SCIM
Internet-Draft
Updates: 7643, 7644 (if approved)
Intended status: Standards Track
Expires: 26 April 2026

P. Hunt, Ed.
Independent Id
N. Cam-Winget
Cisco Systems
M. Kiser
Sailpoint
J. Schreiber
Workday
23 October 2025

SCIM Profile for Security Event Tokens
draft-ietf-scim-events-15

Abstract

This specification defines a set of System for Cross-domain Identity Management (SCIM) Security Events using the Security Event Token Specification to enable the asynchronous exchange of messages between SCIM Service Providers and receivers.

This draft updates RFC7643 defining additional attributes for "urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig" schema and updates RFC7644 with optional new Asynchronous SCIM Request capability.

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 26 April 2026.

Copyright Notice

Table of Contents

1. Introduction and Overview

This specification defines Security Events for SCIM Service Providers and receivers as specified by the Security Event Tokens (SET) [RFC8417]. SCIM Security Events in this specification include: asynchronous request completion, resource replication, and provisioning co-ordination.

This specification defines the use of the HTTP Header "Prefer: respond-async" [RFC7240] to allow a SCIM Protocol Client [RFC7644] to request an asynchronous response (see Section 2.5.1.1).

Using HTTP protocol, a SCIM Protocol Client issues commands to a SCIM Service Provider using HTTP methods such as POST, PATCH, and DELETE [RFC7644] that cause a state change to a SCIM Resource. When multiple independent SCIM Clients update SCIM Resources, individual clients become out of date as state changes occur. Some clients may need to be informed of these changes for co-ordination or reconciliation purposes. This could be done using periodic SCIM GET requests over time, but this rapidly becomes problematic as the number of changes and the number of resources increases.

SCIM Events can be shared over an established Event Feed enabling receivers to monitor and trigger independent asynchronous action. This approach enables greater scale and timeliness, where only changed information is exchanged between parties.

A SET conveys information about a state change that has occurred at a SCIM Service Provider. That SET may be of interest to one or more receivers. But instead of interpreting SETs as commands, each Event Receiver is able to determine the best local follow-up action to take within its own context. For example, a receiver can reconcile schema and resource type differences between domains.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2. Notational Conventions

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

1.3. Definitions

This specification uses definitions from the following specifications:

* JSON Web Tokens (JWT) [RFC7519],

* Security Event Tokens (SET) [RFC8417], and

* System for Cross-Domain Identity Management Protocol [RFC7644].

In JSON Web Tokens and Security Event Tokens, the term "claim" refers to JSON attribute values contained in a JSON Web Token [RFC7519] structure. The term "claim" in tokens is used to indicate that an attribute value may not be verified and its accuracy can be questioned. In the context of SCIM, this distinction is not made. For this specification the terms "claims" and "attributes" are inter-changeable. For consistency, JWT and SET IANA registered attributes will continue to be called claims, while event information attributes (i.e., those in an event payload) will be referred to as attributes.

Additionally, the following terms are defined:

Attributes and Claims
: The JWT specification [RFC7519] upon which SET is based uses the term "claims" to refer to attributes in a JSON token. SCIM in contrast uses the term "attributes" to refer to JSON attributes. For the purposes of this draft, the terms "attributes" and "claims" are equivalent.

Co-ordinated Provisioning (CP)
: Defined in Appendix A.2, in co-ordinated provisioning relationships, an Event Publisher and Receiver typically exchange resource change events without exchanging data (see Section 2.4). For a receiver to know the value of the data, the Event Receiver usually calls back to the SCIM Event Publisher domain to receive a new copy of data (e.g., Uses a SCIM GET request).

Domain Based Replication (DBR)
: Defined in Appendix A.1, in this domain-based replication mode there is an administrative relationship spanning multiple

operational domains, data shared in Events typically uses the "full" mode variation of change events (see Section 2.4) including the "data" payload attribute. This eliminates the need for a callback to retrieve additional data.

Event Feed / Event Stream
: An Event Feed (equivalently Event Stream) is a logical series of events shared with a unique receiving client. A SET transfer (see [RFC8935] and [RFC8936]) Service Provider may offer to allow Event Receivers to "subscribe" to specific event types or events about specific resources (see Feed Management events in Section 2.3).

Event Receiver
: An entity receives events typically via [RFC8935], [RFC8936], or HTTP GET (see Section 2.5.1.1). In the case of SET Push Transfer [RFC8935], the Event Receiver is an HTTP Service Endpoint that receives requests. In the case of SET Poll-Based Transfer [RFC8936], the receiver is an HTTP client that initiates HTTP request to an Event Publisher endpoint.

Event Publisher
: A system that issues SETs based on a resource state change that has occurred at a SCIM Service Provider. For example, events may be the result of a SCIM Create, Modify, or Delete as defined in Section 3 of [RFC7644]. A SCIM Service Provider may be an Event Publisher or an independent service that aggregates events into Event Receiver feeds. As described above, when using [RFC8935], the Event Publisher is an HTTP Client that initiates HTTP POST requests to a defined Event Receiver endpoint. When using [RFC8936], the Event Publisher provides an HTTP endpoint which a receiver may use to "poll" for Security Events.

SCIM Client
: Refers to an HTTP client that initiates SCIM Protocol [RFC7644] requests and receives responses which may cause SCIM Events to be issued by the SCIM Service Provider. A SCIM Client may also be an Event Receiver, typically when making an asynchronous SCIM request (see Section 2.5.1.1).

SCIM Service Provider
: An HTTP server that implements SCIM Protocol [RFC7644] and SCIM Schema [RFC7643]. Upon processing a state change to a SCIM Resource, issues a SCIM Event or causes an Event Publisher to issue a SCIM Event.

SET
: Abbreviation for "Security Event Token" as defined in [RFC8417]

2. SCIM Events

A SCIM event is a signal, in the form of a Security Event Token [RFC8417], that describes some event that has occurred. A SET event consists of a set of standard JWT "top-level" claims and an "events" claim that contains one or more event URI subclaims (JSON attributes) each with a JSON object containing relevant event information.

This specification defines a new URI prefix "urn:ietf:params:scim:event" which is used as the prefix for the following defined SCIM Events (see Section 7.3). Events are grouped into one of two sub-namespaces: "feed" (feed control notices) or "prov" (provisioning).

