Workgroup: SCIM
Updates: 7643, 7644 (if approved)
Published: 28 September 2022
Intended Status: Informational
Expires: 1 April 2023
Authors: M. Peterson One Identity
D. Zollner Microsoft

Cursor-based Pagination of SCIM Resources

Abstract

This document defines additional SCIM query parameters and result attributes to allow use of cursor-based pagination in SCIM implementations that are implemented with existing code bases, databases, or APIs where cursor-based pagination is already well-established.

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 1 April 2023.

Table of Contents

1. Introduction

The two common patterns for result pagination in HTTP-based protocols are index-based pagination and cursor-based pagination. Rather than attempt to compare and contrast the advantages and disadvantages of competing pagination patterns, this document simply recognizes that SCIM service providers are commonly implemented as an interoperability layer on top of already existing application codebases, databases, and/or APIs that already have a well-established pagination pattern.

Translating from an underlying cursor-based pagination pattern to the index-based pagination defined in Section 3.4.2.4 of [RFC7644] ultimately requires the SCIM service provider to fully iterate the underlying cursor, store the results, and then serve indexed pages from the stored results. This task of "pagination translation" dramatically increases complexity and memory requirements for implementing a SCIM Service Provider, and may be an impediment to SCIM adoption for some applications and identity systems.

This document defines a simple addition to the SCIM protocol that allows SCIM service providers to reuse underlying cursors without expensive translation. Support for cursor-based pagination in SCIM encourages broader cross-application identity management interoperability by encouraging SCIM service provider implementations for applications and identity systems where cursor-based pagination is already well-established.

1.1. 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].

2. Query Parameters and Response Attributes

The following table describes the URL pagination parameters requests for using cursor-based pagination:

Parameter	Description
cursor	The string value of the nextCursor attribute from a previous result page. The cursor value MUST be empty or omitted for the first request of a cursor-paginated query.
count	A positive integer. Specifies the desired maximum number of query results per page, e.g., count=10. When specified, the service provider MUST NOT return more results than specified, although it MAY return fewer results. If count is not specified in the query, the maximum number of results is set by the service provider.

Table 1

The following table describes cursor-based pagination attributes returned in a paged query response:

Element	Description
nextCursor	A cursor value string that MAY be used in a subsequent request to obtain the next page of results. Service providers supporting cursor-based pagination MUST include nextCursor in all paged query responses except when returning the last page. nextCursor is omitted from a response only to indicate that there are no more result pages.
previousCursor	A cursor value string that MAY be used in a subsequent request to obtain the previous page of results. Use of previousCursor is OPTIONAL. Service Providers that are unable to support a previousCursor MAY omit previousCursor when sending paged query responses.

Table 2

For example, to retrieve the first 10 Users, use an empty cursor and set the count to 10:

```
GET /Users?cursor&count=10
Host: example.com
Accept: application/scim+json
Authorization: Bearer U8YJcYYRMjbGeepD
```

The response to the query above returns metadata regarding pagination similar to the following example (actual resources removed for brevity):

```
{
  "totalResults":100,
  "itemsPerPage":10,
  "nextCursor":"VZUTiyhEQJ94IR",
  "schemas":["urn:ietf:params:scim:api:messages:2.0:ListResponse"],
  "Rsesources":[{
     ...
   }]
}
```

Given the example above, to continue pagination, set the cursor to the value of nextCursor ("VZUTiyhEQJ94IR") and re-fetch:

```
GET /Users?cursor=VZUTiyhEQJ94I&count=10
Host: example.com
Accept: application/scim+json
Authorization: Bearer U8YJcYYRMjbGeepD
```

If a Service Provider encounters an invalid cursor or count value (or other error condition), the Service Provider SHOULD return appropriate HTTP response status code and JSON detail error response as defined in Section 3.1.2 of [RFC7644].

2.1. Cursors as the Only Pagination Method

A SCIM Service Provider MAY require cursor-based pagination to retrieve all results for a query by including a "nextCursor" value in the response even when the original query does not include the "cursor" parameter.

For example:

```
GET /Users
Host: example.com
Accept: application/scim+json
```

The SCIM Service Provider may responded to the above query with a single page of results and a "nextCursor" value as shown in the below example (Resources omitted for brevity):

```
{
  "totalResults":5000,
  "itemsPerPage":100,
  "nextCursor":"HPq72Pax3JUaNa",
  "schemas":["urn:ietf:params:scim:api:messages:2.0:ListResponse"],
  "Resources":[{
     ...
   }]
}
```

3. Querying Resources Using HTTP POST

Section 3.4.2.4 of [RFC7644] defines how clients MAY execute the HTTP POST verb combined with the "/.search" path extension to issue execute queries without passing parameters on the URL. When using "./search", the client would pass the parameters defined in Section 2

```
POST /User.search
Host: example.com
Accept: application/scim+json
Authorization: Bearer U8YJcYYRMjbGeepD
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:SearchRequest"],
  "attributes": ["displayName", "userName"],
  "filter":
     "displayName sw \"smith\"",
  "cursor": "",
  "count": 10
}
```

Which would return a result containing a "nextCursor" value which may be used by the client in a subsequent call to return the next page of resources

```
{
  "totalResults":100,
  "itemsPerPage":10,
  "nextCursor":"VZUTiyhEQJ94IR",
  "schemas":["urn:ietf:params:scim:api:messages:2.0:ListResponse"],
  "Resources":[{
     ...
   }]
}
```

4. Service Provider Configuration

The /ServiceProviderConfig resource defined in Section 4 of [RFC7644] facilitates discovery of SCIM service provider features. A SCIM Service provider implementing cursor-based pagination SHOULD include the following additional attribute in JSON document returned by the /ServiceProviderConfig endpoint:

pagination A complex type that indicates pagination configuration options. OPTIONAL.

cursor A Boolean value specifying support of cursor-based paginations. REQUIRED.

index A Boolean value specifying support of index-based pagination. REQUIRED.

Before using cursor-based pagination, a SCIM client MAY fetch the Service Provider Configuration document from the SCIM service provider and verify that cursor-based pagination is supported.

For example:

```
GET /ServiceProviderConfig
Host: example.com
Accept: application/scim+json
```

A service provider supporting both cursor-based pagination and index-based pagination would return a document similar to the following (full ServiceProviderConfig schema defined in Section 5 of [RFC7643] has been omitted for brevity):

```
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig"],

    ...

  "pagination": {
     "cursor": true,
     "index": true
  },

  ...

 }
```

5. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC7643] Hunt, P., Ed., Grizzle, K., Wahlstroem, E., and C. Mortimore, "System for Cross-domain Identity Management: Core Schema", RFC 7643, DOI 10.17487/RFC7643, September 2015, <https://www.rfc-editor.org/info/rfc7643>.

[RFC7644] Hunt, P., Ed., Grizzle, K., Ansari, M., Wahlstroem, E., and C. Mortimore, "System for Cross-domain Identity Management: Protocol", RFC 7644, DOI 10.17487/RFC7644, September 2015, <https://www.rfc-editor.org/info/rfc7644>.

Authors' Addresses

Matt Peterson
One Identity

Email: matt.peterson@oneidentity.com

Danny Zollner
Microsoft

Email: danny.zollner@microsoft.com