Security Events Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 6, 2018

P. Hunt, Ed.
Oracle
M. Jones
Microsoft
W. Denniss
Google
M. Ansari
Cisco

February 2, 2018

Security Event Token (SET)
draft-ietf-secevent-token-05

Abstract

This specification defines the Security Event Token (SET) data structure. A SET describes a statement of fact from the perspective of an issuer, which is intended to be shared with one or more recipients. A SET is a JSON Web Token (JWT), which can be optionally signed and/or encrypted. SETs can be distributed via protocols such as HTTP.

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 August 6, 2018.

Copyright Notice

Table of Contents

1. Introduction and Overview

This specification defines an extensible Security Event Token (SET) data structure, which can be exchanged using protocols such as HTTP. The specification builds on the JSON Web Token (JWT) format [RFC7519]

in order to provide a self-contained token that can be optionally signed using JSON Web Signature (JWS) [RFC7515] and/or encrypted using JSON Web Encryption (JWE) [RFC7516].

This specification profiles the use of JWT for the purpose of issuing Security Event Tokens (SETs). This specification defines a base format used by profiling specifications to define actual events and their meanings. This specification uses non-normative example events to demonstrate how events can be constructed.

This specification is scoped to security and identity related events. While security event tokens may be used for other purposes, the specification only considers security and privacy concerns relevant to identity and personal information.

Security Events are not commands issued between parties. A security event is a statement of fact from the perspective of an issuer about the state of a security subject (e.g., a web resource, token, IP address, the issuer itself) that the issuer controls or is aware of, that has changed in some way (explicitly or implicitly). A security subject MAY be permanent (e.g., a user account) or temporary (e.g., an HTTP session) in nature. A state change could describe a direct change of entity state, an implicit change of state, or other higher-level security statements such as:

- The creation, modification, removal of a resource.
- The resetting or suspension of an account.
- The revocation of a security token prior to its expiry.
- The logout of a user session. Or,
- An indication that a user has been given control of an email identifier that was previously controlled by another user.

While subject state changes are often triggered by a user agent or security subsystem, the issuance and transmission of an event may occur asynchronously and in a back channel to the action that caused the change that generated the security event. Subsequently, an Event Recipient, having received a SET, validates and interprets the received SET and takes its own independent actions, if any. For example, having been informed of a personal identifier being associated with a different security subject (e.g., an email address is being used by someone else), the Event Recipient may choose to ensure that the new user is not granted access to resources associated with the previous user. Or, the Event Recipient may not have any relationship with the subject, and no action is taken.

While Event Recipients will often take actions upon receiving SETs, security events cannot be assumed to be commands or requests. The intent of this specification is to define a syntax for statements of fact that Event Recipients may interpret for their own purposes. As such, SETs have no capability for error signaling to ensure the validation of a received SET.

1.1. Notational Conventions

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.

For purposes of readability, examples are not URL encoded. Implementers MUST percent encode URLs as described in Section 2.1 of [RFC3986].

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.

1.2. Definitions

The following definitions are used with SETs:

Security Event Token (SET)
: A SET is a JWT [RFC7519] conforming to this specification that is distributed to one or more Event Recipients.

Event Issuer
: A service provider that creates SETs to be sent to other providers known as Event Recipients.

Event Recipient
: An Event Recipient is an entity that receives SETs through some distribution method. An Event Recipient is the same entity referred as a "recipient" or "receiver" in [RFC7519] and related specifications.

Subject
: A SET describes an event or state change that has occurred about a Subject. A Subject might, for instance, be a principal (e.g., Section 4.1.2 of [RFC7519]), a web resource, an entity such as an IP address, or the issuer of the SET.

Profiling Specification
: A specification that profiles the SET data structure to define one or more specific event types and their associated claims and processing rules.

2. The Security Event Token (SET)

A SET is a JWT [RFC7519] data structure that represents one or more related aspects of a security event about a Subject. The JWT Claims Set in a SET has the following structure:

- The top-level claims in the JWT Claims Set are called the SET "envelope". Some of these claims are present in every SET; others will be specific to particular SET profiles or profile families. Claims in the envelope SHOULD be registered in the "JSON Web Token Claims" registry [IANA.JWT.Claims] or be Public Claims or Private Claims, as defined in [RFC7519].