2.1. Identifying the Subject of an Event

SCIM Events MUST use the "sub_id" claim, defined by [RFC9493], to identify the subject of events. The "sub_id" claim MUST be contained within the main JWT claims body and MUST NOT be located within an event payload within the "events" claim. A SET with multiple event URIs indicates that the events arise from the same transaction or resource state change for a single resource or subject.

The JWT "sub" claim MUST NOT be used to identify subjects to prevent confusion with JWT authorization tokens (originally recommended in Section 3 of [RFC8417]).

```
{
     "iss": "issuer.example.com",
     "iat": 1508184845,
     "aud": "aud.example.com",
     "sub_id": {
        "format": "scim",
        "uri": "/Users/2b2f880af6674ac284bae9381673d462",
        "externalId": "alice@example.com"
     },
     "events": {
       ...
     }
}
```

Figure 1: SCIM Subject Id Example

Instead of "sub", the top-level claim "sub_id" SHALL be used. "sub_id" contains the subclaim attribute "format" set to "scim" to indicate the attributes present in the "sub_id" object are SCIM attributes. The following "sub_id" attributes are defined:

uri
: The SCIM relative path for the resource which usually consists of the resource type endpoint plus the resource "id" (see Section 3.2 of [RFC7644]). For example "/Users/2b2f880af6674ac284bae9381673d462". This attribute MUST be provided in a SCIM Event "sub_id" claim. Note the relative path is the path component after the SCIM Service Provider Base URI as defined in Section 1.3 of [RFC7644]. In cases where the Event Receiver is unable to match a URI, the Event Receiver MAY issue a callback to a previously agreed SCIM Service Provider Base URI plus the relative "uri" value and perform a SCIM GET request per Section 3.4.1 of [RFC7644].

externalId
: If known, the "externalId" value (defined in Section 3.1 of [RFC7643]) of the SCIM Resource that MAY be used by a receiver to identify the corresponding resource in the Event Receiver’s domain.

id
: The SCIM Id attribute (defined in Section 3.1 of [RFC7643]) MAY be used for backwards compatibility reasons in addition to the "uri" claim.

In cases where SCIM identifiers ("id" and "externalId") are not enough to identify a common resource between an Event Publisher and Event Receiver, the "sub_id" object MAY contain attributes whose SCIM attribute types have "uniqueness" set to "server" or "global" as per Section 7 of [RFC7643]. For example, attributes such as "emails" or "username" (defined in Section 4 of [RFC7643]) are unique with in a SCIM Service Provider. Such attributes should allow an Event Publisher and Event Receiver to identify a commonly understood subject resource of an event.

2.2. Common Event Attributes

The following attributes are available for all events defined. Some attributes are defined as SET/JWT claims, while others are "Event Payload" claims as defined in Section 1.2 of [RFC8417]. Only one of "data" or "attributes" claims MUST be provided depending on the event definition.

txn
: "txn" is a SET-defined claim with a STRING value (see Section 2.2 of [RFC8417]) that uniquely identifies a transaction originating at a SCIM Service Provider and/or its underlying data repository or database where one or more SCIM Events may be subsequently issued. In contrast to a "jti" claim (see Section 4.1.7 of

[RFC7519]), which uniquely identifies a token, the "txn" remains the same when one or more SETs are generated for various purposes such as re-transmission, publication to multiple receivers etc. A distinct state change or transaction within a SCIM Service Provider MAY result in multiple SETs issued each with distinct "jit" values an a common "txn" value. "txn" is REQUIRED to support asynchronous SCIM requests, co-ordinated provisioning, and replication to disambiguate or detect duplicate SETs regarding the same underlying transaction.

version
: The Etag version of the resource as a result of the event and corresponds to the Etag response header described in Section 3.14 of [RFC7644].

data
: This event payload attribute contains information described in the SCIM Bulk Operations "data" attribute in Section 3.7 of [RFC7644]. The JSON object contains the equivalent SCIM command processed by the SCIM Service Provider. For example, after processing a SCIM Create operation, the data contained includes the final representation of the created entity by the SCIM Service Provider including the assigned "id" value.

attributes
: This payload contains an array of attributes that were added, revised, or removed. Names of modified attributes SHOULD conform to the ABNF syntax rule for "path"> (Section 3.5.2 of [RFC7644]). For example:
"attributes": ["username","emails","name.familyName"]

2.3. SCIM Feed Events

This section defines events related to changes in the content of an event feed. Such as, SCIM Resources that are being added or removed from an event feed or events used in Co-operative Provisioning scenarios where only a sub-set of entities are shared across an Event Feed. The URI prefix for these events is "urn:ietf:params:scim:event:feed"

2.3.1. urn:ietf:params:scim:event:feed:add

The specified resource has been added to the Event Feed. A "feed:add" does not indicate a resource is new or has been recently created. For example, an existing user has had a new role (e.g., CRM_User) added to their profile which has caused their resource to join a feed.

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "txn": "b7b953f11cc6489bbfb87834747cc4c1",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/2b2f880af6674ac284bae9381673d462",
    "externalId": "jdoe"
  },
  "events":{
    "urn:ietf:params:scim:event:feed:add": {}
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 2: Example SCIM Feed Add Event

2.3.2. urn:ietf:params:scim:event:feed:remove

The specified resource has been removed from the feed. Removal does not indicate that the resource was deleted or otherwise deactivated.


```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/2b2f880af6674ac284bae9381673d462",
    "externalId": "jdoe",
  },
  "events":{
    "urn:ietf:params:scim:event:feed:remove": {}
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 3: Example SCIM Feed Remove Event

2.4. SCIM Provisioning Events

This section defines resource changes that have occurred within a SCIM Service Provider. These events are used in both Domain Based Replication (DBR) and Co-operative Provisioning (CP) mode. The URI prefix for these events is "urn:ietf:params:scim:event:prov".

For each of the following events when the "data" payload attribute is included, the event URI MUST end with "full", otherwise the event URI ends with "notice". In "full" mode, the set of values reflecting the final representation of the resource (such as would be returned in a SCIM protocol response) at the Service Provider are provided using the "data" attribute (see Figure 4). In "notice" mode, the "attributes" attribute is returned listing the set of attributes created or modified in the request (see Figure 5). Exactly one of the payload attributes "data" or "attributes", MUST be present. Both MUST NOT be present simultaneously.

2.4.1. urn:ietf:params:scim:event:prov:create:{notice|full}

Indicates a new SCIM resource has been created by the SCIM Service Provider and has been added to the Event Feed. Note that because the event may be used for replication, the "id" attribute that was assigned by the SCIM Service Provider is shared so that all replicas in the domain MAY use the same resource identifier.

