Network Working Group	P. Hunt, Ed.
Internet-Draft	Oracle Corporation
Intended status: Informational	September 6, 2012
Expires: March 10, 2013	

TOC

SCIM Directory Services
draft-hunt-scim-directory-00

Abstract

This document describes a directory server that implements the SCIM protocol and schema [, its capabilities and access control model], and optional support for LDAPv3 protocol. This specification extends SCIM from provisioning to a general purpose access protocol in support of data management applications (e.g. self-service systems) and RESTful clients that need read/write access to a directory on the Internet, between domains, or within a cloud.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as “work in progress.”

This Internet-Draft will expire on March 10, 2013.

Table of Contents

---

1. Introduction

TOC

SCIM is a protocol**[I-D.scim-api]** and schema**[I-D.scim-core-schema]** designed for provisioning applications and repositories. SCIM was intended to be implemented by applications to enable a common standard protocol for provisioning. This document describes a SCIM Directory, a general purpose RESTful server that can be used by applications as a repository for shared identity information. Specifically, this document reframes the concepts of a directory server as expressed in **[RFC2251]**, and describes how a "SCIM Directory" may simultaneously support LDAPv3 protocol.

For a directory servers supporting both SCIM and LDAPv3, the document describes how SCIM schema, in particular complex attributes is mapped to the LDAPv3 Data Model. The dual-protocol objective of this specification enables eased migration to RESTful Identity Services and avoids the need to run parallel SCIM and LDAPv3 server infrastructures.

This document will describe the following components:

- Data model for a SCIM Directory
- The basic feature set of a SCIM Directory.
- The access control requirements for a SCIM Directory[TBD].
- [OPTIONAL] support for LDAPv3 clients accessing a directory implementing the SCIM Data Model

[TBD: Directory replication. Access control - data layer vs. http layer]

1.1. Applications and Directories

TOC

For the purpose of this document, two different types of SCIM Provider (or server) are implied: being "application providers" and "directory providers". Whereas an "application" implements the SCIM API and Core Schema specifications, a "directory" has further requirements detailed in this document.

1.2. Requirements Language

TOC

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in **RFC 2119** [RFC2119].

---

2. Data Models

TOC

This specification bases the SCIM data model upon that of the SCIM Core Schema specification**[I-D.scim-core-schema]**. It further extends and defines how an LDAP Model can be applied within a SCIM Directory in order to provide dual protocol (LDAPv3 and SCIM) support.

TOC

2.1. SCIM Model

The SCIM Protocol defines a schema suitable for exchange using JSON data objects exchanged over a REST API. SCIM Core Schema provides minimal core schema for representing resources such as users and groups encompassing common attributes used by cloud providers, PortableContacts, and LDAP directory services. Further and most importantly, SCIM Core Schema provides an extension model upon which more resource types may be defined.

TOC

2.1.1. Object Model

A SCIM Directory stores objects based on an extension model where objects of different types extend a parent object type to create specialized objects such as Users and Groups. As defined in the SCIM Core Schema, all objects are extensions of the SCIM Resource object.

