

Network Working Group  
Internet Draft  
Intended status: Proposed Standard  
Expires: August 2013

M. Wahl  
Microsoft  
February 18, 2013

SCIM Profile For Enhancing Just-In-Time Provisioning  
draft-wahl-scim-jit-profile-00.txt

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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>

This Internet-Draft will expire on August 18, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

## Abstract

This document specifies a profile of the System for Cross-Domain Identity Management Protocol (SCIM) for use by servers which rely upon just-in-time provisioning patterns in a protocol (such as SAML) to create user accounts, and need an additional channel to be notified of changes to user accounts.

## Table of Contents

1. Introduction.....	2
1.1. Conventions used in this document.....	4
2. Events in the SCIM client database.....	4
2.1. User is added to the SCIM client database.....	4
2.2. User's username changes.....	4
2.3. User's display name changes.....	4
2.4. User's account is disabled.....	5
2.5. User's account is re-enabled.....	5
2.6. User's account is purged.....	6
3. SCIM interaction profiles.....	6
3.1. Locating a user by their user name.....	6
3.2. Modifying a user.....	7
3.3. Deleting a user.....	8
4. Schema Profile.....	8
5. Security Considerations.....	9
6. IANA Considerations.....	9
7. References.....	9
7.1. Normative References.....	9
7.2. Informative References.....	9

## 1. Introduction

The SCIM protocol [1] is an application-level, REST protocol for provisioning and managing identity data on the web. SCIM can be leveraged for numerous use cases, including transfer of attributes to a relying party web site (see [4] section 3.6).

This profile of SCIM illustrates the interactions between a SCIM client and a SCIM server, in the following scenario:

- o The SCIM client has an associated database (SCIM client database) of user records, and that SCIM client database is leveraged by an identity provider for user authentication.

- o The SCIM server has a different associated database (SCIM server database) of user records, and that SCIM server database is leveraged by a service provider (an application).
- o The service provider trusts the identity provider to authenticate users, and a user's username and other attributes as stored in the SCIM client database are transferred (using a federation or authentication protocol such as SAML -- not SCIM) from the identity provider to the service provider each time a user accesses the service provider.
- o When the service provider receives a user identity from the identity provider in the federation/authentication protocol, and the service provider cannot find a user record with matching username in the SCIM server database, then the service provider creates a new record in the SCIM server database.
- o An identity management system associated with the SCIM client database makes changes to users in the SCIM client database, for instance to de-activate a user, or change the user's display names. These changes are of interest to the service provider.

This profile enables the SCIM client to notify the SCIM server of changes to users in the SCIM client database, so that the SCIM server can make corresponding changes in the SCIM server database, which are then available to the service provider. For example, if the identity provider deletes a user, this deletion event can be transferred to the service provider via SCIM, so that the service provider can clean up any data associated with a user who won't be accessing that service provider again. Or if the user changes their username, then this can be made known to the service provider, so that subsequent requests by that user will be associated to the same account in the SCIM server database.

This profile is not intended to be a comprehensive replication protocol; instead, it provides basic consistency for user records for in two domain's databases, for all users who choose to access the service provider. This profile also explicitly does not create users via SCIM, which is assumed to occur out of band from SCIM, such as through a "just in time" operation in a federation protocol such as SAML [5]. This profile also does not cover establishing common index keys of usernames between a SCIM client and a SCIM server. Finally, management of other object types besides users, and additional attributes beyond basic user status and name, is outside the scope of this profile.

### 1.1. Conventions used in this document

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 [2].

## 2. Events in the SCIM Client Database

A SCIM client will, either upon specific change in the SCIM client database, or at intervals, provide one or more changes to the SCIM server.

### 2.1. User is added to the SCIM client database

The SCIM client does not notify the SCIM server of this event. (In this profile, user creation is assumed to occur out of band from SCIM, such as through a "just in time" operation in a federation protocol such as SAML [5].)