```
{
  "jti": "4d3559ec67504aaba65d40b0363faad8",
  "iat": 1458496404,
  "iss":"https://scim.example.com",
  "aud":[
    "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754",
    "https://scim.example.com/Feeds/5d7604516b1d08641d7676ee7"
  ],
  "sub_id": {
    "format": "scim",
    "uri": "/Users/44f6142df96bd6ab61e7521d9",
     "externalId":"jdoe"
  },
  "events":{
    "urn:ietf:params:scim:event:prov:create:full":{
      "data":{
        "schemas":[ "urn:ietf:params:scim:schemas:core:2.0:User"],
        "emails":[
          {"type":"work","value":"jdoe@example.com"}
        ],
        "userName":"jdoe",
        "name":{
          "givenName":"John",
          "familyName":"Doe"
        }
      }
    }
  }
}
```


Figure 4: Example SCIM Create Event (Full)

```
{
  "jti": "4d3559ec67504aaba65d40b0363faad8",
  "iat": 1458496404,
  "iss":"https://scim.example.com",
  "aud":[
    "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754",
    "https://scim.example.com/Feeds/5d7604516b1d08641d7676ee7"
  ],
  "sub_id": {
    "format": "scim",
    "uri": "/Users/44f6142df96bd6ab61e7521d9",
    "externalId": "jdoe"
  },
  "events": {
    "urn:ietf:params:scim:event:prov:create:notice": {
      "attributes": [
        "id",
        "name",
        "userName",
        "password",
        "emails"
      ]
    }
  }
}
```

Figure 5: Example SCIM Create Event (Notice)

The event shown in Figure 5 notifies the Event Receiver which attributes have changed but does not convey the actual information. The Event Receiver MAY retrieve that information by performing a SCIM GET based on the "sub_id" value provided.

2.4.2. urn:ietf:params:scim:event:prov:patch:{notice|full}

The specified resource has been updated using SCIM PATCH. In "full" mode, the "data" payload attribute is included (see Figure 6). When the event URI ends with "notice", the list of modified attributes is provided (see Figure 7).

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Groups/176f397ec4c44b94b2cfcb759780b8c2",
    "externalId": "crmUsers"
  },
  "events":{
    "urn:ietf:params:scim:event:prov:patch:full": {
      "version": "a330bc54f0671c9",
      "data": {
        "schemas":
      ["urn:ietf:params:scim:api:messages:2.0:PatchOp"],
        "Operations":[{
          "op":"add",
          "path":"members",
          "value":[{
             "display": "Babs Jensen",
             "$ref": "/Users/2819c223...413861904646",
             "value": "2819c223-7f76-453a-919d-413861904646"
          }]
        }]
      }
    }
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 6: Example SCIM Patch Event (Full)

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Groups/176f397ec4c44b94b2cfcb759780b8c2",
    "externalId": "crmUsers"
  },
  "events":{
    "urn:ietf:params:scim:event:prov:patch:notice": {
      "attributes": ["members"],
      "version": "a330bc54f0671c9"
    }
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 7: Example SCIM Patch Event (Notice)

2.4.3. urn:ietf:params:scim:event:prov:put:{notice|full}

The specified resource has been updated (e.g., one or more attributes has changed). In "full" mode, the SCIM PUT request body is included in the "data" attribute (see Figure 8). In "notice" mode, the modified attributes are listed using "attributes" (see Figure 9).

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/2819c223-7f76-453a-919d-413861904646"
  },
  "events":{
    "urn:ietf:params:scim:event:prov:put:full": {
      "version": "a330bc54f0671c9",
      "data": {
        "schemas":["urn:ietf:params:scim:schemas:core:2.0:User"],
        "userName":"jdoe",
        "externalId":"jdoe",
        "name":{
           "formatted":"Mr. Jon Jack Doe III",
           "familyName":"Doe",
           "givenName":"Jon",
           "middleName":"Jack"
        },
        "roles":[],
        "emails":[
          {"value":"jdoe@example.com"},
          {"value":"anon@jdoe.org"}
        ]
      }
    }
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 8: Example SCIM Put Event (Full)

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/2819c223-7f76-453a-919d-413861904646"
  },
  "events":{
    "urn:ietf:params:scim:event:prov:put:notice": {
      "version": "a330bc54f0671c9",
      "attributes": ["userName","externalId","name","roles","emails"]
    }
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```

Figure 9: Example SCIM Put Event (Notice)

2.4.4. urn:ietf:params:scim:event:prov:delete

The specified resource has been deleted from the SCIM Service Provider. The resource is also removed from the feed. When a DELETE is sent, a corresponding "feedRemove" SHALL NOT be issued. A delete event has no payload attributes. Note that because the delete event has no attributes, the qualifiers "full" and "notice" SHALL NOT be used.

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/2b2f880af6674ac284bae9381673d462",
    "externalId": "jDoe"
  },
  "events":{
    "urn:ietf:params:scim:event:prov:delete": {}
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```

Figure 10: Example SCIM Delete Event

2.4.5. urn:ietf:params:scim:event:prov:activate

The specified resource (e.g., User) has been "activated". This does not necessarily reflect any particular state change at the SCIM Service Provider but may simply indicate the account defined by the SCIM resource is ready for use as agreed upon by the Event Publisher and Event Receiver. For example, an activated resource can represent an account that may be logged in.

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/2b2f880af6674ac284bae9381673d462"
  },
  "events":{
    "urn:ietf:params:scim:event:prov:activate": {}
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```

Figure 11: Example SCIM Activate Event

2.4.6. urn:ietf:params:scim:event:prov:deactivate

The specified resource (e.g., User) has been deactivated and disabled. The exact meaning SHOULD be agreed to by the Event Publisher and its corresponding Event Receiver. Typically, this means the subject may no longer have an active security session.

2.5. Miscellaneous Events

This section defines related miscellaneous events such as Asynchronous Request completion that has occurred within a SCIM Service Provider. The URI prefix for these events is "urn:ietf:params:scim:event:misc".

2.5.1. Asynchronous Events

2.5.1.1. Making an Asynchronous SCIM Request

A SCIM Client making SCIM HTTP requests defined in Section 3 of [RFC7644] MAY request asynchronous processing using the "Prefer" HTTP Header as defined in Section 4.1 of [RFC7240]. The client may do this for a number of reasons such as avoiding holding HTTP connections open during long requests, because the result of the request is not needed, or for co-ordination reasons where the result is delivered to another entity for further action.

To initiate an asynchronous SCIM request, a normal SCIM protocol POST, PUT, PATCH, or DELETE request is performed with the HTTP "Prefer" Header set to "respond-async" (Section 4.1 of [RFC7240]). The HTTP "Accept" header MUST be ignored for purposes of an asynchronous response. Additionally, per Section 4.3 of [RFC7240], the "wait" preference SHOULD be supported to establish a maximum time before a SCIM Service Provider MAY choose to respond asynchronously.

In response, the SCIM Service Provider either returns a normal SCIM response or returns HTTP Status 202 (Accepted). The asynchronous response MUST contain no response body. To enable correlation of the future event, the HTTP response header "set-txn" (see Section 3) is returned with a value that MUST match the "txn" claim in a subsequent Security Event Token. Per [RFC7240], Section 3, the response will also include the "Preference-Applied" header. The "Location" header value MUST be one of the following: (a) a URI where the completion SCIM Event Token MAY be retrieved using HTTP GET, or (b) the normal SCIM location header response specified by [RFC7644].

In the following non-normative example, a "Prefer" header is set to "respond-async":

