directory entries at the lowest level (so, for example, for user jsmith@sf.opengroup.org, the hierarchy nodes at successive levels are “*org*”, “*opengroup*”, “*sf*”, and “*jsmith*”).

In practice, neither of these hierarchies is stable under organizational change. Many organizations have found it preferable to adopt a flat

hierarchy with all entries at the top level. This, however, implies that the entries are given identifiers that are unique within the organization, which can also pose problems.

Defining the appropriate form of identifier for the entities whose identity information is managed is a crucial aspect of the identity information architecture. These identifiers constitute the keys by which identity information is accessed. X.500 defines a particular form of identifier for each directory entry – its *distinguished name* – but a given entity can in practice often have multiple identifiers. An example would be a person who is both an employee and a customer of a given enterprise.

Where accountability is a fundamental requirement of identity management, the architecture must support the ability to associate any given system's entity identifiers with an authorized human user, be they administrators or ordinary users. One mechanism is to establish a single *enterprise identity*, associated with an authorized and accountable *human* user, to which all other system and application entity identifiers are associated. In this way the architecture provides support for traceability and accountability.

This suggests that the architecture supports a notion of a primary enterprise identity, bound to an identified registry, to which all other system identifiers are associated. In an individual enterprise, the capability to associate this core identifier with those of various heterogeneous systems enables an enterprise to manage heterogeneous identifiers in a homogeneous way.

The principle then is to associate a single enterprise individual with multiple system identifiers such that accountability is assured.[2]

Identity Relationship Information

The X.500 information model includes an *alias* construct that can be used to define equivalences between directory entries. However, when relating identifiers held by different organizations, and even by different

[2] For several reasons, including privacy and stability, personal names should not be used as principle identifiers.

domains within the same organization, it is now common practice to use some form of identity federation. Federation – from the Latin *fœdus*, covenant[3] – implies an agreement of two or more parties. Federation for the purpose of cross-domain identity sharing implies covenant or contract between the federation partners.

Current standards and models reflect federation more as an *ad hoc* association of partners interested in shared objectives, than official organizations of business affiliates. This is chiefly due to the fact that the federation exists only in the shared keys and protocols that each partner individually defines and maintains – rather than in an *objective* definition of federation that its members then share.

While some protocols support the import and export of a list of partners and their public keys, a federation is created from the perspective of its members and users, not as a definition of an entity itself. The practical consequence is that each partner instantiates its own mechanisms, installing its own keys, the public keys of its partners, and any policy information that the partners may share.

Each of the emerging standards reflects this partner-centric view of a federation, and current implementations echo it. While from each of the Liberty, SAML, and WS-* protocols one can infer a central entity comprised of members, current implementation and interoperability requires the *local* instantiation of a "virtual" federation.

Sources of identity information

In order to manage identities efficiently, it is critical to establish a *master* source for each item of identity information. This source is the true authoritative source for the context. All other sources are replicates or derivatives from that. Thus, updating at the master minimizes the need to maintain multiple copies or to determine which has the true current value for an attribute.

[3] Wikipedia, en.wikipedia.org/wiki/Federation.

In some cases, the master source is a user of the identity management system, or an administrator. In others, it may be outside. For example, payroll systems are often used as the authoritative source for who is an employee. In these cases, users or administrators may input the information to the identity management system. However, it is preferable to input it automatically through some kind of automation or workflow mechanism.

This leads to three administrative models for identity information, as shown in Figure 2.

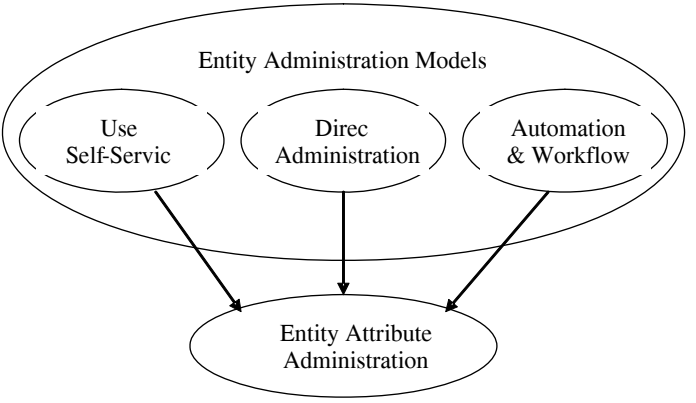


Figure 2: Enterprise Identity Management

User Self-Service

User self-service denotes capabilities by which users can modify specific attributes of identity records as permitted by enterprise policy.