- Envelope claims that are profiled and defined in this specification are used to validate the SET and provide information about the event data included in the SET. The claim "events" contains the event identifiers and event-specific data expressed about the Security Subject. The envelope MAY include event-specific or profile-specific data.

- Each member of the "events" JSON object is a name/value pair. The JSON member name is a URI string value is an event identifier, and the corresponding value is a JSON object known as the event "payload". The payload JSON object contains claims that pertain to that event identifier and need not be registered as JWT claims. These claims are defined by the Profiling Specification that defines the event. An event with no payload claims SHALL be represented as the empty JSON object ("{}").

- When multiple event identifiers are contained in a SET, they represent multiple aspects of the same state transition that occurred to the Security Subject. They are not intended to be used to aggregate distinct events about the same subject. Beyond this, the interpretation of SETs containing multiple event identifiers is out of scope for this specification; Profiling Specifications MAY define their own rules regarding their use of SETs containing multiple event identifiers, as described in Section 3. Possible uses of multiple values include, but are not limited to:

  * Values to provide classification information (e.g., threat type or level).

* Additions to existing event representations.

* Values used to link potential series of events.

* Specific-purpose event URIs used between particular Event Issuers and Event Recipients.

2.1. Illustrative Examples

2.1.1. SCIM Example

The following is a non-normative example showing the JWT Claims Set for a hypothetical SCIM [RFC7644] password reset SET. This example uses a second "events" value ("https://example.com/scim/event/passwordResetExt") to convey additional information about the state change -- in this case, the current count of reset attempts:

```
{
  "iss": "https://scim.example.com",
  "iat": 1458496025,
  "jti": "3d0c3cf797584bd193bd0fb1bd4e7d30",
  "aud": [
    "https://jhub.example.com/Feeds/98d52461fa5bbc879593b7754",
    "https://jhub.example.com/Feeds/5d7604516b1d08641d7676ee7"
  ],
  "sub": "https://scim.example.com/Users/44f6142df96bd6ab61e7521d9",
  "events": {
    "urn:ietf:params:scim:event:passwordReset":
      { "id": "44f6142df96bd6ab61e7521d9"},
    "https://example.com/scim/event/passwordResetExt":
      { "resetAttempts": 5}
  }
}
```

Figure 1: Example SCIM Password Reset Event

The JWT Claims Set consists of:

o The "events" claim specifying the hypothetical SCIM URN ("urn:ietf:params:scim:event:passwordReset") for a password reset, and a second value, "https://example.com/scim/event/passwordResetExt", that is used to provide additional event information such as the current count of resets.

o The "iss" claim, denoting the Event Issuer.

o The "sub" claim, specifying the SCIM resource URI that was affected.

- The "aud" claim, specifying the intended audiences for the event. (The syntax of the "aud" claim is defined in Section 4.1.3 of [RFC7519].)

In this example, the SCIM event indicates that a password has been updated and the current password reset count is 5. Notice that the value for "resetAttempts" is in the event payload of an event used to convey this information.

2.1.2. Logout Example

Here is another example JWT Claims Set for a security event token, this one for a Logout Token:

```
{
   "iss": "https://server.example.com",
   "sub": "248289761001",
   "aud": "s6BhdRkqt3",
   "iat": 1471566154,
   "jti": "bWJq",
   "sid": "08a5019c-17e1-4977-8f42-65a12843ea02",
   "events": {
     "http://schemas.openid.net/event/backchannel-logout": {}
   }
}
```

Figure 2: Example OpenID Back-Channel Logout Event

Note that the above SET has an empty JSON object and uses the JWT registered claims "sub" and "sid" to identify the subject that was logged out.