```
PUT /Users/2819c223-7f76-453a-919d-413861904646
Host: scim.example.com
Prefer: respond-async
Content-Type: application/scim+json
Authorization: Bearer h480djs93hd8

{
  "schemas":["urn:ietf:params:scim:schemas:core:2.0:User"],
  "id":"2819c223-7f76-453a-919d-413861904646",
  "userName":"bjensen",
  "externalId":"bjensen",
  "name":{
    "formatted":"Ms. Barbara J Jensen III"
  },
  "roles":[],
  "emails":[
    {
        "value":"bjensen@example.com"
    }
  ]
}
```

Figure 12: Example Asynchronous SCIM Protocol Request

The SCIM Service Provider responds with HTTP 202 Accepted and includes the set-txn header:

```
HTTP/1.1 202 Accepted
set-txn: 734f0614e3274f288f93ac74119dcf78
Preference-Applied: respond-async
Location:
  "/Users/2819c223-7f76-453a-919d-413861904646"
```

Figure 13

2.5.1.2. Asynchronous Bulk Endpoint Requests

Section 3.7 of [RFC7644] provides the ability to submit multiple SCIM operations in a single "bulk" request. When an asynchronous response is requested, a single Asynchronous Request Completion Event MUST be generated for each requested operation. For example, if a single "bulk" request had 10 operations, then 10 Asynchronous Event completions events would be generated.

The "txn" claim MUST be set to the value originally returned to the requesting SCIM Client (see Section 2.5.1.1) appended with a colon ":" and the zero-based array index of the operation expressed in the "Operations" attribute of the original bulk request. The "bulkId" parameter MUST NOT be used for this purpose as it is a temporary identifier and is not required for every operation.

For example, if a SCIM Service Provider received a Bulk request with two or more operations, and had a "txn" claim value of "2d80e537a3f64622b0347b641ebc8f44", then the first Asynchronous Response Event Token representing the first operation has a "txn" claim value of "2d80e537a3f64622b0347b641ebc8f44:0", the second operation has a value of "2d80e537a3f64622b0347b641ebc8f44:1", and so on.

If a SCIM Service Provider optimizes the sequence of operations (per Section 3.7 of [RFC7644]), the Asynchronous Request Completion events generated MAY be generated out of sequence from the original request. In this case, the "txn" claims in those events MUST use operation numbers that correspond to the order in the original request.

2.5.1.3. urn:ietf:params:scim:event:misc:asyncresp

The Asynchronous Response event signals the completion of a SCIM request. The event payload contains the attributes defined in Section 3.7 of [RFC7644] and is the same as a single SCIM Bulk Response Operation as per Section 3.7.3. In the event, the "txn" claim MUST be set to the value originally returned to the requesting SCIM Client (see Section 2.5.1.1).

```
{
  "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/2819c223-7f76-453a-919d-413861904646"
  },
  "txn": "734f0614e3274f288f93ac74119dcf78",
  "events":{
    "urn:ietf:params:scim:event:misc:asyncresp": {
        "method": "PUT",
        "version": "W\/\"huJj29dMNgu3WXPD\"",
        "status": "200"
    }
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 14: Example SCIM Asynchronous Response Event

If an error occurs during asynchronous processing, the event operation MUST include a "response" attribute indicating a non-200-series HTTP status as defined in Section 3.7 of [RFC7644], and that "response" attribute MUST contain the sub-attributes defined in Section 3.12 of [RFC7644]. The "status" attribute of the event operation typically matches the "status" attribute of the response.

```
{
 "jti": "6164f3bbf6ff41a88dc94f18cb0620e8",
 "sub_id": {
   "format": "scim",
   "uri": "/Users/2819c223-7f76-453a-919d-413861904646"
 },
 "txn": "734f0614e3274f288f93ac74119dcf78",
 "events":{
   "urn:ietf:params:scim:event:misc:asyncresp": {
       "method": "PUT",
       "version": "W\/\"huJj29dMNgu3WXPD\"",
       "status": "400",
       "response": {
           "schemas": [
               "urn:ietf:params:scim:api:messages:2.0:Error"
           ],
           "scimType":"invalidSyntax",
           "detail": "Request is unparsable",
           "status":"400"
       }
   }
 },
 "iat": 1458505044,
 "iss":"https://scim.example.com",
 "aud":[
  "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
 ]
}
```