SCIM Object Hierarchy

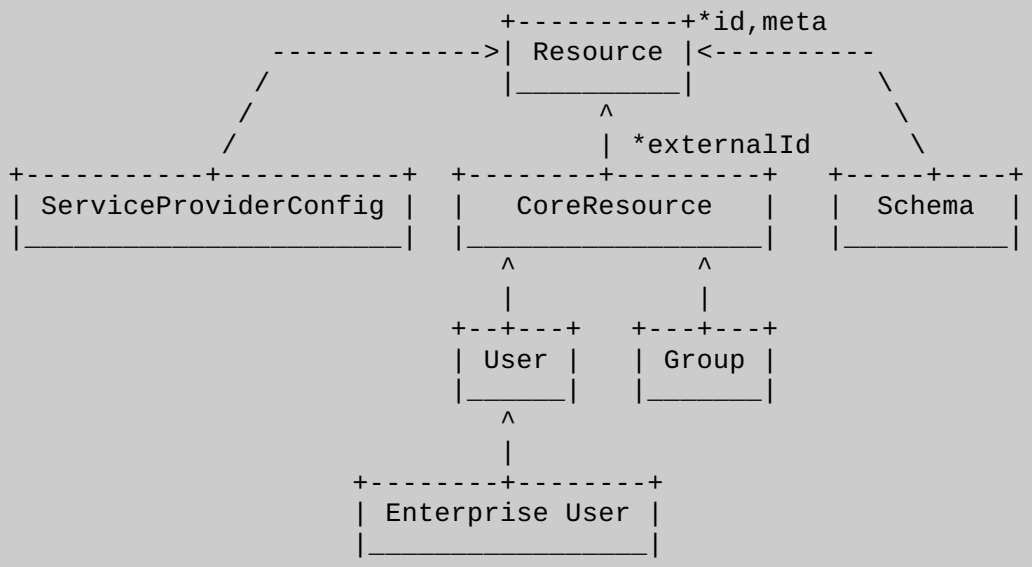

Figure 1

The root "Resource" object defines a couple of attributes to track object identifier ("id") and meta-data associated with the object ("meta"). 3 main objects descend from the "Resouce" object type: ServiceProviderConfig, CoreResource, and Schema. ServiceProviderConfig and Schema are typically used for discovery, providing information about the server and contain no user or group information. CoreResource is the parent object for most entities in a SCIM Directory (e.g. Users and Groups). SCIM Core Schema defines 2 main objects: Users and Groups, and defines EnterpriseUser which is an extension of the User object.

For examples of SCIM objects (e.g. Users and Groups) see the SCIM Core Schema specification **[I-D.scim-core-schema]**.

[TBD add additional resources or entities (e.g. Organizations, OUs, Roles) defined in a SCIM Directory if any]

TOC

2.1.2. Attributes

SCIM defines that Resources contain a collection of attributes identified by one or more object types (or schemas) (e.g. User and EnterpriseUser). Each SCIM Resource is identified and addressed by a unique object identifier or 'id'. The value of the id is always issued by the Service Provider and is a stable, non-reassignable identifier.

Each SCIM resource supports one or more SCIM attributes. SCIM Attribute types consist of:

- String - A sequence of characters.
- Boolean - A literal "true" or "false".
- Decimal - A real number with at least one digit to the left and right of a period.
- Integer - A decimal number with no fractional digits
- Binary - A base64Binary encoded value.
- Complex - A singular or multi-valued attribute whose value is a composition of one or more simple attributes.

A SCIM directory supports multi-valued attributes which are unordered lists of attributes. Each attribute MAY contain sub-attributes (including complex attributes). Multi-valued attributes contain the following normative sub-attributes as defined in SCIM Core Schema **section 3.2** [I-D.scim-core-schema]: type, primary, display, operation, value.

TOC

2.1.3. Directory SCIM Schema Extensions

[Attribute extensions for for IDM Apps (e.g. self service) to do password management and password policy functions. --> Note: not password sync as in provisioning context]

[New Resource a "Credential" - used to hold passwords, X509 certs, etc. Could be defined as sub entity of Users -> e.g. /Users/cn=phil hunt,ou=idm,o=oracle.com/Credentials]

TOC

2.2. LDAP Model

In order to leverage existing data, the SCIM LDAP Model's objective is to represent data in a SCIM Directory as if it were in a LDAPv3 Directory as defined in **[RFC2251]** and in **[RFC2256]** in order to provide backwards support for LDAPv3 clients and to avoid the need to develop parallel LDAPv3 & SCIM infrastructure. Not all features from the SCIM Model are mapped to LDAPv3.The specification provides LDAPv3 compatibility so that existing LDAPv3 clients MAY not need to be updated.

TOC

2.2.1. Syntax Support

While the SCIM Core Schema breaks attribute values into a simple list of types, in many cases the underlying format for both SCIM and LDAP is string data and binary data. In many cases, conversion of the value itself is relatively straight forward. More complex syntaxes such as PostalAddress (OID 1.3.6.1.4.1.1466.115.121.1.41) may require more complex value translation.

TOC

2.2.2. Object Classes

SCIM's Resource Object model works in a similar way to LDAP's object class model. Where an analog mapping between SCIM and LDAP exists, objectclasses SHOULD be mapped to the extent that server schema configuration allows. For example, a SCIM User is mapped to LDAP InetOrgPerson. Or in the case of Microsoft AD, a SCIM User is mapped to an Active Directory User.

Mapping between LDAP Object Classes and SCIM Objects

LDAP Class	SCIM Object
top	Resource
inetOrgPerson	User
groupOfUniqueNames	Group
organization	[TBD]
organizationalUnit	[TBD]

Table 1: Object Mapping

In cases where LDAPv3 objectclass definitions may be in conflict with SCIM schema, the SCIM schema validation SHALL take precedence. Objectclass enforcement for SCIM Directories supporting LDAP is OPTIONAL.

