CIM User and Security Model White Paper

Version 2.7

June 2, 2003

Abstract

The DMTF Common Information Model (CIM) is a conceptual information model for describing computing and business entities in enterprise and Internet environments. It provides a consistent definition and structure of data, using object-oriented techniques. The CIM Schema establishes a common conceptual framework that describes the managed environment.

The User and Security Model provides classes to manage and retrieve organizational data and information about "users" of services and their credentials. As part of this work, systems' accounts for users, and the key services involved in managing authentication and authorization are modeled.

This white paper contains a short description of the CIM User and Security Model and an example instantiation of the model, complete with MOF files.

Notice

DSP0139 Status: Preliminary

DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems management and interoperability. DMTF specifications and documents may be reproduced for uses consistent with this purpose by members and non-members, provided that correct attribution is given. As DMTF specifications may be revised from time to time, the particular version and release cited should always be noted.

Implementation of certain elements of this standard or proposed standard may be subject to third party patent rights, including provisional patent rights (herein "patent right"). DMTF makes no representation to users of the standard as to the existence of such right, and is not responsible to recognize, disclose, or identify any or all such third party patent rights, owners, or claimants, not for any incomplete or inaccurate identification or disclosure of such rights, owners, or claimants. DMTF shall have no liability to any party, in any manner or circumstance, under any legal theory whatsoever, for failure to recognize, disclose, or identify any such third party patent rights, or for such party's reliance on the standard or incorporation thereof in its product, protocols, or testing procedures. DMTF shall have no liability to any party implementing such standard, whether such implementation is foreseeable or not, nor to any patent owner or claimant, and shall have no liability or responsibility for costs or losses incurred if a standard is withdrawn or modified after publication, and shall be indemnified and held harmless by any party implementing the standard from any and all claims of infringement by a patent owner for such implementations.

For information about patents held by third-parties which have notified the DMTF that, in their opinion, such patents may relate to or impact implementation of DMTF standards, visit http://www.dmtf.org/about/policies/disclosures.php.

Table of Contents

1. Introduction

The CIM ***User and Security*** Working Group charter assigns the task of defining the objects and access methods required to manage *principals*, where a *principal* is a representation of a user of a managed system and its resources. These users may be human users, services (e.g., a service running on a network device), and groups thereof. The mapping of this CIM Common Model to current directory models is also specifically included in the working group charter.

User administration includes such functions as managing accounts for users or groups of users on a computer system or in an administrative domain, and establishing and applying authentication and authorization policies for controlling access to system resources. As such, the User and Security Model provides management information that identifies the users and groups thereof, their credentials, the systems' accounts for those users, the managed resources which are protected, and key services involved in managing access to those resources, as well as the relationships between all these objects.

The User and Security Model is also expected to provide a framework for further extensions to the security portion of the model. Because system security is such a large topic, the CIM Security model is certainly not complete, but it does provide commonly needed classes from which vendor products may derive their specific information models. Future CIM work is expected to continue to expand on the foundation set of classes in this CIM Schema.

One of the common data sources – and the only standards-based source – for information about users is an LDAP-accessible directory. There are a number of commonly used and standard LDAP classes defined, and the User and Security model is designed to take advantage of information stored according to these LDAP schemata. A straightforward mapping of these classes was chosen to minimize the complexity of the transforms from LDAP schema to CIM schema, and vice-versa.

The CIM User and Security Working Group charter also identifies requirements that are not covered in this version of the model. The model is not intended to explicitly cover entities such as customers, although the classes needed to model the customer-supplier relationship might be derived from the model. As previously noted, the current version does not purport to provide a complete security model.

Further, the user and security model does not attempt to address the complete needs of "white pages" or "yellow pages" applications. Nor does the model attempt to address the management of people, places, and things (e.g., facilities management), apart from the relationship between people and computer system resources.