Figure 15: Example SCIM Asynchronous Error Response Event

The following 4 figures show Asynchronous Completion events for the example in Section 3.7.3 of [RFC7644].

```
{
  "jti": "dbae9d7506b34329aa7f2f0d3827848b",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/92b725cd-9465-4e7d-8c16-01f8e146b87a"
  },
  "txn": "2d80e537a3f64622b0347b641ebc8f44:1",
  "events":{
    "urn:ietf:params:scim:event:misc:asyncresp": {
         "method": "POST",
         "bulkId": "qwerty",
         "version": "W\/\"oY4m4wn58tkVjJxK\"",
         "status": "201"
    }
  },
  "iat": 1458505044,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 16: Example SCIM Asynchronous Response Event Operation 1/4


```
{
  "jti": "ca977d05ba5c43929e3a69023d5392a9",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/b7c14771-226c-4d05-8860-134711653041"
  },
  "txn": "2d80e537a3f64622b0347b641ebc8f44:2",
  "events":{
    "urn:ietf:params:scim:event:misc:asyncresp": {
         "method": "PUT",
         "version": "W\/\"huJj29dMNgu3WXPD\"",
         "status": "200"
    }
  },
  "iat": 1458505045,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 17: Example SCIM Asynchronous Response Event Operation 2/4

```
{
  "jti": "4bb87d70a4ab463bbdcd1f99111cbbf1",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/5d8d29d3-342c-4b5f-8683-a3cb6763ffcc"
  },
  "txn": "2d80e537a3f64622b0347b641ebc8f44:3",
  "events":{
    "urn:ietf:params:scim:event:misc:asyncresp": {
          "method": "PATCH",
          "version": "W\/\"huJj29dMNgu3WXPD\"",
          "status": "200"
    }
  },
  "iat": 1458505046,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 18: Example SCIM Asynchronous Response Event Operation 3/4


