Abstract

The System for Cross-Domain Identity Management (SCIM) specifications define a profile of HTTP protocol and a schema that enable managing identities in cross-domain scenarios. This specification extends SCIM protocol resource retrieval and query functions to enable paging of SCIM resources that contain large complex multi-valued attributes such as SCIM Groups.

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 https://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 February 8, 2020.

Copyright Notice

Copyright (c) 2019 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 (https://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. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of
1. Introduction and Overview

SCIM Protocol [RFC7644] is an application-level, HTTP protocol for provisioning and managing identity data on the web and in cross-domain environments such as enterprise to cloud, or inter-cloud scenarios. The protocol supports creation, modification, retrieval, and discovery of core identity resources such as Users and Groups, as well as custom resources and resource extensions.

The definition of resources, attributes, and overall schema are defined in the SCIM Core Schema document (see [RFC7643]).

This specification extends SCIM resource retrieval and query functions to enable paging of SCIM resources that may contain attributes with large numbers of values. For example, a SCIM Group may contain thousands or millions of members.

1.1. Intended Audience

This document is intended as a guide to extend SCIM protocol usage for both SCIM HTTP service providers and HTTP clients who may provision information to service providers or retrieve information from them.

1.2. Notational Conventions

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 [RFC2119]. These
keywords are capitalized when used to unambiguously specify
requirements of the protocol or application features and behavior
that affect the interoperability and security of implementations.
When these words are not capitalized, they are meant in their
natural-language sense.

For purposes of readability examples are not URL encoded.
Implementers MUST percent encode URLs as described in Section 2.1 of
[RFC3986].

Throughout this document all figures may contain spaces and extra
line-wrapping for readability and space limitations. Similarly, some
URI’s contained within examples, have been shortened for space and
readability reasons.

1.3. Definitions

This specification uses the definitions from the SCIM Schema
Specification [RFC7643] and the SCIM Protocol Specification
[RFC7644].

2. Multi-Value Paging Extension

Detecting the availability of multi-valued attribute paging extension
is covered in Section 3.

When supported, returned values for multi-valued attributes can be
filtered or paged using filters and/or paging parameters appended to
attributes specified in the SCIM "attributes" parameter. Attributes
listed in the attributes parameter MAY be appended with value
qualifiers using square brackets("[ ]") that contains a "valFilter"
(see Figure 1 [RFC7644]), paging parameters (see Section 3.9
[RFC7644]), or a combination of both separated by the ";" character.

In order to qualify specific attributes without changing the default
list of attributes returned for a query, an asterix "*" MAY be used
in the attributes parameter to indicate the default set of attributes
is to be returned in addition to any specific attributes listed. For
example: "attributes=*,members[type eq "user"]" specifies all default
attributes are to be returned and only values of "members" which have
"type" set to "user".

When an attribute has a multi-value filter or paging qualifier, the
service provider SHALL include additional "meta" sub-attributes (see
Section 3.1 of [RFC7643]). The name of the multi-valued attribute
plus the String "cnt" is used to indicate the count of attribute
values available expressed as an Integer (see Section 2.3.4 of
[RFC7643]). When a "valFilter" expression is used, the number SHALL
indicate the total number of matches that may be returned based on
the filter. When no filter expression is specified, the number SHALL
indicate the total number of values. For an example, see
"emails.cnt" in Figure 2. This count indicates that there is only one
value with "type" equal to "work ".

When "startIndex" is used as an attribute paging qualifier and the
value is greater than the number of values, the server SHALL omit the
attribute from the result to indicate no values exist at that index.

In the following example, a user is returned, but only "work" emails
are to be returned.

GET /Users/2819c223-7f76-453a-919d-413861904646?
   attributes=*,emails[type eq "work"]
Host: example.com
Accept: application/scim+json
Authorization: Bearer h480djs93hd8

Figure 1: Using a filter to return only work email values
The service provider responds with:

HTTP/1.1 200 OK
Content-Type: application/scim+json
Location: https://example.com/v2/Users/2819c223-7f76-453a-919d-413861904646
ETag: W/"f250dd84f0671c3"

```json
{
    "schemas": ["urn:ietf:params:scim:schemas:core:2.0:User"],
    "id": "2819c223-7f76-453a-919d-413861904646",
    "externalId": "bjensen",
    "meta": {
        "resourceType": "User",
        "created": "2011-08-01T18:29:49.793Z",
        "lastModified": "2011-08-01T18:29:49.793Z",
        "location": "https://example.com/v2/Users/2819c223-7f76-453a-919d-413861904646",
        "version": "W/"f250dd84f0671c3"",
        "emails.cnt": 1
    },
    "name": {
        "formatted": "Ms. Barbara J Jensen III",
        "familyName": "Jensen",
        "givenName": "Barbara"
    },
    "userName": "bjensen",
    "phoneNumbers": [
    {
        "value": "555-555-8377",
        "type": "work"
    }
    ],
    "emails": [
    {
        "value": "bjensen@example.com",
        "type": "work"
    }
    ]
}
```

Figure 2: Response with filtered emails attribute
In the following example, all Groups are searched and only Groups whose name starts with "Group" are selected. Additionally, the members attribute values are filtered to return only member values with "type" equal to "groups" (as in sub-groups) returning only the first 5 values using the attributes paging qualifying parameters.