An enterprise may recognize the user as the authoritative source of particular data. It can therefore economize on its administrative operations by transferring responsibility for these attributes to the user.

The essential characteristics for user self-service of identity attributes are as follows:

- They are recognized by policy as being within the purview of the user.
- Access to modify these attributes is constrained to the user or authorized delegates; i.e., administrators or user-identified delegates.
- Any consequence for improperly or incorrectly entered or modified attributes is borne by the user alone.

The benefit of user self-service must be weighed against the requirements of the enterprise. While the user may properly be considered the source of a mailing address change, an incorrectly entered address may result in liability to the enterprise. If, for example, the address is used for mailing payroll checks, an incorrectly entered address may result in significant administrative overhead and cost to rectify.

In the notion of federated identity management, there is the concept of user-initiated account linking, by which an individual with authority over various accounts associates each with the others. The fundamental purpose from this perspective is to enable single sign-on.

However, in the enterprise, the principle is that user accounts are governed by enterprise policy, and authority to associate identifiers rests with the enterprise, not the user to whom the accounts are associated. Therefore, while a useful concept for a user having authority over disassociated accounts, the concept of user-initiated account linking has less appeal within the enterprise.

Note: This is one model for the enterprise use-case; specifically where enterprise governance across associated accounts is mandatory. This use-case can be extended for the case of associated or federated enterprises where the lines of authority in the federation are consequent to the enterprise relationships (contracts, covenants, regulations) between each, *not* the relationship between the individual user and any given enterprise. This perspective is not to discount the user-initiated account association (linking) use-case, but rather to distinguish it under the guise of enterprise governance and accountability.

Direct Administration

Direct administration is modification of identity information by a duly authorized delegate of the enterprise. Typically, these are called *user administrators*, *security administrators*, or the *help desk*.

Ideally, identity management architecture should be able to support the assignment of administrative capability to specified delegates of the enterprise, and to select identities within the enterprise's scope of control. The first is simply the establishment of administratively capable user groups; the second is often termed *delegate user administration*, and refers specifically to the assignment of administrative duties to encompass a defined subset of an enterprise's total identity population.

Highly secure identity management architecture will enable, but not require, separation of duties among administrators. Delegate user administration is one example, where "Chinese-wall" type access policies may proscribe administrators from different departments – i.e., analysis and brokerage – from administering each other's department identities. Secure deployments may also require separation of duties between those that administer users, and those that manage the records or logs used to audit identity management activity.

Automation and Workflow

Many identity management architectures provide for automated distribution of identity information through the system, using mechanisms such as directory replication, meta-directory, virtual directory, synchronization, and provisioning.

In architectures supporting automated administration, means should be provided to support integration and/or transmission of identity information from authoritative sources to the identity management system.

There are two fundamental approaches. The first is real-time integration, where authoritative data is accessed as needed by the identity management system, in the normal course of its operations. The second is where authoritative data is transmitted from the authoritative source to

the identity management system. In this case, the data forms the basis for a comprehensive identity record that is managed solely within the identity management system.

While the first approach offers the advantage of single run-time source and storage of authoritative identity data, many (if not most) “systems of record” are constrained through either policy or technology from acting in a real-time technical role to ancillary systems. Inadequate technical interface, operational constraint, or security of authoritative data are the most common reasons for the lack of “real-time” interface to authoritative identity data.

The second approach, identity data transmission from authoritative source to identity management system, reduces inter-system dependency and potential corruption of authoritative data from a non-authoritative source.[4]

[4] Architectural principle: unidirectional flow of authoritative data; i.e., circular modification not permitted.