2.1.3. Consent Example

In the following example JWT Claims Set, a fictional medical service collects consent for medical actions and notifies other parties. The individual for whom consent is identified was originally authenticated via OpenID Connect. In this case, the issuer of the security event is an application rather than the OpenID provider:

```
{
  "iss": "https://my.med.example.org",
  "iat": 1458496025,
  "jti": "fb4e75b5411e4e19b6c0fe87950f7749",
  "aud": [
    "https://rp.example.com"
  ],
  "events": {
    "https://openid.net/heart/specs/consent.html": {
      "iss": "https://connect.example.com",
      "sub": "248289761001",
      "consentUri": [
        "https://terms.med.example.org/labdisclosure.html#Agree"
      ]
    }
  }
}
```

Figure 3: Example Consent Event

In the above example, the attribute "iss" contained within the payload for the event "https://openid.net/heart/specs/consent.html" refers to the issuer of the Security Subject ("sub") rather than the event issuer "https://my.med.example.org". They are distinct from the top-level value of "iss", which always refers to the issuer of the event -- a medical consent service that is a relying party to the OpenID Provider.

2.1.4. RISC Example

The following example JWT Claims Set is for an account disabled event. This example was taken from a working draft of the RISC events specification, where RISC is the OpenID RISC (Risk and Incident Sharing and Coordination) working group [RISC]. The example is subject to change.

```
{
  "iss": "https://idp.example.com/",
  "jti": "756E69717565206964656E746966696572",
  "iat": 1508184845,
  "aud": "636C69656E745F6964",
  "events": {
    "http://schemas.openid.net/secevent/risc/event-type/\
    account-disabled": {
      "subject": {
        "subject_type": "iss-sub",
        "iss": "https://idp.example.com/",
        "sub": "7375626A656374"
      },
      "reason": "hijacking",
      "cause-time": 1508012752
    }
  }
}
```

Figure 4: Example RISC Event

Notice that parameters to the event are included in the event payload, in this case, the "reason" and "cause-time" values. The subject of the event is identified using the "subject" payload value, which itself is a JSON object.

2.2. Core SET Claims

The following claims from [RFC7519] are profiled for use in SETs:

iss
: A string identifying the service provider publishing the SET (the issuer). In some cases, the SET issuer is not the issuer of the Security Subject. Therefore, implementers cannot assume that the issuers are the same unless the Profiling Specification specifies that they are for SETs conforming to that profile. This claim is REQUIRED.

iat
: As defined by Section 4.1.6 of [RFC7519], a value representing when the event was issued. This claim is REQUIRED.

jti
As defined by Section 4.1.7 of [RFC7519] contains a unique identifier for an event. The identifier SHOULD be unique within a particular event feed and MAY be used by clients to track whether a particular event has already been received. This claim is REQUIRED.

aud
The syntax of the claim is as defined in Section 4.1.3 of [RFC7519]. This claim contains one or more audience identifiers for the SET. This claim is RECOMMENDED.

sub
As defined by Section 4.1.2 of [RFC7519], a String or URI value representing the principal or the subject of the SET. This is usually the entity whose "state" was changed. For example, an IP Address was added to a black list. A URI representing a user resource that was modified. A token identifier for a revoked token. If used, the Profiling Specification SHOULD define the content and format semantics for the value. This claim is OPTIONAL, as the principal for any given profile may already be identified without the inclusion of a subject claim. Note that some SET profiles MAY choose to convey event subject information in the event payload (either using the "sub" member name or another name), particularly if the subject information is relative to issuer information that is also conveyed in the event payload, which may be the case for some identity SET profiles.

exp
As defined by Section 4.1.4 of [RFC7519], this claim is time after which the JWT MUST NOT be accepted for processing. In the context of a SET however, this notion does not apply, since a SET represents something that has already occurred and is historical in nature. While some profiles MAY choose to use this claim, its use is NOT RECOMMENDED.

The following new claims are defined by this specification:

events
This claim contains a set of event statements that each provide information describing a single logical event that has occurred about a Security Subject (e.g., a state change to the subject). Multiple event identifiers with the same value MUST NOT be used. The "events" claim SHOULD NOT be used to express multiple independent logical events.

The value of the "events" claim is a JSON object whose members are name/value pairs whose names are URIs identifying the event

statements being expressed. Event identifiers SHOULD be stable values (e.g., a permanent URL for an event specification). For each name present, the corresponding value MUST be a JSON object. The JSON object MAY be an empty object ("{}"), or it MAY be a JSON object containing data described by the Profiling Specification.

txn
An OPTIONAL string value that represents a unique transaction identifier. In cases in which multiple related JWTs are issued, the transaction identifier claim can be used to correlate these related JWTs.

toe
A value that represents the date and time at which the event occurred. This value is a NumericDate (see Section 2 of [RFC7519]). By omitting this claim, the issuer indicates that they are not sharing an event time with the recipient. (Note that in some use cases, the represented time might be approximate.) This claim is OPTIONAL.