### 2.2. User's username changes

When a user's username changes in the SCIM client database, then the SCIM client will perform the following procedure.

- o The SCIM client will attempt to locate the user in the SCIM server using the old username, as described in section 3.1 of this document.
- o If the user could not be located (no matching record is returned from the GET request), then the procedure ends.
- o Otherwise, if the user's record was found in the SCIM server, then the SCIM client will send a patch, as described in section 3.2 of this document, to set the value of the username attribute to the new username.
- o If the SCIM server returns a 400-series error indication from the patch, then the SCIM client SHOULD NOT retry the operation.

### 2.3. User's display name changes

When a user's display name changes in the SCIM client database, then the SCIM client will perform the following procedure.

- o The SCIM client will attempt to locate the user in the SCIM server using the user's username, as described in section 3.1 of this document.

- o If the user could not be located (no matching record is returned from the GET request), then the procedure ends.
- o Otherwise, if the user's record was found in the SCIM server, then the SCIM client will send a patch, as described in section 3.2 of this document, to set the value of the displayName attribute, and OPTIONALLY the value of the name attribute, to the new name.
- o If the SCIM server returns a 400-series error indication from the patch, then the SCIM client SHOULD NOT retry the operation.

#### 2.4. User's account is disabled

When a user's account is disabled in the SCIM client database, then the SCIM client will perform the following procedure.

- o The SCIM client will attempt to locate the user in the SCIM server using the user's username, as described in section 3.1 of this document.
- o If the user could not be located (no matching record is returned from the GET request), then the procedure ends.
- o If the user's record was found in the SCIM server, and the GET returned the "active" attribute type in that record and that attribute had the value false, then the procedure ends.
- o Otherwise, then the SCIM client will send a patch, as described in section 3.2 of this document, to set the value of the active attribute to false.
- o If the SCIM server returns a 400-series error indication from the patch, then the SCIM client SHOULD NOT retry the operation.

#### 2.5. User's account is re-enabled

When a user's account is re-enabled in the SCIM client database after having previously been disabled, then the SCIM client will perform the following procedure.

- o The SCIM client will attempt to locate the user in the SCIM server using the user's username, as described in section 3.1 of this document.
- o If the user could not be located (no matching record is returned from the GET request), then the procedure ends.

- o If the user's record was found in the SCIM server, and the GET returned the "active" attribute type in that record and that attribute had the value true, then the procedure ends.
- o Otherwise, then the SCIM client will send a patch, as described in section 3.2 of this document, to set the value of the active attribute to true.
- o If the SCIM server returns a 400-series error indication from the patch, then the SCIM client SHOULD NOT retry the operation.

#### 2.6. User's account is purged

When a user's account is purged in the SCIM client, then the SCIM client will perform the following procedure.

- o The SCIM client will attempt to locate the user in the SCIM server using the user's username, as described in section 3.1 of this document.
- o If the user could not be located (no matching record is returned from the GET request), then the procedure ends.
- o Otherwise, then the SCIM client will send a delete, as described in section 3.3 of this document.
- o If the SCIM server returns a 400-series error indication from the delete, then the SCIM client SHOULD NOT retry the operation.

### 3. SCIM Interaction Profile

#### 3.1. Locating a user by their user name

In order to modify or delete a user record in a SCIM server, the SCIM client needs to first discover the id of that record as stored in the SCIM server. This is done by searching for the userName attribute, which is defined in section 6.1 of the SCIM Core Schema [3].

