IETF 113
SCIM Profile for Security Events

March 23, 2022

Phil Hunt, Independent Identity Inc
Nancy Cam-Winget, Cisco Systems

Agenda

- Security Event Tokens?
  - Origin and usage
- Shared Signals
- Use Cases
  - Domain Replication
  - Cross-domain Co-ordination
  - Signals
  - Misc
- Delivery Streams
  - Bus vs. Pont-to-Point SET
- Events
- Discussion
  - Scope of spec
  - Next steps

Origin of Security Event Token

- Originated within the SCIM WG
  - A mechanism to send back-channel messages "triggers"
- In 2015, several groups (SCIM, OAuth, OpenID) considering JWT.
  - The SCIM WG proposed a common standard form which became SET under the newly formed SECEVENTs group.
  - Unfortunately, the SCIM WG was "paused" while this happened as major deliverable deemed complete*
- Profiles the use of JWT for passing Security Events
  - Signable, securable, transportable in many ways
- What is an "Event"?
  - A statement about something that occurred about a subject
  - Interpreted by the receiver for independent action*
- This specification profiles SET for SCIM scenarios

Specifications

- RFC8417 – Security Event Token
- Delivery
  - Support for transfer and acknowledgement
  - Defines publishers and receivers
  - Limited SET recovery due to perceived stream scale (OIDC)
    - Publisher not obliged to retain after ack
    - Receiver implements own recovery once transferred*
  - Delivery Methods
    - RFC8935 – HTTP Push Delivery
    - RFC8936 – HTTP Polling Delivery
      - Includes support for "long-polling" to enable real-time
      - Costly when a publisher has 1k+ streams

Related: Shared Signals Events

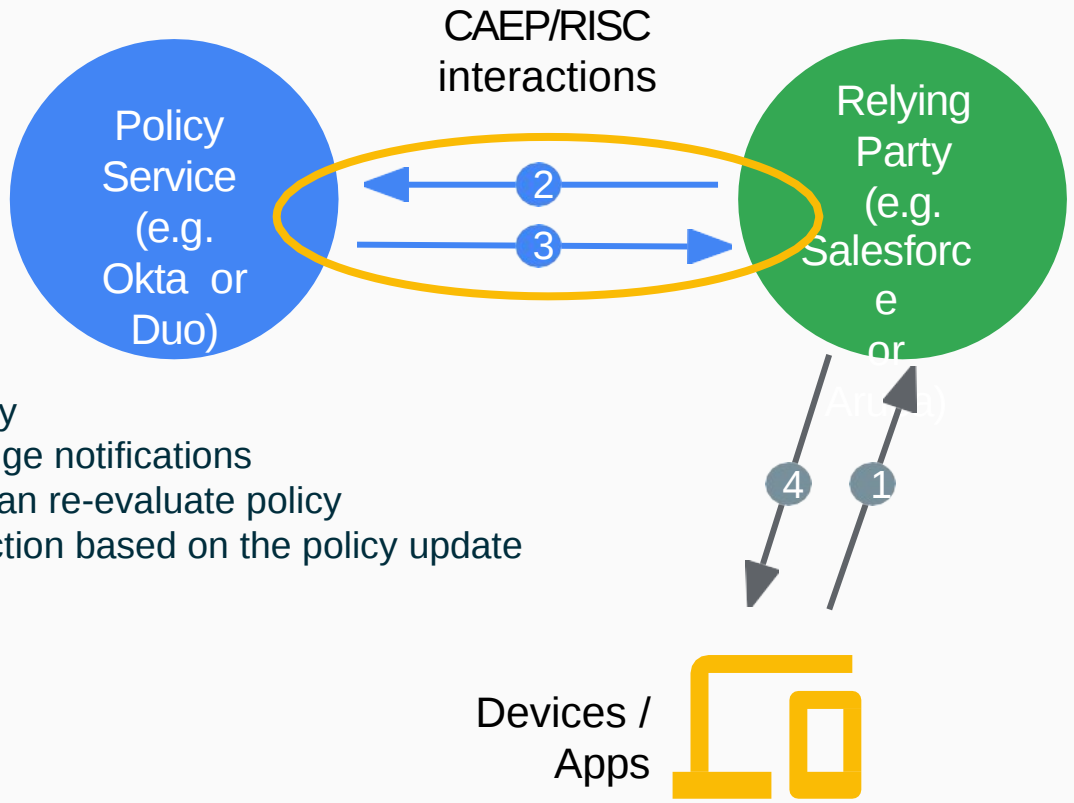
SSE is a framework that defines an API to enable:

- https://openid.net/wg/sse/
- RFC8935/8936 delivery streams plus
- Secure management of the streams (feeds)
- Streams carry events using RFC 8417 (Security Event Token)
- Events use Identifiers specified in draft-ietf-secevent-subject-identifiers
- 2 events schemas defined: CAEP and RISC

RISC Events
Account Credential Change Required
Credential Compromised
Account Purged/Disabled/Enabled
Identifier Changed/Recycled
Recovery Activated/Information Changed

CAEP Events
Credential Change
Session Revoked
Token Claims Change
Device Compliance Change
Assurance Level Change

Example Flow



1. Service Request: request service from a relying party
2. Context Update: Relying party can provide any change notifications
3. Policy Update: a subscriber to CAEP/RISC events can re-evaluate policy
4. Remediative Action: Relying party can enforce an action based on the policy update

Reference implementation: https://sharedsignals.guide/

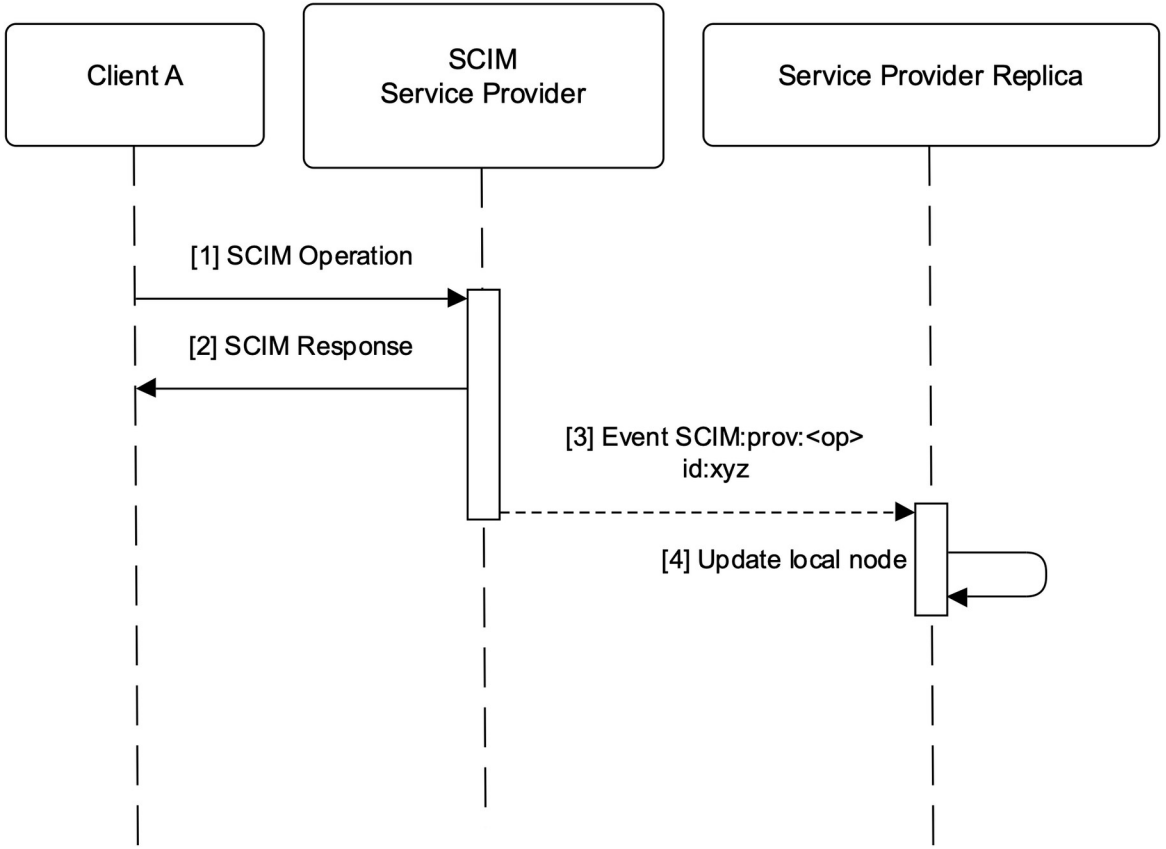
Major SCIM Use Case

Domain Based Replication

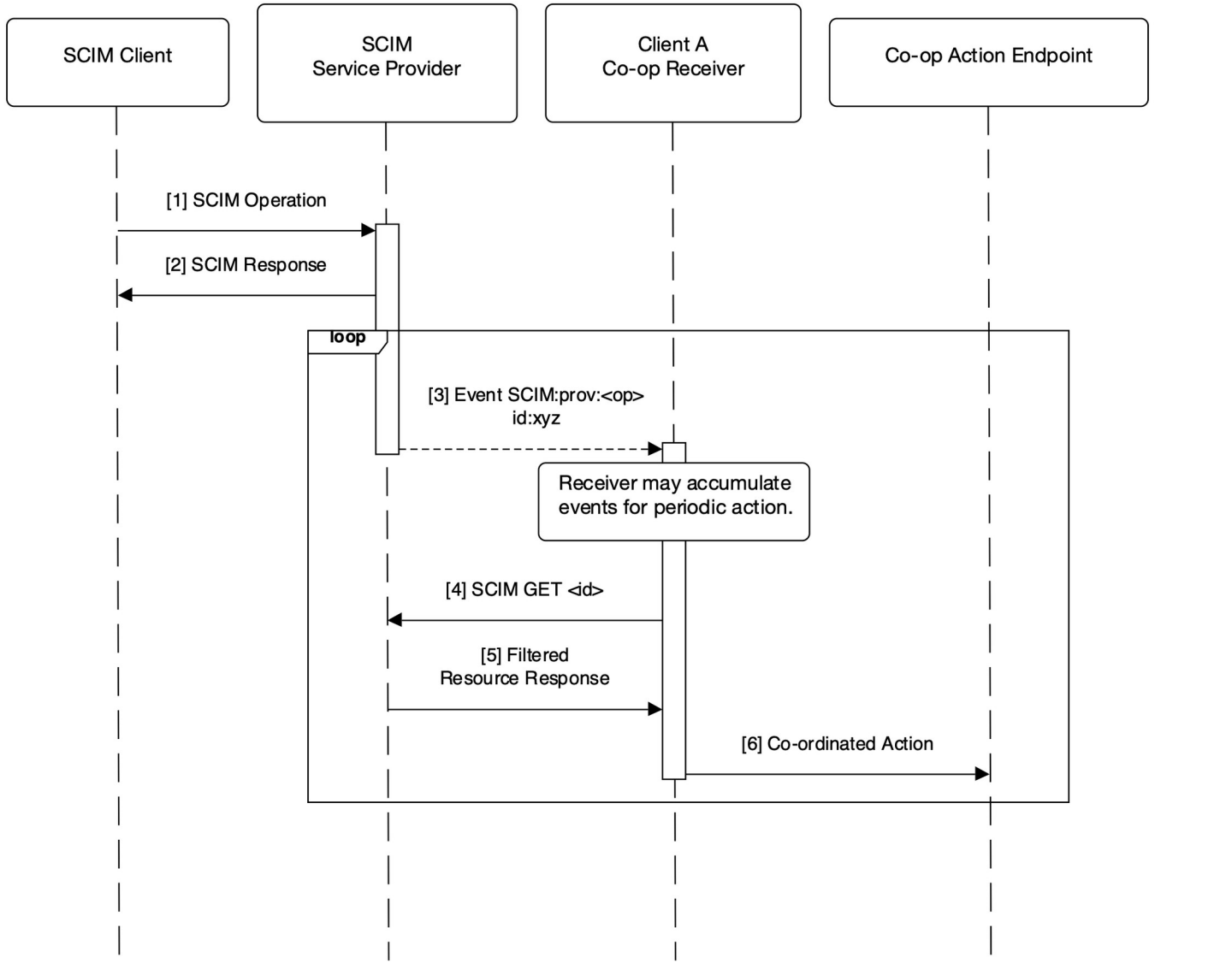
- Common Schema and Resources
- Single administrative domain
- Many nodes to synchronize
- May be multi SCIM implementations
- Acts as a common User repository or directory
- Messages convey transactions