2.3. Explicit Typing of SETs

This specification registers the "application/secevent+jwt" media type, which can be used to indicate that the content is a SET. SETs MAY include this media type in the "typ" header parameter of the JWT representing the SET to explicitly declare that the JWT is a SET. This MUST be included if the SET could be used in an application context in which it could be confused with other kinds of JWTs.

Per the definition of "typ" in Section 4.1.9 of [RFC7515], it is RECOMMENDED that the "application/" prefix be omitted. Therefore, the "typ" value used SHOULD be "secevent+jwt".

2.4. Security Event Token Construction

This section describes how to construct a SET.

The following is an example JWT Claims Set for a hypothetical SCIM SET (which has been formatted for readability):

```
{
  "iss": "https://scim.example.com",
  "iat": 1458496404,
  "jti": "4d3559ec67504aaba65d40b0363faad8",
  "aud": [
   "https://scim.example.com/Feeds/98d52461fa5bbc879593b7754",
   "https://scim.example.com/Feeds/5d7604516b1d08641d7676ee7"
  ],

  "events": {
    "urn:ietf:params:scim:event:create": {
      "ref":
        "https://scim.example.com/Users/44f6142df96bd6ab61e7521d9",
      "attributes": ["id", "name", "userName", "password", "emails"]
    }
  }
}
```

Figure 5: Example Event Claims

The JSON Claims Set is encoded per [RFC7519].

In this example, the SCIM SET claims are encoded in an unsecured JWT. The JOSE Header for this example is:

```
{"typ":"secevent+jwt","alg":"none"}
```

Base64url encoding of the octets of the UTF-8 representation of the JOSE Header yields:

```
eyJ0eXAiOiJzZWNldmVudCtqd3QiLCJhbGciOiJub25lIn0
```

The above example JWT Claims Set is encoded as follows:

```
eyJqdGkiOiI0ZDM1NTllYzY3NTA0YWFiYTY1ZDQwYjAzNjNmYWFkOCIsImlhdCI6MTQ1
ODQ5NjQwNCwiaXNzIjoiaHR0cHM6Ly9zY2ltLmV4YW1wbGUuY29tIiwiYXVkIjpbImh0
dHBzOi8vc2NpbS5leGFtcGxlLmNvbS9GZWVkcy85OGQ1MjQ2MWZhNWJiYzg3OTU5M2I3
NzU0IiwiaHR0cHM6Ly9zY2ltLmV4YW1wbGUuY29tL0ZlZWRzLzVkNzYwNDUxNmIxZDA4
NjQxZDc2NzZlZTciXSwiZXZlbnRzIjp7InVybjppZXRmOnBhcmFtczpzY2ltOmV2ZW50
OmNyZWF0ZSI6eyJyZWYiOiJodHRwczovL3NjaW0uZXhhbXBsZS5jb20vVXNlcnMvNDRm
NjE0MmRmOTZiZDZhYjYxZTc1MjFkOSIsImF0dHJpYnV0ZXMiOlsiaWQiLCJuYW1lIiwi
dXNlck5hbWUiLCJwYXNzd29yZCIsImVtYWlscyJdfX19
```

The encoded JWS signature is the empty string. Concatenating the parts yields this complete SET:

