

# Push-Based SET Token Delivery Using HTTP

Annabelle Backman

IETF 104 – March 2019

# SET Transmission Request

```
POST /Events HTTP/1.1  
Host: recipient.example.com  
Accept: application/json  
Authorization: Bearer h480djs93hd8  
Content-Type: application/secevent+jwt
```

*...SET in request body...*

# Success Response

```
HTTP/1.1 202 Accepted
```

# Failure Response

```
HTTP/1.1 400 Bad Request
```

```
Content-Type: application/json
```

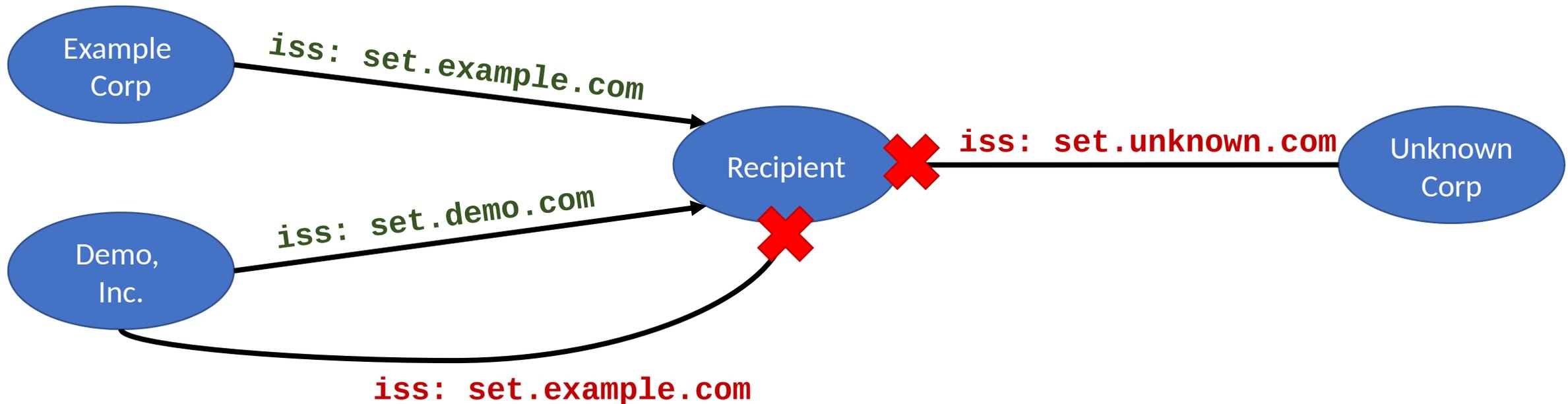
```
{  
  "err": "authentication_failed",  
  "description": "Access token is expired."  
}
```

# Current Status

- 04 draft published 2019-01-23
- WGLC in January/February: not enough responses
- 05 draft published 2019-03-11
- Implementations:
  - Google: live as transmitter, for Google OIDC clients
  - Amazon: WIP

# 03 → 05: Validation

- SET Recipient is willing to receive SETs from the SET Issuer.
- ...and from this SET Transmitter.



# 03 → 05: Redefined Error Codes

<b>invalid_request</b>	The request body cannot be parsed as a SET, or the event payload within the SET does not conform to the event's definition.
<b>invalid_key</b>	One or more keys used to encrypt or sign the SET is invalid or otherwise unacceptable to the SET Recipient.
<b>authentication_failed</b>	The SET Recipient could not authenticate the SET Transmitter from the contents of the request.
<b>access_denied</b>	The SET Transmitter is not authorized to transmit the provided SET to the SET Recipient.

# 03 → 05: Miscellaneous

- Accept - Language and Content - Language for error descriptions
- Required when using TLS:
  - at least TLS 1.2
  - newest TLS version "that meets [Recipient's] security requirements."
- Implementations:
  - Google: live as transmitter, for Google OIDC clients
  - Amazon: WIP

# WGLC and Continued Work

**Please read and review!**