Chapter 5

Identity Management Technical Architecture

The objective of this chapter is to provide a technical reference architecture that can form the basis for the ensuing implementation work. As with the earlier architectures discussed, identity management serves as a key enabler for reliable interfaces, both for content and for context.

A Technical Reference Model (TRM) has two main components:

- A *taxonomy*, which defines terminology and provides a coherent description of the components and conceptual structure of an information system
- An associated *TRM graphic*, which provides a visual representation of the taxonomy, as an aid to understanding

The identity management TRM graphic presented in Figure 3 provides a widely-accepted core taxonomy for identity management, and an appropriate visual representation of that taxonomy.

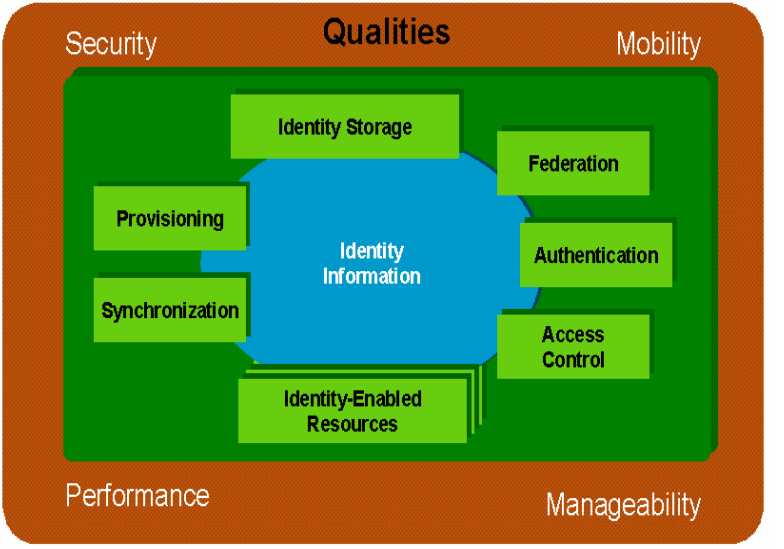


Figure 3: An Identity Management Technical Reference Model

Note that this is not an implementation block diagram. It does not say that an enterprise's identity management subsystem must consist of an

identity store, a federation module, an authentication module, and so on. Typically, an enterprise's identity management components will include identity stores, federation modules, etc., possibly of several different kinds. It is part of the architect's job to specify the particular components used in the enterprise, and to specify how they inter-relate. The reference model provides a starting point for this activity.

In defining an enterprise identity management implementation architecture, an architect may develop implementation block diagrams, and should develop component specifications that include the following three types of specification:

- *Functional specifications* that describe aspects of functional behavior
- *Interface specifications* that describe component interfaces
- *Qualities specifications* – these define quality aspects such as security, manageability, mobility, and performance, as applicable

These specifications – particularly the interface specifications – should where possible be based on international, consortium, or industry standards. A list of standards that are appropriate for identity management component interfaces can be found in The Open Group Standards Information Base (SIB).[5]

The following sections discuss the elements of the model that are illustrated in the graphic.

Identity information

Identity information comprises items of information, each of which is associated with an identity. Such an item can be an identifier, a credential, a permission or role, or an item related to a specific identity-enabled resource. Note that an item can fall into more than one of these classes; *email address* is a prime example, as it can be used as an identifier as well as being related to the email service.

[5] The Open Group Standards Information (SIB) is at: www.opengroup.org/sib.

Identity information is shared (under appropriate security constraints) between all of an enterprise's identity management components. The interface specifications for each identity management component should include:

- An *Identity Representation Specification* that defines how identity information is represented at the component's interfaces (for example, using the X.500 model and the *inetorgperson* schema)
- An *Identity Information Packaging and Transport Specification* that defines how identity information is packaged and transported at the component's interfaces (for example, using LDAP v3)

Identity storage

An identity store holds identity information. Identity stores include not only systems that use the ITU X.500 protocols or the IETF Lightweight Directory Access Protocol (LDAP), but also relational databases, flat files, and data stores of other kinds.