eyJ0eXAiOiJzZWNldmVudCtqd3QiLCJhbGciOiJub25lIn0.
eyJqdGkiOiI0ZDM1NTllYzY3NTA0YWFiYTY1ZDQwYjAzNjNmYWFkOCIsImlhdCI6MTQ1
ODQ5NjQwNCwiaXNzIjoiaHR0cHM6Ly9zY2ltLmV4YW1wbGUuY29tIiwiYXVkIjpbImh0
dHBzOi8vc2NpbS5leGFtcGxlLmNvbS9GZWVkcy85OGQ1MjQ2MWZhNWJiYzg3OTU5M2I3
NzU0IiwiaHR0cHM6Ly9zY2ltLmV4YW1wbGUuY29tL0ZlZWRzLzVkNzYwNDUxNmIxZDA4
NjQxZDc2NzZlZTciXSwiZXZlbnRzIjp7InVybjppZXRmOnBhcmFtczpzY2ltOmV2ZW50
OmNyZWF0ZSI6eyJyZWYiOiJodHRwczovL3NjaW0uZXhhbXBsZS5jb20vVXNlcnMvNDRm
NjE0MmRmOTZiZDZhYjYxZTc1MjFkOSIsImF0dHJpYnV0ZXMiOlsiaWQiLCJuYW1lIiwi
dXNlck5hbWUiLCJwYXNzd29yZCIsImVtYWlscyJdfX19.

Figure 6: Example Unsecured Security Event Token

For the purpose of having a simpler example in Figure 6, an unsecured token is shown. When SETs are not signed or encrypted, the Event Recipient MUST employ other mechanisms such as TLS to provide integrity, confidentiality, and issuer validation, as needed by the application.

When validation (i.e., auditing), or additional transmission security is required, JWS signing and/or JWE encryption MAY be used. To create and or validate a signed and/or encrypted SET, follow the instructions in Section 7 of [RFC7519].

3. Requirements for SET Profiles

Profiling Specifications for SETs define the syntax and semantics of SETs conforming to that SET profile and rules for validating those SETs. The syntax defined by profiling specifications includes what claims and event payload values are used by SETs utilizing the profile.

Defining the semantics of the SET contents for SETs utilizing the profile is equally important. Possibly most important is defining the procedures used to validate the SET issuer and to obtain the keys controlled by the issuer that were used for cryptographic operations used in the JWT representing the SET. For instance, some profiles may define an algorithm for retrieving the SET issuer's keys that uses the "iss" claim value as its input. Likewise, if the profile allows (or requires) that the JWT be unsecured, the means by which the integrity of the JWT is ensured MUST be specified.

Profiling Specifications MUST define how the event Subject is identified in the SET, as well as how to differentiate between the event Subject's Issuer and the SET Issuer, if applicable. It is NOT RECOMMENDED for Profiling Specifications to use the "sub" claim in

cases in which the Subject is not globally unique and has a different Issuer from the SET itself.

Among the syntax and semantics of SETs that Profiling Specifications define is whether and how multiple "events" values are used for SETs conforming to those profiles. Many valid choices are possible. For instance, some profiles might allow multiple event identifiers to be present and specify that any that are not understood by recipients be ignored, thus enabling extensibility. Other profiles might allow multiple event identifiers to be present but require that all be understood if the SET is to be accepted. Some profiles might require that only a single value be present. All such choices are within the scope of Profiling Specifications to define.

Profiling Specifications MUST clearly specify the steps that a recipient of a SET utilizing that profile MUST perform to validate that the SET is both syntactically and semantically valid.

4. Security Considerations

4.1. Confidentiality and Integrity

SETs may contain sensitive information. Therefore, methods for distribution of events SHOULD require the use of a transport-layer security mechanism when distributing events. Parties MUST support TLS 1.2 [RFC5246] and MAY support additional transport-layer mechanisms meeting its security requirements. When using TLS, the client MUST perform a TLS/SSL server certificate check, per [RFC6125]. Implementation security considerations for TLS can be found in "Recommendations for Secure Use of TLS and DTLS" [RFC7525].

Security Events distributed through third parties or that carry personally identifiable information SHOULD be encrypted using JWE [RFC7516] or secured for confidentiality by other means.

Unless integrity of the JWT is ensured by other means, it MUST be signed using JWS [RFC7515] so that the SET can be authenticated and validated by the Event Recipient.

4.2. Delivery

This specification does not define a delivery mechanism for SETs. In addition to confidentiality and integrity (discussed above), implementers and Profiling Specifications MUST consider the consequences of delivery mechanisms that are not secure and/or not assured. For example, while a SET may be end-to-end secured using JWE encrypted SETs, without TLS, there is no assurance that the

correct endpoint received the SET and that it could be successfully processed.