Co-ordinated Provisioning

- Differing Schema and Resources
- May be multi-admin domain
- Point-to-point cross-domain link
- Often has differing implementations
- May be related to cross-domain workflows and entitlements
- Messages convey "triggers"



Sequence for Domain Based Replication



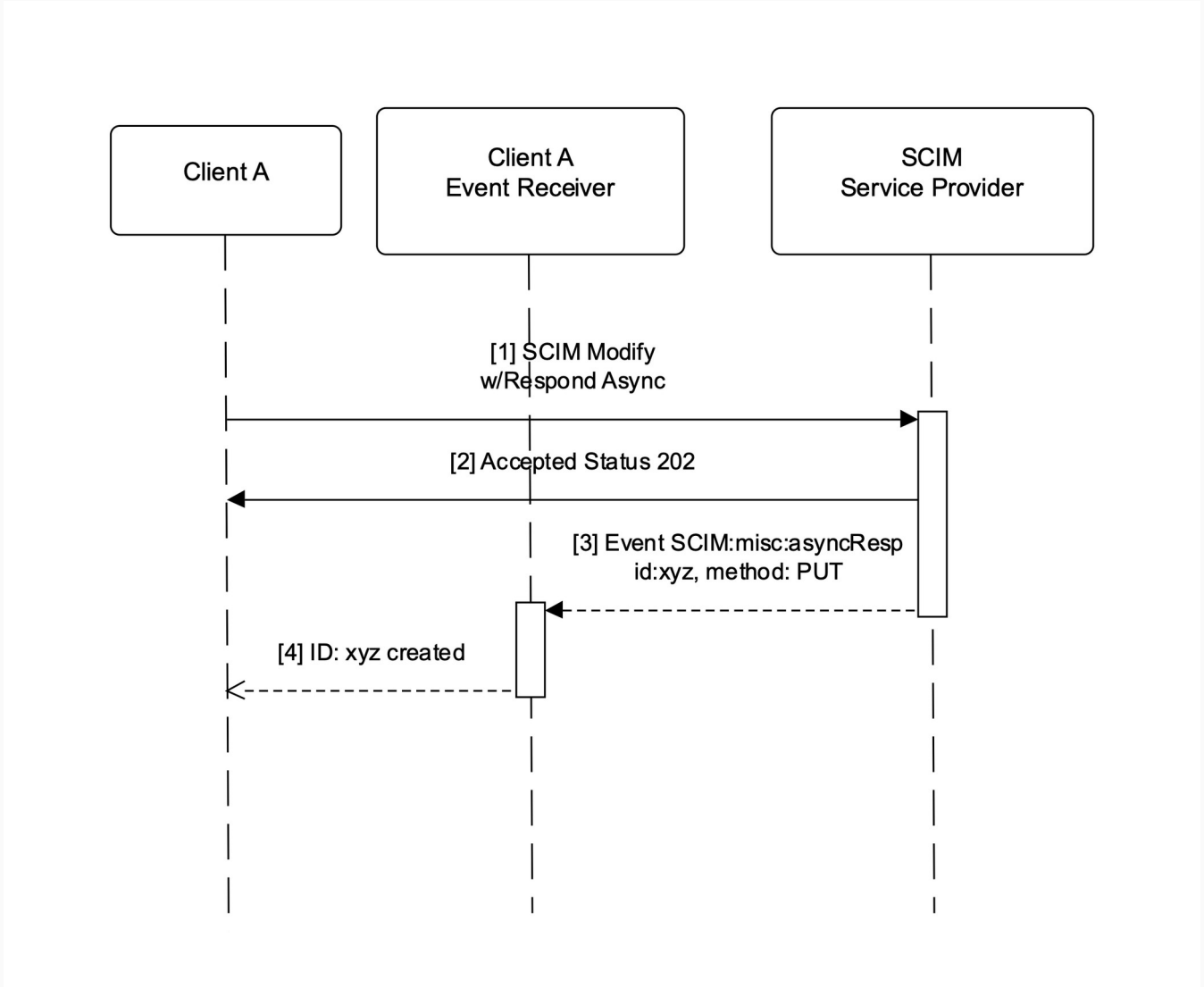
Co-ordinated Provisioning

Use Cases...