Note that a single logical store may be implemented as multiple physical stores. There are two commonly-encountered models for this: *distributed directory* and *meta/virtual directory*.

Distributed Directory

The concept of directory originated in the X.500 standards of the ITU. The ITU developed a distributed model of directory services. This is shown in Figure 4.

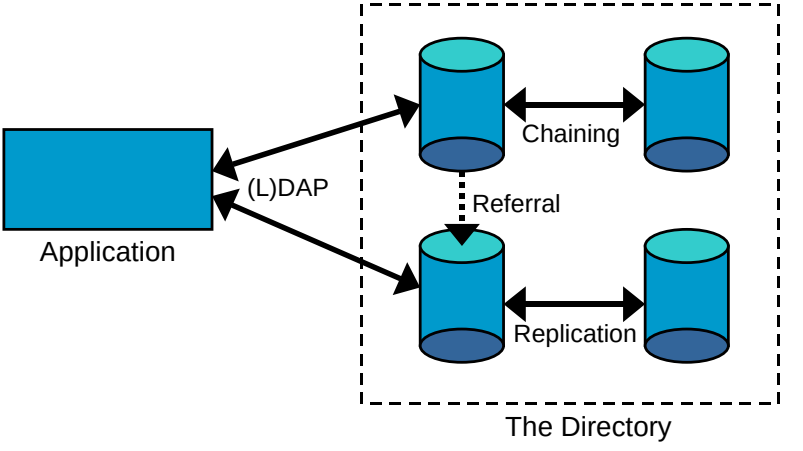


Figure 4: The X.500 Directory Model

X.500-compliant products are available from a number of suppliers. In addition, there are proprietary products from other major directory suppliers that provide similar functionality.

Meta or Virtual Directory

The concept of meta-directory was introduced because many existing repositories of enterprise information do not support a common access protocol, and different products that do support a directory access protocol (LDAP or DAP), except for the minority that support X.500, often cannot communicate with each other.

A meta-directory holds the join of the information in a number of directories and other physical information stores in an enterprise. It enables other components to access that information as though it was stored in a single directory. This is illustrated in Figure 5.

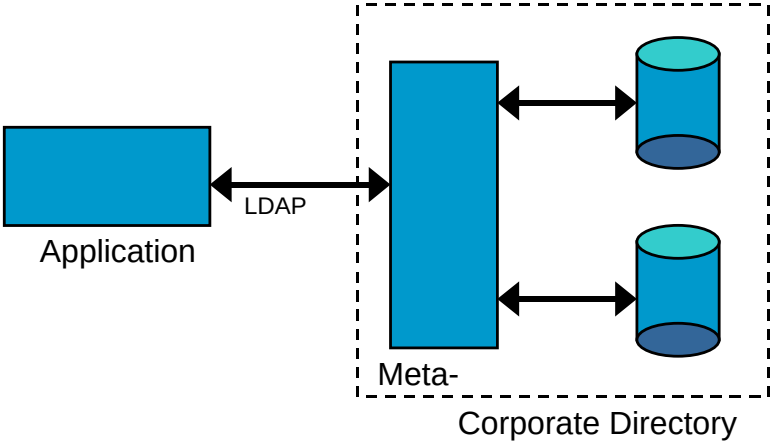


Figure 5: Meta-Directory Model

A virtual directory is similar in concept to a meta-directory, but does not incorporate its own physical information store. It handles requests to access information by forwarding them on-the-fly to the physical information stores that it co-ordinates.

Meta-directories and virtual directories present a standard interface – LDAP – to the components that use them, but their interfaces to the physical information stores that they co-ordinate are generally non-standard.

Identity-enabled resources

Identity-enabled resources are resources, services, and applications (the term “resource” has a broad meaning in this context) that rely on identity information, either for access control or to provide functionality. They include operating systems, database management systems, and enterprise applications.

Many resources rely on identity information, either for access control or to provide functionality. In some cases they hold this information internally. In other cases they obtain it from dedicated identity stores.