TOC

2.2.3. Attributes

TOC

2.2.3.1. Attribute Type Mapping

LDAP has many more attribute types than SCIM does. The following table lists a set of default syntax mappings between SCIM and LDAP.

Default Mapping between SCIM Types and LDAP Syntax

SCIM Type	LDAP Syntax
String	IA5 String (case-sensitive)
Boolean	Boolean
Decimal	Numeric String
Integer	INTEGER
DateTime	Generalized Time
Binary (base64)	Binary (BER encoded)
Complex	Mapped by individual sub-attribute type.

Table 2: Attribute Type Mapping

TOC

2.2.3.2. Complex Attributes

Complex attributes SHOULD be represented in LDAP in two ways:

1. The complex attribute MAY be mapped to a single LDAP attribute using the "value" sub-attribute if present, OR by using '$' delimited notation whereby sub-attributes are concatenated into a single LDAP value.
2. A complex attribute is represented as a set of LDAP attributes whereby each sub-attribute is referenced by parent.subattribute name format using dotted (".") notation. For example, an LDAP attribute name of address.street would return the Street name sub-attribute value.

TOC

2.2.3.3. Multi-Valued Attributes

SCIM multi-valued attributes MAY be used to map to LDAPv3.

- For multi-valued attribute that contain the sub-attribute "value" (the attribute's significant value), the server SHOULD map these values to a corresponding LDAP attribute value when the LDAP attribute is multi-valued.
- When the LDAP attribute is single-valued, then the SCIM value tagged with the sub-attribute "primary" as "true", SHALL be mapped.
- When the LDAP attribute is single-valued, and when the scim sub-attribute "primary" is not set, the first value in the list SHALL be used.
- If the SCIM attribute does not use the multi-valued attribute sub-attribute standards, the values converted MAY be a '$' delimited concatentation of all sub-attribute fields into a single String per value.

LDAPv3 clients wishing to return a particular attribute value MAY use LDAP "Attribute Description Options" (as described in section 4.1.5 of **[RFC2251]**) to select a SCIM value by SCIM type sub-attribute (e.g. home, work). For example, to return the SCIM work value of "phoneNumbers", an LDAPv3 client would request telephoneNumber;work as the attribute name. An LDAPv3 client would request mail;home to request the home value of SCIM emails attribute.

TOC

2.2.3.4. Attribute Name Mapping

The LDAP Model should maintain a table which allows attributes to be referenced by OID, or by LDAP attribute and defining which SCIM Attribute is the equivalent.

The mapping MAY assume the following defaults:

1. If not defined in LDAP, a SCIM Attribute name SHOULD be useable in LDAP providing there is no naming conflict. Where a conflict exists, the existing LDAP mapping SHALL take precedence over the default.
2. Each complex attribute subattribute SHOULD be useable in LDAP by using the SCIM dotted notation (e.g. address.street).
3. A complex attribute name (e.g. address) SHOULD be accessible in LDAP. If a "value" sub-attribute is defined, the value returned is the "value" sub-attribute (equivalent to address.value). Otherwise, the value returned is a '$' delimeted concatenation of all sub-attributes in the complex attribute.

The following table provides a list of suggested mappings:

SCIM	LDAP
id	dn/distinguishedName
externalId	externalId*
userName	uid
name.formatted	cn
name.familyName	sn (surname)
name.givenName	givenName
name.middleName	initials
name.honorificPrefix	name.honorificPrefix*
name.honorificSuffix	generationQualifier
displayName	displayName
nickName	nickName*
profileUrl	labeledURI
employeeNumber	employeeNumber
userType	employeeType
title	title
manager	manager
preferredLanguage	preferredLanguage
locale	locale*
utcOffset	utcOffset*

costCenter	costCenter*
organization	o
division	ou/organizationalUnit
department	department* [check this]
emails.value (complex)	mail/rfc822address
phonenumber.value (complex)	telephoneNumber
im	im*
photo	jpegPhoto
address / address.formatted [difference?]	postalAddress
address.streetAddress	street
address.locality	l
region	[also defined as locality]
address.postalCode	postalCode
address.country	c
group	memberOf/isMemberOf* [check]
entitlements	entitlements*
roles	nsRoleDn / roles / organizationalRole[?]
x509Certificates	userCertificate
active	active*

Note: aliases separated by "/". * means not defined in LDAP (extension to LDAP)

Table 3: SCIM and LDAP Attribute Names

2.2.4. LDAP Operations

TOC

All LDAP operations remain unchanged and MUST be implemented. As all LDAP operations center around a Distinguished Name (DN), the DN MAY be mapped to the SCIM "Id" field if distinguished names have been used, or it MAY be mappped to externalId. [Should we force mapping to id or some other field?]