```
{
  "jti": "6a7843a7f5244d0eb62ca38b641d9139",
  "sub_id": {
    "format": "scim",
    "uri": "/Users/e9025315-6bea-44e1-899c-1e07454e468b"
  },
  "txn": "2d80e537a3f64622b0347b641ebc8f44:4",
  "events":{
    "urn:ietf:params:scim:event:misc:asyncresp": {
          "method": "DELETE",
          "status": "204"
    }
  },
  "iat": 1458505047,
  "iss":"https://scim.example.com",
  "aud":[
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754"
  ]
}
```


Figure 19: Example SCIM Asynchronous Response Event Operation 4/4

3. set-txn HTTP Response Header for Asynchronous Requests

This specification defines a new HTTP Header field "set-txn" which serves the purpose of conveying request completion information to SCIM HTTP clients that request an asynchronous response as described in Section 2.5.1.1. The header field MUST be used in SCIM Responses when HTTP Status 202 Accepted is being returned with no message body.

The "set-txn" HTTP Header field value is a unique STRING (e.g., a GUID) used by the SCIM HTTP client to look for a matching SET event with a matching "txn" claim (see Section 2 of [RFC8417]) confirming the request completion status as described in Section 2.5.1.1.

Intermediaries SHOULD NOT insert, modify, or delete the field’s value.

SCIM clients MAY ignore the header in cases where confirmation of completion is not required. For example a SCIM client may simply not want to wait for synchronous completion.

4. Events Discovery Schema for Service Provider Configuration

Section 5 of [RFC7643] defines SCIM Service Provider configuration schemas. This section defines additional attributes that enable a SCIM Client to discover the additional capabilities defined by this specification.

securityEvents
: A SCIM Complex attribute that specifies the available capabilities related to asynchronous Security Events based on [RFC8417]. This attribute is OPTIONAL and when absent indicates the SCIM Service Provider does not support or is not currently configured for Security Events. The following sub-attributes are defined:

  asyncRequest
  : A case-insensitive string value specifying one of the following:

    * "none" indicates asynchronous SCIM requests defined in Section 2.5.1.1 are not supported;

    * "long" indicates the server completes requests asynchronously at server discretion (e.g. based on a max wait time);

    * "request" indicates the server completes requests asynchronously when requested by the SCIM Client.

eventUris
: A multivalued string listing the SET Event URIs (defined in [RFC8417]) that the server is capable of generating and deliverable via a SET Stream (see [RFC8935] and [RFC8936]). This information is informational only. Stream registration and configuration are out of scope of this specification.

5. Security Considerations

As this specification is based upon the Security Event Tokens specification and the associated delivery specifications the following Security Considerations are also applicable to this specification:

* Section 5 of [RFC8417] (Security Event Token)

* Section 5 of [RFC8935] (Push-based Delivery Using HTTP)

* Section 4 of [RFC8936] (Poll-Based Delivery Using HTTP)

SETs may contain sensitive information, including Personally Identifiable Information (PII). In such cases, SET Transmitters and SET Recipients MUST protect the confidentiality of the SET contents in transit using TLS [BCP195].

When co-ordinating provisioning between entities, the long-term series of changes may be critical to the information integrity and recovery requirements of both sides. To address this, Event Publishers can make events available for receivers for longer periods of time than might typically be used for recovering from momentary delivery failures and retries per [RFC8935] or [RFC8936]. Similarly, Event Receivers MUST ensure events are persisted directly or indirectly to meet local recovery needs before acknowledging the SET Events were received.

An attacker might leverage transaction and/or signal information contained in SET Event Publisher or Receiver system. To mitigate this, access to event recovery and forwarding MUST be limited to the parties needed to support recovery or SET forwarding.

When SET Events are transferred in such a way as the Event Publisher is not communicating directly to the Event Receiver, it may become possible for an attacker or other system to insert an event. To mitigate, Event Receivers MUST verify the originator of a SET using JWS [RFC7515] signatures when the Event Publisher is not communicating directly with the Event Receiver. Validating event signatures may also be useful for auditing purposes as signed SET Events are protected from tampering in the event that an intermediate system, such as a TLS-terminating proxy, decrypts the SET payload before sending it onward to its intended recipient.

In operation, some SCIM Resources such as SCIM Groups may have a high rate of change. For examples groups with more than a few thousand member values could lead to excessive change rates that could lead to a loss of SET Events between Event Publishers and Event Receivers. To mitigate this risk, consider the following to help mitigate throughput issues:

* The use of SCIM PUT (Section 3.5.1 of [RFC7644]), particularly with large SCIM Groups, can result in excessive data being conveyed in Security Event payloads. Instead, it is RECOMMENDED to use SCIM PATCH (Section 3.5.2 of [RFC7644]) to focus on updating and notifying about changed information. Alternatively, use SCIM PUT Event Notice (urn:ietf:params:scim:event:prov:put:notice) as a trigger to later retrieve the full information when needed.

* Use SCIM Patch Event Notice (urn:ietf:params:scim:event:prov:patch:notice) to reduce event content combined with periodic SCIM GETs (see section 3.4 of [RFC7644]) to retrieve current group state.

* Aggregate multiple PATCH Events into a single event. Providing the exact date of each membership change is not critical but instead that the information content remains intact.

When using Asynchronous SCIM Requests (see Section 2.5.1.1), a SCIM Service provider returns a SCIM Accepted response with a URI for retrieving the event result. An unauthorized entity or attacker could obtain asynchronous request completion event information by querying the asynchronous operation result endpoint used by a SCIM Service Provider. To mitigate, the returned URI endpoint MUST be protected requiring an HTTP Authorization header or some other form of client authentication.

6. Privacy Considerations

As this specification is based upon the Security Event Tokens and the associated delivery specifications the following Privacy Considerations are also applicable to this specification:

* Section 6 of [RFC8417] (Security Event Token)

* Section 6 of [RFC8935] (Push-based Delivery Using HTTP)

* Section 5 of [RFC8936] (Poll-Based Delivery Using HTTP)

This specification enables the sharing of information between domains. The specification assumes that implementers and deployers are operating under one of the following scenarios:

* A common administrative domain where there is one administrative owner of the data. In these cases, the goal is to protect privacy and security of the owner and user data by keeping information systems co-ordinated and up-to-date. For example, the domains decide to use Domain Based Replication mode to keep employee information synchronized.

* In a co-operative or co-ordinated relationship, parties have decided to share a limited amount of data and/or signals for the benefits of their users. Depending on end-user consent, information is shared on an as-authorized and/or as-needed basis. For example, the domains agree to use Co-ordinated Provision mode that exchanges things like account status or specific minimal attribute information that must be fetched on request after receiving notice of a change. This enables authorization to be verified each time data is transferred.

In general, the sharing of SCIM Event information falls within a pre-existing SCIM Client and Service Provider relationship and carry no additional personal information.

7. IANA Considerations

7.1. SCIM Asynchronous Txn Header Registration

This specification registers the HTTP "set-txn" field name in the "HTTP Field Name Registry" defined in Section 16.3.1 of [RFC9110].

Field name:
set-txn

Status:
Permanent

Specification Document:
See Section Section 3 of this document.

7.2. Registering Event Capability with Scim Service Provider Config

For the SCIM Schema Registry Section 10.4 of [RFC7643], under Service Provider Configuration Schema ("urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig"), add Section 4 of this document to the Reference column.

7.3. Registration of the SCIM Event URIs Sub-Registry

IANA will add a new registry called “SCIM Event URIs” to the “System for Cross-domain Identity Management (SCIM) Schema URIs” registry group define by Section 10.1 of [RFC7643] at https://www.iana.org/assignments/scim.

Procedures and definitions for this sub-registry build upon Section 10.3.1 of [RFC7643]. New registrations for this sub-registry are evaluated on a first-come, first-served basis for relevance to SCIM-based systems, and, to avoid possible duplication or conflict with other event definitions that may lie outside SCIM (e.g., Shared Signals [SSF]).

Namespace ID:
The sub-namespace ID of "event" is assigned within the "scim" namespace.

Syntactic Structure:
The Namespace Specific String (NSS) of all URNs that use the "event" Namespace ID has the following structure:

"urn:ietf:params:scim:event:{class}:{name}:{other}

The keywords have the following meaning:

class
The class of events which is one of: "feed", "prov" or "misc".

name
A US-ASCII string that conforms to URN syntax requirements (see [RFC8141]) and defines a descriptive event name (e.g., "create").

other
An optional US-ASCII string that conforms to URN syntax requirements (see [RFC8141]) and serves as an additional sub-category or qualifier. For example "full" and "notice".

Identifier Uniqueness Considerations:
The designated contact is responsible for reviewing and enforcing uniqueness.

Identifier Persistence Considerations:
Once a name has been allocated it MUST NOT be re-allocated for a different purpose. The rules provided for assignments of values within a sub-namespace MUST be constructed so that the meaning of values cannot change. This registration mechanism is not appropriate for naming values whose meaning may change over time.

Registration format:
An event registration MUST include the following fields:

* Event Uri

* Descriptive Name

* Reference to event definition

Initial values to be added to the SCIM Events Registry are listed in Section 7.4.

7.4. Initial Events Registry

Summary of Event URI registrations:

Event URI	Name	Ref.
urn:ietf:params:scim:event:feed:add	Resource added to Feed Event	Section 2.3.1 of this document.
urn:ietf:params:scim:event:feed:remove	Remove resource From Feed Event	Section 2.3.2 of this document.
urn:ietf:params:scim:event:prov:create:notice	New Resource Event (notice	Section 2.4.1 of this

	only)	document.
urn:ietf:params:scim:event:prov:create:full	New Resource Event (full data)	Section 2.4.1 of this document.
urn:ietf:params:scim:event:prov:patch:notice	Resource Patch Event (notice only)	Section 2.4.2 of this document.
urn:ietf:params:scim:event:prov:patch:full	Resource Patch Event (full data)	Section 2.4.2 of this document.
urn:ietf:params:scim:event:prov:put:notice	Resource Put Event (notice only)	Section 2.4.3 of this document.
urn:ietf:params:scim:event:prov:put:full	Resource Put Event (full data)	Section 2.4.3 of this document.
urn:ietf:params:scim:event:prov:delete	Resource Deleted Event	Section 2.4.4 of this document.
urn:ietf:params:scim:event:prov:activate	Resource Activated Event	Section 2.4.5 of this document.
urn:ietf:params:scim:event:prov:deactivate	Resource Deactivated Event	Section 2.4.6 of this document.
urn:ietf:params:scim:event:misc:asyncresp	Asynchronous Request Completion	Section 2.5.1 of this document.

Table 1

8. References

8.1. Normative References

[BCP195] Best Current Practice 195, <https://www.rfc-editor.org/info/bcp195>. At the time of writing, this BCP comprises the following:

Moriarty, K. and S. Farrell, "Deprecating TLS 1.0 and TLS 1.1", BCP 195, RFC 8996, DOI 10.17487/RFC8996, March 2021, <https://www.rfc-editor.org/info/rfc8996>.

Sheffer, Y., Saint-Andre, P., and T. Fossati, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 9325, DOI 10.17487/RFC9325, November 2022, <https://www.rfc-editor.org/info/rfc9325>.

[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>.

[RFC7240] Snell, J., "Prefer Header for HTTP", RFC 7240, DOI 10.17487/RFC7240, June 2014, <https://www.rfc-editor.org/info/rfc7240>.

[RFC7515] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Signature (JWS)", RFC 7515, DOI 10.17487/RFC7515, May 2015, <https://www.rfc-editor.org/info/rfc7515>.

[RFC7519] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Token (JWT)", RFC 7519, DOI 10.17487/RFC7519, May 2015, <https://www.rfc-editor.org/info/rfc7519>.

[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>.

[RFC8141] Saint-Andre, P. and J. Klensin, "Uniform Resource Names (URNs)", RFC 8141, DOI 10.17487/RFC8141, April 2017, <https://www.rfc-editor.org/info/rfc8141>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8417] Hunt, P., Ed., Jones, M., Denniss, W., and M. Ansari, "Security Event Token (SET)", RFC 8417, DOI 10.17487/RFC8417, July 2018, <https://www.rfc-editor.org/info/rfc8417>.

[RFC8935] Backman, A., Ed., Jones, M., Ed., Scurtescu, M., Ansari, M., and A. Nadalin, "Push-Based Security Event Token (SET) Delivery Using HTTP", RFC 8935, DOI 10.17487/RFC8935, November 2020, <https://www.rfc-editor.org/info/rfc8935>.

[RFC8936] Backman, A., Ed., Jones, M., Ed., Scurtescu, M., Ansari, M., and A. Nadalin, "Poll-Based Security Event Token (SET) Delivery Using HTTP", RFC 8936, DOI 10.17487/RFC8936, November 2020, <https://www.rfc-editor.org/info/rfc8936>.

[RFC9110] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/info/rfc9110>.

[RFC9493] Backman, A., Ed., Scurtescu, M., and P. Jain, "Subject Identifiers for Security Event Tokens", RFC 9493, DOI 10.17487/RFC9493, December 2023, <https://www.rfc-editor.org/info/rfc9493>.