The client can issue a SCIM query request for the namespace ending with /Users with a query parameter of a filter for userName matching the user name. For example (lines wrapped for clarity):

```
GET /TBD/scimbase/Users?filter=username%20eq%20%22matt@example.co
m%22 HTTP/1.1
Host: example.com
Accept: application/json
Authorization: Bearer deadbeef
```

If found, the server will respond with a HTTP 200 message containing a single result resource, with one or more attributes:

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "totalResults":1,
  "schemas":["urn:scim:schemas:core:1.0"],
  "Resources":[
    {
      "id":"2819c223-7f76-453a-919d-413861904646,
...      "userName":"matt@example.com"
    }
  ]
}
```

If not found, the SCIM server will respond with a HTTP 200 message containing zero result resources.

### 3.2. Modifying a user

For this interaction, the SCIM client needs the id of the user. If it does not already have it, it can obtain the id of the user as described in section 3.1.

If the SCIM client can locate the user record, then the client can modify the attributes of the user in the SCIM server by issuing a POST with an override to PATCH. The body of the message will be a JSON structure with a "schemas" key, and one or more keys. For example (lines wrapped for clarity)

```
POST /TBD/scimbase/Users/acbf3ae7-8463-4692-b4fd-9b4da3f908ce
HTTP/1.1
Host: example.com
Accept: application/json
Content-Type: application/json
Authorization: Bearer deadbeef
X-HTTP-Method-Override: PATCH
Content-Length: 72

{
  "schemas": ["urn:scim:schemas:core:1.0"],
  "displayName": "Babs Jensen"
}
```

### 3.3. Deleting a user

For this interaction, the SCIM client needs the id of the user. If it does not already have it, it can obtain the id of the user as described in section 3.1.

If the SCIM client can locate the user record, then the client can request deletion of user in the server by issuing a POST with an override to DELETE. For example (lines wrapped for clarity)

```
POST /TBD/scimbase/Users/acbf3ae7-8463-4692-b4fd-9b4da3f908ce
HTTP/1.1
Host: example.com
Authorization: Bearer deadbeef
X-HTTP-Method-Override: DELETE
```

If the user cannot be found, the server will return error code 404.

## 4. Schema Profile

A server that implements this profile is REQUIRED to recognize and store in its database the "username" attribute of the SCIM User Schema. (This attribute is described in section 6 of the SCIM Schema [3].)

The SCIM server MUST recognize and SHOULD store the "displayName", "active" and "name" (with components "givenName" and "familyName") attributes of the SCIM User Schema. If the SCIM server does not store one or more of those attributes, then any changes to them requested by the SCIM client SHOULD be silently discarded.



The SCIM server MUST recognize the "schemas" attribute of the SCIM Core Schema (in section 5 of the SCIM Schema [3]), but the value is not modified by the SCIM client in this profile.

The password attribute is not used in this profile.

## 5. Security Considerations

As described in the SCIM protocol security considerations [1], the SCIM interactions are to be protected using TLS.

Note that for each of the interactions in section 3, the SCIM client will need an OAuth bearer token. Such tokens are normally short-lived (hours). How the SCIM client locates an OAuth endpoint to obtain this token is currently outside the scope of this document.

## 6. IANA Considerations

There are no IANA considerations in this document.

## 7. References

### 7.1. Normative References

- [1] Drake, T., Mortimore, C., Ansari, M., Grizzle, K., Wahlstroem, E., "System for Cross-Domain Identity Management:Protocol", draft-ietf-scim-api-00, August 2012.
- [2] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [3] Mortimore, C., Harding, P., Madsen, P., Drake, T., "System for Cross-Domain Identity Management:Core Schema", draft-ietf-scim-core-schema-00, August 2012.

### 7.2. Informative References

- [4] Hunt, P., Khasnabish, B., Nadalin, A., Zeltsan, Z., Li, K., "SCIM Use Cases", draft-zeltsan-scim-use-cases-01, February 2013.
- [5] Cantor, S., Kemp, J., Philpott, R., Maler, E., "Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0", <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf> , March 2005.

Authors' Addresses

Mark Wahl

Microsoft Corporation  
1 Microsoft Way  
Redmond WA 98052 USA

Email: [mark.wahl@microsoft.com](mailto:mark.wahl@microsoft.com)