It is a good design principle to maximize the use of dedicated identity stores and minimize the holding of identity information internally by resources. This makes system administration easier, makes the user experience more consistent, and simplifies provisioning and synchronization.

Provisioning

Provisioning is the process of configuring accounts and identity information in resources for the purpose of controlling access to them.

There are three key aspects of provisioning:

- *Account provisioning* deals with identity-related information associated with any entity – which may be human, machine, application, building, room, in fact any object – whose identity can be authenticated. The definition "any entity whose identity can be authenticated" is the one used in ISO/IEC 10181-2. Identity-related information is attributes which support the process of authenticating that entity; e.g., for humans this will be personal attributes, affiliations, etc.
- *Resource provisioning* deals with the management of permissions associated with business assets such as computers, databases, applications, etc.
- *Account de-provisioning* deals with the termination of access rights to business assets – systems and services – and re-allocation of those systems and services.

Synchronization

Identity synchronization is the process of copying identity information (especially passwords) between identity stores and resources in order to create a consistent set of identity information across them all. It includes both replication of information between identity stores and projection of identity store information onto resources.

Note that replication is also used to copy information between different physical stores in a logical identity store. In some cases, it may be possible to describe an architecture either as including different synchronized identity stores or as including a single logical identity store with internal replication. In such cases, the choice of description is a matter of judgment for the architects concerned.

Access control

Access control refers to the control of access to resources. It includes the determination of whether an entity may use a resource (authorization) and the enforcement of the result of that determination (access permission enforcement). It may be carried out by dedicated access control components, by access control functionality within resources, or by a combination of these two methods.

Different kinds of access – for example, read access and modify access – may be subject to access control. An entity may be granted some kinds of access and denied others, to the same resource.

Different kinds of access apply to different resources. A file system might have read, write, and modify access, while an application might have user and supervisor access, for example.

Access control is closely related to authorization and permissions management. The meanings of these terms overlap, and they are sometimes used interchangeably.

Permissions management refers to the management of information about what entities should be allowed to do. Appropriate use of resources is assured through the management and enforcement of permissions associated with those resources. Permissions include access permissions and more: they include permission to read, compare, write, modify, create, destroy, execute, copy, print, forward, delegate, purchase, authorize, approve, sell, sublease, assign, transfer, hire, fire, promote, and so on.

The term “authorization” has two distinct meanings:

- Authorization as the process of determining whether an entity should be allowed to do something. In this respect, authorization to access resources is an aspect of access control.
- Authorization as the process of assigning permissions to entities.

Because of the potential for confusion arising from the term having two meanings, it is not used in this TRM. It is a good design principle to carry out access control in dedicated access control components as far as possible, and in resources as little as possible (unless there are unusual security considerations, which may require special measures for particular components). This simplifies system design and administration, and facilitates a better user experience (including single/simplified sign-on). The ability to do it is on occasions limited by the fact that many bought-in components are designed to do their own access control, and do not have interfaces that can be used by dedicated access control components.

Authentication

Authentication is the process of establishing confidence in the truth of some claim. In the context of identity management, an authentication system provides an understood level of confidence that an identifier refers to a specific entity, or that an attribute applies to a specific entity.

Authentication may be carried out by dedicated authentication components, by authentication functionality within resources, or by a combination of these two methods.

It is a good design principle to carry out authentication in dedicated authentication components as far as possible, and in resources as little as possible. This simplifies system design and administration, facilitates a better user experience, and leads to better overall system security. The ability to do it is on occasions limited by the fact that many bought-in components are designed to do their own authentication.

The interface specifications of an authentication component may include an identity information assurance specification that defines how information obtained from other building blocks is assured.

Federation

Identity federation is a standard way of allowing enterprises to provide services directly for people registered at other (partner) enterprises. It can also be used within an enterprise between departments or divisions with different identity management systems. It can apply to non-human entities, such as applications, as well as to people.

Within a federation of services, an enterprise (or department) can obtain trusted information about a user from the user's home organization (or information-providing service). The enterprise does not need to register and maintain that user's identity, and the user is spared from having to obtain and remember a new login in order to interact with the enterprise.

