SET Token Delivery Using HTTP

Annabelle Backman

IETF 101 – March 2018
SET Token Delivery Using HTTP

• draft-ietf-secevent-delivery-02

• Terminology:
  • Event Transmitter: Party that has the SET.
  • Event Receiver: Party that wants the SET.

• Two mechanisms for transmitting SETs over HTTP:
  • Push: Transmitter sends to Receiver
  • Poll: Receiver pulls from Transmitter
HTTP Push

- Transmitter sends SET to Receiver
- Receiver replies with 202 or 400.
POST /Events HTTP/1.1
Host: receiver.example.com
Accept: application/json
Authorization: Bearer h480djs93hd8
Content-Type: application/secevent+jwt

...SET in request body...
HTTP Push

HTTP/1.1 202 Accepted
HTTP Push

HTTP/1.1 400 Bad Request
Content-Type: application/json

{
   "err": "jwtAud",
   "description": "Invalid audience value."
}
HTTP Poll

• Receiver asks Transmitter for SETs.

• Receiver acknowledges SETs in a later request.
POST /Events HTTP/1.1
Host: transmitter.example.com
Accept: application/json
Authorization: Bearer h480djs93hd8
Content-Type: application/json

{
   "returnImmediately":false
}
HTTP Poll

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

{
    "sets": {
        "set_jti_1": set_1,
        ...
    }
}
HTTP Poll

POST /Events HTTP/1.1
Host: transmitter.example.com
Accept: application/json
Authorization: Bearer h480djs93hd8
Content-Type: application/json

{
    "ack": [ "set_jti_1", "set_jti_2", ..., "set_jti_n" ]
}
HTTP Poll

POST /Events HTTP/1.1
Host: transmitter.example.com
Accept: application/json
Authorization: Bearer h480djs93hd8
Content-Type: application/json

{"setErr": { "set_jti_1": set_err_1, ... }}
HTTP Poll

POST /Events HTTP/1.1
Host: transmitter.example.com
Accept: application/json
Authorization: Bearer h480djs93hd8
Content-Type: application/json

{
    "maxEvents": 10,
    "returnImmediately": false,
    "ack": [ set_jti_1, set_jti_2, ..., set_jti_n ],
    "setErr": { set_jti_1: set_err_1, ... }
}
Current Status

• 02 draft published 2018-03-04

• Open issues:
  • HTTP status codes for error responses
  • Which method is MTI?

• Implementations:
  • Google: HTTP Push implemented
  • Amazon: HTTP Push in progress
  • Microsoft: HTTP Poll in progress/implemented (?)
HTTP status codes for error responses

Error response structure:

```json
{
  err: "jwtAud",
  description: "Invalid audience value."
}
```
## HTTP status codes for error responses

<table>
<thead>
<tr>
<th>err</th>
<th>Description</th>
<th>HTTP Status Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>json</td>
<td>Invalid JSON object.</td>
<td></td>
</tr>
<tr>
<td>jwtParse</td>
<td>Invalid or unparsable JWT or JSON structure.</td>
<td>400</td>
</tr>
<tr>
<td>jwtHdr</td>
<td>In invalid JWT header was detected.</td>
<td></td>
</tr>
<tr>
<td>jwtCrypto</td>
<td>Unable to parse due to unsupported algorithm.</td>
<td></td>
</tr>
<tr>
<td>jws</td>
<td>Signature was not validated.</td>
<td></td>
</tr>
<tr>
<td>jwe</td>
<td>Unable to decrypt JWE encoded data.</td>
<td></td>
</tr>
<tr>
<td>jwtAud</td>
<td>Invalid audience value.</td>
<td></td>
</tr>
<tr>
<td>jwtIss</td>
<td>Issuer not recognized.</td>
<td></td>
</tr>
<tr>
<td>setType</td>
<td>An unexpected Event type was received.</td>
<td></td>
</tr>
<tr>
<td>setParse</td>
<td>Invalid structure was encountered.</td>
<td></td>
</tr>
<tr>
<td>setData</td>
<td>SET event claims incomplete or invalid.</td>
<td></td>
</tr>
<tr>
<td>dup</td>
<td>A duplicate SET was received and has been ignored.</td>
<td></td>
</tr>
</tbody>
</table>
## HTTP status codes for error responses

<table>
<thead>
<tr>
<th>err</th>
<th>Description</th>
<th>HTTP Status Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>json</td>
<td>Invalid JSON object.</td>
<td></td>
</tr>
<tr>
<td>jwtParse</td>
<td>Invalid or unparsable JWT or JSON structure.</td>
<td>400</td>
</tr>
<tr>
<td>jwtHdr</td>
<td>An invalid JWT header was detected.</td>
<td></td>
</tr>
<tr>
<td>jwtCrypto</td>
<td>Unable to parse due to unsupported algorithm.</td>
<td></td>
</tr>
<tr>
<td>jws</td>
<td>Signature was not validated.</td>
<td></td>
</tr>
<tr>
<td>jwe</td>
<td>Unable to decrypt JWE encoded data.</td>
<td></td>
</tr>
<tr>
<td>jwtAud</td>
<td>Invalid audience value.</td>
<td>403</td>
</tr>
<tr>
<td>jwtIss</td>
<td>Issuer not recognized.</td>
<td></td>
</tr>
<tr>
<td>setType</td>
<td>An unexpected Event type was received.</td>
<td>400</td>
</tr>
<tr>
<td>setParse</td>
<td>Invalid structure was encountered.</td>
<td></td>
</tr>
<tr>
<td>setData</td>
<td>SET event claims incomplete or invalid.</td>
<td></td>
</tr>
<tr>
<td>dup</td>
<td>A duplicate SET was received and has been ignored.</td>
<td>400</td>
</tr>
</tbody>
</table>
## Push vs. Poll: What is MTI?

<table>
<thead>
<tr>
<th></th>
<th>Push</th>
<th>Poll</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pros</strong></td>
<td>• Very simple protocol.</td>
<td>• Works for &quot;everyone.&quot;</td>
</tr>
<tr>
<td><strong>Cons</strong></td>
<td>• Receiver has to expose an endpoint.</td>
<td>• Transmitter has to persist all SETs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• More orchestration for receiver.</td>
</tr>
</tbody>
</table>
HTTP Poll by Proxy

- Push between Transmitter and Proxy.
- Poll between Proxy and Receiver.
Split the draft?

ONE DRAFT ENTERS, TWO DRAFTS LEAVE
SECEVENTS: FURY ROAD

HTTP POLL

HTTP PUSH

IETF PROCESS
Fundamental Questions

• Can we define a single MTI protocol for "SET Delivery?"

• *Must* we define a single MTI protocol for "SET Delivery?"

• Who is this protocol for?