2. The User & Security Model

The objective of the User and Security Model is to provide a set of relationships between the various representations of users, their credentials, the managed elements that represent the resources, and the resource managers involved in system user administration. Thus, the CIM User and Security Model adds to the pre-existing set of requirements for the introduction of a "top" object class in the CIM Core Model. The introduction of CIM_ManagedElement and the associations that reference it (e.g., CIM_Dependency) provide a foundation for the linkages between the User and Security Model and the CIM_ManagedSystemElement derived classes that represent system components and resources.

Users may be members of the groups, or have roles that convey responsibilities or function. We derive **CIM_Group** and **CIM_Role** from the **CIM_Collection class** (which is now a subclass of CIM_ManagedElement). Membership in these collections is explicit (e.g., using the MemberOfCollection association), and – when mapped from an LDAP-accessible repository – the member or RoleOccupant attribute values are used to determine the membership associations.

Similarly, users or collections of users may be part of an organizational structure. The **CIM_OrganizationalEntity** class and its subclasses captures organizational data (such as addresses and phone numbers) and relationships (using the **CIM_OrgStructure** association). For mappings from LDAP-accessible directories, the OrgStructure association may be used to represent the directory information tree (DIT) structure for organization and CIM_OrganizationalUnit instances, and for those group, role, and person instances that are in the DIT of these organizational entities.

As previously discussed, users may be people, or they may be non-human entities – such as a service running as part of an application system – and they may be collections thereof. The User and Security Model factors the user into several classes. There are managed elements that have a user relationship to a system or set of systems (conveyed using the CIM_ElementAsUser association), and two classes that are used to represent the users' access to system resources: **CIM_UsersAccess** and **CIM_Account**.

CIM_UsersAccess is the nexus of a user's system access information, such as credentials and system accounts, independent of the associated element that has access. That is, a managed element such as a **Person** instance might have several user accesses: for example, one could be for an administrative set of authorities in an administrative domain, and another for access for other general business processes (such as routine access of mail). The CIM_UsersAccess class instances, then, provide a user's view of their relationship to the systems with which they interact. The **CIM_ElementAsUser** association is used to convey the "ownership" relationship between the managed element that has access and the CIM_UsersAccess instances. The **ElementID** property provides the name scoping, and the Name property provides a unique label for the users' access instance within the scope of the managed element identified by the ElementID.

CIM_Account, on the other hand, can be used as the nexus of a system's information *about* a user. The **CIM_UsersAccount** association provides the relationship back to the user (for traversals for information such as a person's name or the credentials that may be used for access to the account, etc.). A system instance (e.g., **CIM_ComputerSystem, CIM_AdminDomain, CIM_ApplicationSystem**) provides namespace scoping via the weak aggregation of accounts. Instances of CIM_Account are defined within the scope of their aggregating system. The management of these account instances, however, need not be from a service on that system. **CIM_AccountManagementService** instances may have **CIM_ManagesAccountOnSystem** relationships for accounts on any system and, therefore, **CIM_ManagesAccount** relationships as well. For example, this might occur when the accounts are on an administrative domain and the account management service instances are hosted on a subset of the computers in that administrative domain.

Although not complete in this release of the Model, several classes are defined to provide operational implementation of some security policies. (This is distinct from the specification of a device-independent security policy, or the resulting device-specific configuration of those policies). The **CIM_AuthenticationRequirement** class permits the specification of the credentials, required for authentication, for access to specific target resources. On the other hand, **CIM_AccessControlInformation** permits the specification of authorization policies that match users (subjects) and resources (targets) with a set of permissions (access type, access qualifier, and permission).

When used as a subject, a collection may have several levels of indirection in specifying its members. The **CIM_MemberPrincipal** association is used to specify the level of indirection. The principal may be identified by its CIM_UsersAccess or CIM_Account instance(s), in which case it is that account or user's access that is a subject in the collection. It may be identified by the managed element that has the user's access or account – in which case all CIM_UsersAccess instances and all CIM_Account instances associated with the member managed element are subjects – or it may be identified by a credential management service, in which case all credentials issued by the member service are considered valid subjects for the access control policy.