A federation system creates associations between sets of identity information, including information held by different organizations, to enable authentication and access control systems to support this kind of federated operation.

The interface specifications for a federation component should include an identity information assurance specification that defines how information obtained from other federation systems is assured.

Qualities

Besides the set of components making up our TRM, there is a set of attributes or qualities that are applicable across the components. The graphic in Figure 3 captures this concept by depicting the TRM components sitting on a backplane of qualities. The qualities that apply should be specified in detail during the development of a target architecture, and may be more or less important, depending on the context.

Qualities that are important in the context of identity management include Security, Mobility, Performance, and Manageability.

Chapter 6

Identity Management in a Service Oriented Architecture

This section discusses the implementation of identity management within a Service Oriented Architecture (SOA).

Service-orientation

Service-orientation is a way of thinking in terms of services and service-based development and the outcomes of services. It is a style that is increasingly being used for enterprise IT architecture.

A service:

- Is a logical representation of a repeatable business activity that has a specified outcome (e.g., check customer credit, provide weather data, consolidate drilling reports)
- Is self-contained
- May be composed of other services
- Is a “black box” to consumers of the service

The SOA architectural style has the following distinctive features:

- It is based on the design of the services – which mirror real-world business activities – comprising the enterprise (or inter-enterprise) business processes.
- Service representation utilizes business descriptions to provide context (i.e., business process, goal, rule, policy, service interface, and service component) and implements services using service orchestration.
- It places unique requirements on the infrastructure – it is recommended that implementations use open standards to realize desired goals for interoperability and location transparency.
- Implementations are environment-specific – they are constrained or enabled by context and must be described within that context.
- It requires strong governance of service representation and implementation.
- It requires a “Litmus test”, which determines what is a “good service”.

Definition for SOA

SOA has no complete definition that is commonly agreed between industry groups. While most groups share the same basic conceptual definition, they nearly all diverge in their agreement on what a “service” is. This crucial divergence adversely impacts interoperable implementations.

The Open Group definition has been developed by its SOA Working Group to suit the objectives of the Working Group; see www.opengroup.org/projects/soa.

The commonly shared conceptual definition is that SOA is a software architecture that uses loosely coupled independent software services that an application can call upon to perform a task (e.g., support the requirements of business processes and software users) without the service needing to know anything about the calling application, or the application needing to know how the service performs it task(s). So, in an SOA environment the resources in a network are available as independent services that can be accessed by applications without knowledge of their underlying platform implementation.

The definitions diverge significantly on what constitutes a “service” and a “resource”, and what the interface definition is to an SOA service. This divergence hinders interoperable implementations.

Identity providers

In web services architecture, the term “identity provider” refers to a special type of web service – one that provides authenticated entity identifiers (attributes) to another service. The service provider relies on the authenticity of the identifying attributes provided by the identity provider, according to a pre-arranged agreement or contract between them. This concept of identity provider can be applied more generally in all forms of SOA.

Implicit in the concept of identity provider is some charter that defines the authority of the provider, and the semantics of the attributes it provides to its clients. Its clients may be either other service providers,

users who in turn present the credentials representing identity attributes, or businesses who contract with the identity provider in order to provide service to their own clients or users.

In an SOA, requirements like those abbreviated above may be categorized by the *role* of the organization providing service. All federated identity providers by definition must provide basic identity management functionality. While these capabilities may not reflect the breadth of complete identifier lifecycle management, federated identity providers need at least the fundamental capabilities of registering (creating) entity identifiers, modifying their attributes, and deleting (or archiving) their accounts.

Any particular organization may perform the role of an identity provider, or a service provider, or both. The architecture for dual role (identity provider and service provider) organizations needs to reflect each as separate components of the architecture. Furthermore, an organization typically realizes that each activity is driven from distinctly different business processes. The deployment of the supporting infrastructure will, however, likely support each distinct role, so this fact should be captured as well.