- Security Signals
  - Receiver MAY be an IDP, or Security AI System
  - Related to the Shared Signals WG
  - Certain SCIM change events are of interest:
    - Password change / reset
    - Authentication factor changes
    - User account status changes (activation, suspension etc)
    - Password validation failure count

Use Cases...

- Miscellaneous
  - Feed control confirmations (subject add/remove)
  - RFC7240 HTTP Respond Async Request
    - Anticipated due to some workflow or long-running operations
    - Useful if SCIM API implementation needs to be async
      - E.g. performance needed for high-update rate



HTTP Async Request Flow

SCIM Event

- Defines common attributes
- SET Event Claims
  - toe – "time of event" (may be earlier than SET iat)
  - txn – transaction identifier
  - events – a claim carrying one or more events about a txn
  - Sub, iss, iat, jti, aud, sub, exp also have profiling
- Scim Events Claims
  - sub* – to contain the URL of the SCIM Resource impacted
  - A set of event URIs for (Create, Put, Patch, Delete, etc)
    - Sub-claims: id, externalId, data, attributes

Subject/Identifiers

- CAEP/RISC use a more complex subject identifiers supporting more complex semantics
  - E.g. in OIDC, a SET issued by an OIDC client refers to a "sub" which only has meaning to the OP. Normally "iss" disambiguates, but the SET would have "iss" set to the Client!
- SCIM Profile uses SCIM "id" and "externalId" as the agreement on how to identify a SET subject.
  - Subject Identifiers draft is still in last call after several years
    - It's a tough nut to crack
  - SCIM should have a cross-domain relationship where "id" is understood.
  - common understanding of identifiers

```
{
  "jti": "4d3559ec67504aaba65d40b0363faad8",

  "iat": 1458496404,
  "iss":"https://scim.example.com",
  "aud":[
    "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754",
    "https://scim.example.com/Feeds/5d7604516b1d08641d7676ee7"
  ],
  "sub": "/Users/44f6142df96bd6ab61e7521d9",
  "events":{
    "urn:ietf:params:event:SCIM:prov:create":{
      "id":"44f6142df96bd6ab61e7521d9",
      "externalId":"jdoe",
      "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"
        }
      }
    }
  }
}
```


Create Event for Domain Based Replication

Defined Events

- Feed Mgmt
  - Add and Remove Subject to a Feed
- SCIM API Events
  - Create, Put, Patch, Delete
  - Activate, Deactivate
- Signals
  - AuthMethod, pwdReset
- Misc
  - AsyncResp

Delivery

- SCIM Event Profile does not define delivery streams (nor how to manage it)
- Two common stream patterns expected
  - SECEvents defines SET Push and Poll for Point-to-point
    - Shared Signals Framework builds management and other features into SECEVENTs mechanisms
    - See: https://sharedsignals.guide
  - Lots of message bus systems out there (e.g. Kafka)
    - Difficult to force a single choice
    - Buses useful in they may contain historical record and recovery mechanisms
    - Much easier to do connection management at scale
    - Bi-directional flows possible

Out-of-scope

- How receivers should act
  - This does not impact over-the-wire interop
  - In the security world, this impacts demarcation, proprietary, and confidentiality boundaries
    - An event is just a statement of fact not a command
- Delivery mechanism defined by shared signals and bus systems
  - No need to re-invent a SCIM specific protocol?

Questions for WG

1. Currently sub provides a convenient callback URL
    - Do we need it?
2. Feed Management
    - Should we use SSE equivalents?
        - Note that SCIM has a simple subject identifier agreement
3. Is Async Request Response Useful?
4. Other events? (e.g. signals)
5. Other concerns?

Thank You!