8.2. Informative References

[I-D.hunt-idevent-scim] Hunt, P., Denniss, W., and M. Ansari, "SCIM Event Extension", Work in Progress, Internet-Draft, draft-hunt-idevent-scim-00, 20 March 2016, <https://datatracker.ietf.org/doc/html/draft-hunt-idevent-scim-00>.

[SSF] OpenID Foundation, "Shared Signals Framework".

Appendix A. Use Cases

SCIM Events may be used in a number of ways. The following non-normative sections describe some of the expected uses.

A.1. Domain Based Replication

The objective of "Domain Based Replication" events (DBR) is to synchronize resource changes between SCIM Service Providers in a common administrative domain. In this mode, complete information about modified resources are shared between replicas for immediate processing.

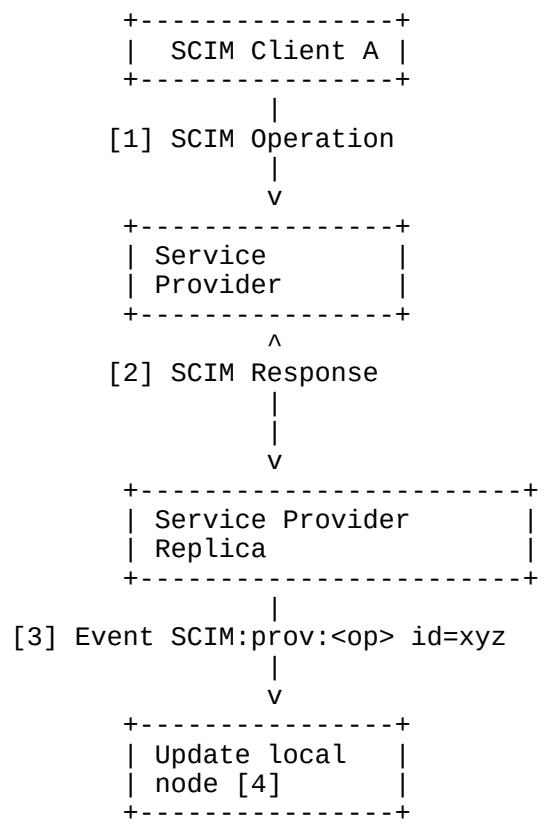

Figure 20: Domain Based Replication Sequence

From a security perspective, it is assumed that servers sharing DBR events are secured by a common access policy and all servers are required to be up-to-date. From a privacy perspective, because all servers are in the same administrative domain, the primary objective is to keep individual Service Provider nodes or cluster synchronized.

A.2. Co-ordinated Provisioning

In "Co-ordinated Provisioning" (CP), SCIM resource change events perform the function of change notification without the need to provide raw data. In any Event Publisher and Receiver relationship, the set of SCIM Resources (e.g., Users) that are linked or co-ordinated is managed within the context of an event feed and may be a subset of the total set of resources on either side. For example, an event feed could be limited to users who have consented to the sharing of information between domains. To support capability, "feed" specific events are defined to indicate the addition and removal of SCIM Resources from a feed. For example, when a user consents to the sharing of information between domains, events about the User may be added to the feed between the Event Publisher and Receiver.

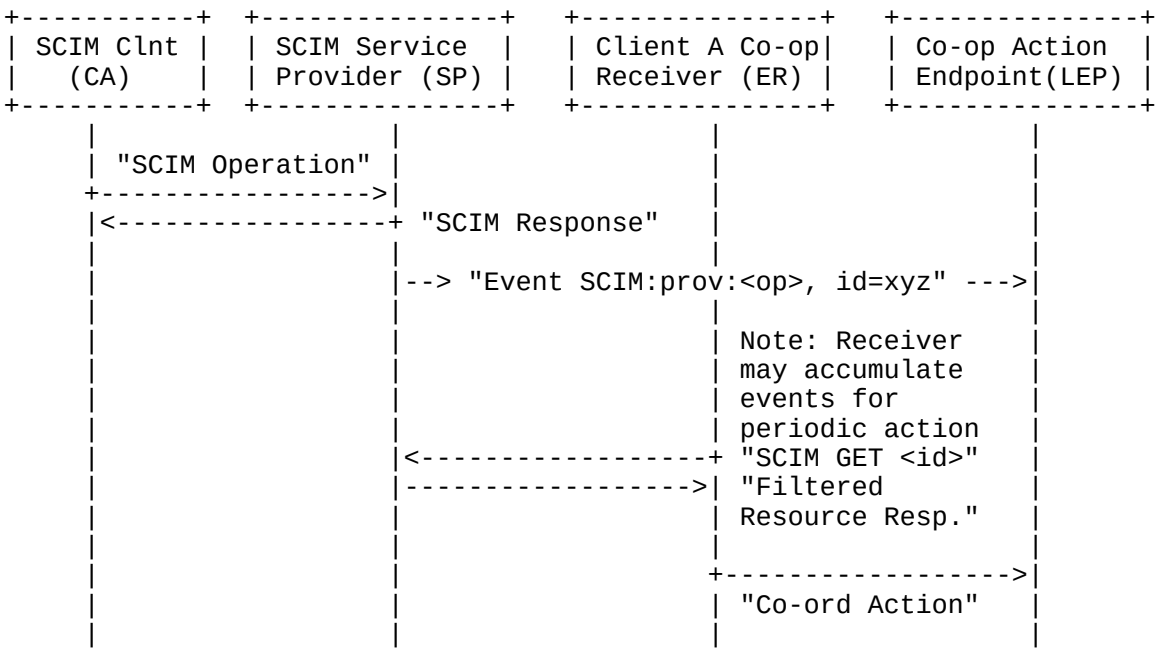

Figure 21: Co-Ordinated Provisioning Sequence

In CP mode, the receiver of an event must call back to the originating SCIM Service Provider (e.g., using a SCIM GET request) to reconcile the newly changed resource in order to obtain the changes.

Co-ordinated provisioning has the following benefits:

* Differences in schema (e.g., attributes) between domains. For example, a receiving domain may only be interested in or allowed to access to a few attributes (e.g., role-based access data) to enable access to an application.

* Different Event Receivers may have differing needs when accessing information and thus be assigned varying access rights. Minimal information events combined with callbacks for data allows data filtering to be applied.

* Receivers can take independent action. Such as deciding which attributes or resource lifecycle changes to accept. For example, in the case of a conflict, a receiver can prioritize one domain source over another.

* A receiver may throttle or buffer changes rather than act immediately on a notification. For example, for a frequently changing resource, the receiver may choose to make a scheduled SCIM GET for resources that have been marked "dirty" by events received in the last scheduled cycle.

A disadvantage of the CP approach is that it may be considered costly in the sense that each event received might trigger a callback to the event issuer. This cost should be weighed against the cost producing filtered information in each event for each receiver. Furthermore, a receiver is not required to make a callback on every provisioning event.

It is assumed that an underlying relationship between domains exists that permits the exchange of personal information and credentials. For example, in a cross-domain scenario a SCIM Service Provider would have been previously authorized to perform SCIM provisioning operations and publish change events. As such, appropriate confidentiality and privacy agreements should be in place between the domains.

When sharing information between parties, CP Events minimize the information shared in each message and require the Security Event Receiver to receive more information from the Event Publisher as needed. In this way, the Event Receiver is able to have regular access to information through normal SCIM protocol access restrictions. The Event Receiver and Publisher may agree to communicate these updates through a variety of transmission methods such as push and pull based HTTP like in [RFC8935], [RFC8936], or HTTP GET (see Section 2.5.1.1), streaming technologies (e.g., Kafka or Kinesis), or via webhooks as in the Shared Signals Framework [SSF].

Acknowledgements

The authors would like to thank the following contributors:

* Morteza Ansari and William Denniss, who contributed significantly to [I-D.hunt-idevent-scim], upon which this draft is based.

* The participants of the SCIM working group and the id-event list for their support of this specification.

* Thanks to Deb Cooley, Dean Saxe, Elliot Lear, Pamela Dingle, Mark Nottingham, R Gideon, Paulo Jorge Correia, Shuping Peng, Elwyn Davies, Luigi Lannone, Mohamed Boucadair, Roman Danyliw, Ketan Talaulikar, Mahesh Jethanandani, and Mike Bishop for their write-ups and reviews

Change Log

This section is to be removed before publishing as an RFC.

Draft 00 - PH - First WG Draft

Draft 01 - PH - Moved non-normative sections to Appendix, Security, and Privacy Considerations

Draft 02 - PH - Clarifications on Async Events, IANA Considerations

Draft 03 - PH - Fixed Header Field registration to RFC9110."Preference-Applied" header in async response. Support for Async Bulk requests. Added IANA SCIM Event Registry

Draft 04 - PH - Removed Event Delivery Feeds and Appendix A(not normative), Removed "sig" events, change bulk txn separator to ":", Updated SubId Reference to RFC9493, other comments, fixed IANA registry paragraph, SCIM Signals Removed

Draft 05 - PH - Removed Signals Events, Removed Delivery Section (not normative), Version(etag) definition added, Security Considerations revisions, Syntax for Attributes

Draft 06 - PH - Editorial edits and clarifications, add SSF reference

Draft 07 - PH - Document date update only

Draft 08 - PH - Update to Security Considerations to frame as risk/correction

Draft 09 - PH - Incorporating feedback from AD

Draft 10 - PH - IANA and ARTART Feedback

Draft 11 - PH - GenArt, OpsDir Feedback including new section on set-txn header, removed unicode art characters.

Draft 12 - PH - Update reference to Shared Signals to stable, IESG feedback

Draft 13 - PH - Tweaked usage of normative language

Draft 14 - PH - Modified IANA procedures for event registry

Draft 15 - PH - Replace tt elements with plain quotes

Authors' Addresses

Phil Hunt (editor)
Independent Identity Inc
Email: phil.hunt@independentid.com

Nancy Cam-Winget
Cisco Systems
Email: ncamwing@cisco.com

Mike Kiser
Sailpoint Technologies
Email: mike.kiser@sailpoint.com

Jen Schreiber
Workday, Inc.
Email: jennifer.winer@workday.com