Identity management services

In an SOA, services are typically defined at a finer granularity than the building blocks of our Technical Reference Model (TRM). There is no standard set of identity management services for SOA, so the following sections describe some example services that it could be appropriate to define as part of an enterprise SOA.

Services common to identity and service providers

This is comprised of the methods and protocols by which agreements between federation partners are implemented. They include meta-data exchange, key exchange, service end-point definition, and policy exchange. The architect should specify the process and methods by which this information is created and exchanged, including protocols by which it may be transmitted between yet-to-be partners.

Services implemented by identity providers

Authentication

The users managed by an identity provider as part of federated sign-on authenticate with the identity provider. The architecture will express one or more authentication service in its description that supports this capability. It is generally up to the identity provider to determine what methods and protocols it will employ.

Credential Services

Credential services are responsible for the creation of security tokens required (through prior agreement) by service providers. This includes both the *assertion* format, and the underlying credential represented as the *security token*.

Note that this service is often combined with the following federated identity protocol service. It is useful, however, to represent these services separately, because credential services may be more tightly bound to the authentication method employed, rather than to the specific assertion format and protocol used to transmit it.

Federated Identity Protocol Services

Federated identity protocol services represent the communication layer by which entity identifiers are transmitted from one party to another. They may be comprised of passive *browser protocols*, whereby an *artifact* is transmitted from the identity provider through the user's browser to a service provider, at which point the service provider validates over a back-channel (not through the user's browser) over SOAP[6] to the identity provider to receive the proper user credentials.

[6] Simple Object Access Protocol (SOAP), defined by the W3C; refer to www.w3.org/TR/soap.

Registration and User/Account Lifecycle Service

This service represents the protocols by which users are provisioned to service providers out-of-band of a user's federated sign-on session. A service provider may require this capability in cases where the service provider may need to provision in advance of clients.

Credential Validation

This service enables policy-based validation of credentials delivered to the service provider.

Federated service provider

The "pure" service provider delivers service based on partner agreements, with the aim of eliminating user administration within its domain. It relies solely on the assertions provided by an identity provider and the policies on which its federation relationship(s) is (are) based.

SSO Protocol Services

These services represent the means by which an initial service request is associated with one or more identity providers, and the protocol(s) specific to that identity provider are identified and employed to enable federated Single Sign-on (SSO). Single token architectures support a narrow set of protocols and security tokens, and do not require this discriminating capability. Where a broader set of protocols is to be supported (partners who implement dissimilar technologies), this architectural capability is critical.

Credential Acceptance

Once a service provider has validated a credential with an identity provider, it must either accept or reject the credential for the service request. This is the first level of access control performed, and the only one exposed to the federation layer.

Identifier Mapping

Identifier mapping is the representation in the service provider's domain of the federated identity asserted by the identity provider.

Glossary

ADM	Architecture Development Method (TOGAF)
COTS	Commercial Off-the-Shelf
CRCS	Code Revision and Control System
CRM	Customer Relationship Management
IETF	Internet Engineering Task Force
ITU	International Telecommunications Union
KYC	Know Your Customer
LDAP	Lightweight Directory Access Protocol
SIB	Standards Information Base
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SSO	Single Sign-on
TOGAF	The Open Group Architecture Framework
TRM	Technical Reference Model

References

The following documents are referenced in this Guide:

- Business Scenario, July 2002, Identity Management (K023), published by The Open Group (www.opengroup.org/bookstore/catalog/k023.htm).

  This Business Scenario explores the requirements for identity management, the environment within which it must exist, and the implementation architectures that have been proposed for it.

- White Paper, March 2004, Identity Management (W041), published by The Open Group (www.opengroup.org/bookstore/catalog/w041.htm).

  This document explores key concepts of identity management (trust, authentication, provisioning, authorization, and directories), places these concepts within their business, personal, and technical perspectives, and proposes a set of steps to promote the resolution of industry-wide impediments to interoperable identity management solutions.

- TOGAF, Version 8.1.1 'Enterprise Edition', September 2006 (I061), published by The Open Group (www.opengroup.org/bookstore/catalog/i061.htm).