GET /v2/Groups?filter=displayName sw 'Group'\&attributes=*,members[type eq "Group"]\&count=5\&startIndex=1

Host: example.com
Accept: application/scim+json
Authorization: Bearer h480djs93hd8

Figure 3: Querying multiple groups with attribute qualifiers

The server responds with 2 matched resources. The first resource only has one Group member value, while the second resource has 7 member values and has been limited to the first 5 members per the "count" paging parameter.

HTTP/1.1 200 OK
Content-Type: application/scim+json

{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:ListResponse"],
  "totalResults": 2,
  "Resources": [
    {
      "id": "c3a26dd3-27a0-4dec-a2ac-ce211e105f97",
      "schemas": ["urn:ietf:params:scim:schemas:core:2.0:Group"],
      "displayName": "Group A",
      "meta": {
        "resourceType": "Group",
        "created": "2011-08-01T18:29:49.793Z",
        "lastModified": "2011-08-01T18:29:51.135Z",
        "location": "https://example.com/v2/Groups/c3a26dd3-27a0-4dec-a2ac-ce211e105f97",
        "version": "W\"mvwNGaxB5SDq074p\"",
        "members.cnt": 1
      },
      "members": [
        {
          "value": "6c5bb468-14b2-4183-baf2-06d523e03bd3",
          "$ref": "https://example.com/v2/Groups/6c5bb468-14b2-4183-baf2-06d523e03bd3",
          "type": "Group"
        }
      ]
    },
    {
      "id": "c3a26dd3-27a0-4dec-a2ac-ce211e105f97",
      "schemas": ["urn:ietf:params:scim:schemas:core:2.0:Group"],
      "displayName": "Group B",
      "meta": {
        "resourceType": "Group",
        "created": "2011-08-01T18:29:49.793Z",
        "lastModified": "2011-08-01T18:29:51.135Z",
        "location": "https://example.com/v2/Groups/c3a26dd3-27a0-4dec-a2ac-ce211e105f97",
        "version": "W\"mvwNGaxB5SDq074p\"",
        "members.cnt": 1
      },
      "members": [
        {
          "value": "6c5bb468-14b2-4183-baf2-06d523e03bd3",
          "$ref": "https://example.com/v2/Groups/6c5bb468-14b2-4183-baf2-06d523e03bd3",
          "type": "Group"
        }
      ]
    }
  ]
}
{  
  "id": "6c5bb468-14b2-4183-baf2-06d523e03bd3",  
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:Group"],  
  "displayName": "Group B",  
  "meta": {  
    "resourceType": "Group",  
    "created": "2011-08-01T18:29:50.873Z",  
    "lastModified": "2011-08-01T18:29:50.873Z",  
    "location": "https://example.com/v2/Groups/6c5bb468-14b2-4183-baf2-06d523e03bd3",  
    "version": "W\"wGB85s2QJMjiNnuI\"",  
    "members.cnt":7  
  },  
  "members": [  
    {  
      "value": "c3a26dd3-27a0-4dec-a2ac-ce211e105f97",  
      "$ref": "https://example.com/v2/Groups/c3a26dd3-27a0-4dec-a2ac-ce211e105f97",  
      "type": "Group"  
    }  
  ]}
In Figure 3 the client may observe that the number of matches available for the second Group (whose "id" is "6c5bb468-14b2-4183-baf2-06d523e03bd3") is 7. In Figure 4, the client may return the second page, by repeating the query with "startIndex" set to 6.

In the following example, paging of member values of a specific group is requested.

GET /v2/Groups/6c5bb468-14b2-4183-baf2-06d523e03bd3?attributes=*,members[type eq "Group"]&count=5&startIndex=6

Host: example.com
Accept: application/scim+json
Authorization: Bearer h480djs93hd8

Figure 5: Query returning the second page of values for an attribute
HTTP/1.1 200 OK
Content-Type: application/scim+json
Location: https://example.com/v2/Groups/e9e30dba-f08f-4109-8486-d5c6a331660a
ETag: W/"lha5bbazU3fNvfe5"

{
    "id": "6c5bb468-14b2-4183-baf2-06d523e03bd3",
    "schemas": ["urn:ietf:params:scim:schemas:core:2.0:Group"],
    "displayName": "Group B",
    "meta": {
        "resourceType": "Group",
        "created": "2011-08-01T18:29:50.873Z",
        "lastModified": "2011-08-01T18:29:50.873Z",
        "location": "https://example.com/v2/Groups/6c5bb468-14b2-4183-baf2-06d523e03bd3",
        "version": "W/"wGB85s2QJMjiNnuI",
        "members.cnt": 7
    },
    "members": ["value": "596ec090-2f66-4d3e-ad4c-68d9ac05ad53",
                 "$ref": "https://example.com/v2/Groups/596ec090-2f66-4d3e-ad4c-68d9ac05ad53",
                 "type": "Group"
               ]
}

Figure 6: Returning the second page of values for an attribute

3. Service Provider Configuration Feature Discovery

Multi-value paging support may be determined by querying the "/ServiceProviderConfig" endpoint and looking up the Boolean attribute "mvpaging" indicating support for multi-valued paging.
4. Security Considerations

To be completed

5. Privacy Considerations

To be completed.

6. IANA Considerations

No IANA considerations.

7. Normative References


Appendix A. Acknowledgments

The editors would like to acknowledge the contribution and work of the past draft editors:

Appendix B. Change Log

[[This section to be removed prior to publication as an RFC]]

Draft 00 - PH - Initial draft
Authors’ Addresses

Phil Hunt (editor)
Oracle Corporation

Email: phil.hunt@yahoo.com

Gregg Wilson
Oracle Corporation

Email: gregg.wilson@oracle.com