Credentials have a weak association to their **CIM_CredentialManagementService** which owns and manages the credential lifecycle. Normally, when a credential management service has credentials which it uses for access to other systems or services, one would expect the service to have a CIM_ElementAsUser association to a CIM_UsersAccess, which in turn would have a CIM_UsersCredential association to a credential. **CIM_CAHasPublicCertificate** is an important optimization to this general case that is included in the model to map more closely to the PKI (Public Key Infrastructure) schema documented in the IETF RFC 2587.

Verification services have not been included in this release of the Model, except as a high-level, abstract class (**CIM_VerificationService**). Although it may be used with other authentication service types, the **CIM_AuthenticateForUse** association would normally associate an authentication requirement with the using verification services.

Finally, one of the important relationships between users and systems is the system administrator relationship. With the adoption of the User & Security Model, the

PrimaryOwnerContact and **PrimaryOwnerName** properties in **CIM_System** are augmented with associations that may provide more complete information about the administrators.

3. User & Security Example

The User and Security example in the accompanying file provides a view of how the model may be instantiated. It covers the main points, but does not include an instantiation of every class, property, association, and aggregation.

The example is divided into four parts:

(a) The environment is defined and initialized. The environment is comprised of a white pages part and a system part.

The white pages part defines human user 'John Smith,' with his affiliations to his organization, organizational unit, and group.

The system part defines a scope of administration or administrative domain and a set of services for that domain: computer systems, a certificate authority, a verification service, an authorization service, and an account management service. The relationships between these services in the administrative domain are also defined.

(b) The security policy for the environment is established.

The system administrator role is created for the administrative domain. The domain's authentication policy is created where the authentication service for the domain is the certificate authority. Authorization policy and access control are created for the domain.

(c) A new system administrator is added as a user.

The human user defined in the white pages part is added to the role of system administrator for the domain. Administrative operations create an account, a credential, and access privileges for that system administrator (John Smith) in the administrative domain.

(d) A new application service is added as a user.

An application service is installed, and the authentication and authorization policies are defined for that service. A non-human user is created for that application service. Operations create an account, a credential, and access privileges for that application service in the administrative domain.

The details of this example are found in the file user-security-example.zip. The zip file contains:

- Outline.doc – an outline of the classes and associations that are instantiated for each part of the example.
- Visio.vsd – the Visio 5 drawing of the instantiated classes and associations.

- Visio.pdf – a viewable and printable version of the Visio 5 drawing for the reader who does not have access to the Visio application.
- Example.mof – the master file that includes the class, property, and association definitions used and the instantiated definitions.
- Classes.mof – the new class definitions needed for this example. These are necessary because abstract definitions cannot be instantiated.
- Instances.mof – the class, property, and association definitions instantiated in this example. Aliasing is used in this file.

4. Future Work

Based on implementation feedback, simplification and clarification of the User and Security Model is underway for CIM V2.8.

UsersAccess will be clarified as an "identity" and the credentials needed to authenticate that identity will be defined via policy. AccessControlInformation will be clarified as "privilege", which is granted to subjects (identities, roles or collections of these) regarding target resources. These changes are necessary to individually model (and therefore allow the persistence of) an identity, and also describe whether trust is currently established for that identity. This is not the same as the concept of a User within the semantics of an OrganizationalEntity. And, to better support role-based access control, privileges are being defined. These are not restricted to the scope of a system and extend beyond "access". Therefore, these concepts will be modeled in two new classes and existing schema that overlaps with these definitions will be deprecated.

Also, a simplification of CIM_Person has been requested since that class includes a great deal of white and yellow pages data, in addition to basic 'user' information. The goal is to separate the Person class' user-related information from the white and yellow pages data, and move it up in the object hierarchy. The white and yellow pages data will then remain in Person.

Appendix A - Change History

Version 1.0	29 February 2000	Initial Draft
Version 1.1	10 March 2002	Updated for CIM V2.6
Version 1.2	9 June 2003	Updated for CIM V2.7