  TOGAF 8.1 represents a consensus industry framework and method for enterprise architecture that is available for use by organizations around the world, members of The Open Group and non-members alike. TOGAF is a framework – a detailed method and a set of supporting tools – for developing IT architectures. It may be used freely by any organization wishing to develop an IT architecture for use within that organization. TOGAF, Version 8.1 uses the same underlying method for developing IT architectures that was developed and refined – with a particular focus on Technology Architecture – in the versions of TOGAF up to and including Version 7. However, it applies that method to the other domains of an overall enterprise architecture: the Business Architecture, the Data Architecture, and the Applications Architecture, as well as the Technology Architecture.

- IETF RFC 2798: Definition of the *inetOrgPerson* LDAP Object Class, M. Smith, April 2000.

- ISO/IEC 10181-2:1996: Information Technology – Open Systems Interconnection – Security Frameworks for Open Systems: Authentication Framework

About the Authors

Christopher J. Harding

Forum Director SOA & Semantic Interoperability, The Open Group

Dr. Chris Harding has been with The Open Group for ten years, and is currently responsible for managing and supporting its work on semantic interoperability and SOA.

Before joining The Open Group, he was a consultant, and a designer and development manager of communications software. With a PhD in mathematical logic, he welcomes the current upsurge of interest in semantic technology, and the opportunity to apply logical theory to practical use.

He has presented at Open Group and other conferences on a range of topics, and contributes regular articles to ebizQ. He is a certified TOGAF practitioner.

Roger K. Mizumori

President, Waterforest Consulting Services
Technical Designer, Boeing

Roger Mizumori is an innovative leader with extensive expert experience in a wide range of emerging technologies (Wireless, Knowledge Management, Directories, and Unified Messaging). He has successfully managed many large integration projects, including cross-industry efforts. Roger has consulted to Fortune 100 enterprises as well as start-up companies. He also developed/published a Messaging Services Planning approach cited by Bill Gates (COMDEX '97). He has also taught at the University of Phoenix and at Bellevue Community College. Roger recently returned to Boeing to work on the 787 Dreamliner.

Roger is active in Industry groups (co-Chair, Mobile Management Forum; Vice-Chair, Electronic Messaging Association; Executive Board, XAPIA; US Dept of State [CCITT], and The Open Group). He is an Author – The Practical Guide to X.400 Addressing (International Thompson, Boston, MA, June 1995, ISBN 1-85032-210-4) – and has spoken at many international events and conferences. Previously, he was with Compaq Computer Corporation/Digital Equipment Corporation, Enterprise Solutions Ltd., Boeing, Occidental Petroleum, Claremont Colleges, and US Civil Service.

Roger holds a Certificate in Content Management from the University of Washington, an MA in Education (Quantitative Analysis) from the Claremont Graduate School, and a BA in Mathematics from the Claremont McKenna College.

Ronald B. Williams

Product Architect, IBM Tivoli Access Manager,
IBM Corporation
Software Group, Tivoli Software

As part of the Security Group for IBM Tivoli Software, Ron is lead architect responsible for the architecture and design of the Tivoli Access Manager family of products. He designs security architecture for IBM products, corporate operations, and for IBM's customers. He is a member of The Open Group Security Forum, and of the Oasis technical committees for Web Services Security, Web Services Secure Exchange, and Extensible Access Control Markup Language (XACML).

A graduate of the University of California, Santa Barbara, Ron is a Security Architect, System Designer, and frequent speaker on Information Security. He has been principal investigator and co-author of numerous patents pertaining to secure web authentication, authorization, and session management systems. He has developed security infrastructures for healthcare and for global financial institutions.

He developed his foundation in Healthcare Security and Privacy as a security architect for Kaiser Permanente's national strategic planning staff, joined the DASCOM, *the Authorization Authority* as an Enterprise Architect in 1999, which was subsequently acquired by IBM. His professional focus is the application of standards-based security technologies to both product and client solution architectures.

Ron lives in Austin, Texas, with his wife of 22 years, two children, two dogs, and a growing collection of guitars.

index