The LDAP Bind operation which allows authentication information to be exchanged with the server may need special consideration. The SCIM server implementation MAY choose to pass this exchange through to the authentication system supporting the HTTP Authentication (securing SCIM RESTful access), or it may choose to implement directly by mapping to SCIM attributes. [Not sure this matters. It does not affect inter-op IMHO]

2.2.5. LDAP Controls

TOC

LDAP Controls MAY be supported. [anything we need to cover here?]

3. Server Topology

TOC

A SCIM Directory while appearing to be a single logical server to a client (e.g. through the use of a proxy), MAY be comprised of several servers as supported by HTTP 1.1 and HTTP redirection **RFC2616** [RFC2616].

[Any issues for Proxy or Firewall/Routing servers?]

4. SCIM API Support

TOC

The following OPTIONAL SCIM API components are REQUIRED [or SHOULD] to be implemented in SCIM Directory implementations.

- Filtering
- Sorting
- PATCH operation
- BULK operation

---

5. Authentication & Access Control

TOC

5.1. Authentication

TOC

The SCIM protocol does not directly define authentication. Authentication for SCIM Directory is based entirely on standard HTTP 1.1 authentication models. A SCIM Directory MAY be used as a credential repository for public keys, and secrets, but the protocol itself does not define authentication.

SCIM Directories SHOULD support OAuth2 **[I-D.ietf-oauth-v2]** to support the authentication of client applications accessing SCIM Directory resources on their own or on behalf of an authorizing user.

5.2. Access Control

TOC

[TBD]

The access control requirements for SCIM Directories are specified by **[RFC2820]** and MUST be supported.

[TBD: OAuth2 additional requirements: scope, multiple security contexts (client app and user)]

[TBD: Federated credentials: ACLs should be able to support approved security contexts from credentials not bound to the local directory]

---

6. IANA Considerations

TOC

[TO BE DETERMINED]

---

7. Security Considerations

TOC

[TO BE DETERMINED]

---

8. References

TOC

8.1. Normative References

TOC

[I-D.scim-api] Drake, T., Mortimore, C., Ansari, M., Grizzle, K., and E. Wahlstroem, "**System for Cross-Domain Identity Management:Protocol 1.1**," draft-scim-api-01 (work in progress), August 2012 (**TXT**).

[I-D.scim-core- Mortimore, C., Harding, P., Madsen, P., and T. Drake, "**System for Cross-Domain Identity Management:**

core-schema] Core Schema 1.1," draft-scim-core-schema-01 (work in progress), August 2012 (TXT).

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," BCP 14, RFC 2119, March 1997 (TXT, HTML, XML).

[RFC2251] Wahl, M., Howes, T., and S. Kille, "Lightweight Directory Access Protocol (v3)," RFC 2251, December 1997 (TXT, HTML, XML).

[RFC2252] Wahl, M., Coulbeck, A., Howes, T., and S. Kille, "Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions," RFC 2252, December 1997 (TXT, HTML, XML).

[RFC2256] Wahl, M., "A Summary of the X.500(96) User Schema for use with LDAPv3," RFC 2256, December 1997 (TXT, HTML, XML).

[RFC2820] Stokes, E., Byrne, D., Blakley, B., and P. Behera, "Access Control Requirements for LDAP," RFC 2820, May 2000 (TXT).

8.2. Informative References

TOC

[I-D.ietf-oauth-v2] Hardt, D., "The OAuth 2.0 Authorization Framework," draft-ietf-oauth-v2-31 (work in progress), August 2012 (TXT, PDF).

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1," RFC 2616, June 1999 (TXT, PS, PDF, HTML, XML).

[RFC4513] Harrison, R., "Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms," RFC 4513, June 2006 (TXT).

Appendix A. Acknowledgements

TOC

The author would like to thank the members of the SCIM WG for input to this document. Thanks to Chris Phillips, Kelly Grizzle, Trey Drake, and Paul Madsen for contributions on LDAP vs. SCIM Attribute Naming. Particular thanks to those that participated in the SCIM Directory discussions at IETF Vancouver and more recently by email:

- Bert Greevenbosch, Huawei
- Alexey Melnikov, Isode
- Prateek Mishra, Oracle
- Chuck Mortimore, Salesforce.com
- Tony Nadalin, Microsoft

Author's Address

TOC

Phil Hunt (editor)
Oracle Corporation
Vancouver
CA
Email: phil.hunt@yahoo.com