4.3. Sequencing

This specification defines no means of ordering multiple SETs in a sequence. Depending on the type and nature of the events represented by SETs, order may or may not matter. For example, in provisioning, event order is critical -- an object cannot be modified before it is created. In other SET types, such as a token revocation, the order of SETs for revoked tokens does not matter. If, however, the event conveys a logged in or logged out status for a user subject, then order becomes important.

Profiling Specifications and implementers SHOULD take caution when using timestamps such as "iat" to define order. Distributed systems will have some amount of clock skew. Thus, time by itself will not guarantee order.

Specifications profiling SET SHOULD define a mechanism for detecting order or sequence of events when the order matters. For example, the "txn" claim could contain an ordered value (e.g., a counter) that the issuer includes.

4.4. Timing Issues

When SETs are delivered asynchronously and/or out-of-band with respect to the original action that incurred the security event, it is important to consider that a SET might be delivered to an Event Recipient in advance of or behind the process that caused the event. For example, a user having been required to log out and then log back in again, may cause a logout SET to be issued that may arrive at the same time as the user agent accesses a web site having just logged in. If timing is not handled properly, the effect would be to erroneously treat the new user session as logged out. Profiling Specifications SHOULD be careful to anticipate timing and subject selection information. For example, it might be more appropriate to cancel a "session" rather than a "user". Alternatively, the specification could use timestamps that allow new sessions to be started immediately after a stated logout event time.

4.5. Distinguishing SETs from ID Tokens

Because [RFC7519] states that "all claims that are not understood by implementations MUST be ignored", there is a consideration that a SET might be confused with ID Token [OpenID.Core] if a SET is mistakenly or intentionally used in a context requiring an ID Token. If a SET could otherwise be interpreted as a valid ID Token (because it

includes the required claims for an ID Token and valid issuer and audience claim values for an ID Token) then that SET profile MUST require that the "exp" claim not be present in the SET. Because "exp" is a required claim in ID Tokens, valid ID Token implementations will reject such a SET if presented as if it were an ID Token.

Excluding "exp" from SETs that could otherwise be confused with ID Tokens is actually defense in depth. In any OpenID Connect contexts in which an attacker could attempt to substitute a SET for an ID Token, the SET would actually already be rejected as an ID Token because it would not contain the correct "nonce" claim value for the ID Token to be accepted in contexts for which substitution is possible.

Note that the use of explicit typing, as described in Section 2.3, will not achieve disambiguation between ID Tokens and SETs, as the ID Token validation rules do not use the "typ" header parameter value.

4.6. Distinguishing SETs from Access Tokens

OAuth 2.0 [RFC6749] defines access tokens as being opaque. Nonetheless, some implementations implement access tokens as JWTs. Because the structure of these JWTs is implementation-specific, ensuring that a SET cannot be confused with such an access token is therefore likewise, in general, implementation specific. Nonetheless, it is recommended that SET profiles employ the following strategies to prevent possible substitutions of SETs for access tokens in contexts in which that might be possible:

- Prohibit use of the "exp" claim, as is done to prevent ID Token confusion.
- Where possible, use a separate "aud" claim value to distinguish between the Event Recipient and the protected resource that is the audience of an access token.
- Modify access token validation systems to check for the presence of the "events" claim as a means to detect security event tokens. This is particularly useful if the same endpoint may receive both types of tokens.
- Employ explicit typing, as described in Section 2.3, and modify access token validation systems to use the "typ" header parameter value.

4.7. Distinguishing SETs from other kinds of JWTs

JWTs are now being used in application areas beyond the identity applications in which they first appeared. For instance, the Session Initiation Protocol (SIP) Via Header Field [RFC8055] and Personal Assertion Token (PASSporT) [I-D.ietf-stir-passport] specifications both define JWT profiles that use mostly or completely different sets of claims than are used by ID Tokens. If it would otherwise be possible for an attacker to substitute a SET for one of these (or other) kinds of JWTs, then the SET profile must be defined in such a way that any substituted SET will result in its rejection when validated as the intended kind of JWT.

The most direct way to prevent confusion is to employ explicit typing, as described in Section 2.3, and modify applicable token validation systems to use the "typ" header parameter value. This approach can be employed for new systems but may not be applicable to existing systems.

Another way to ensure that a SET is not confused with another kind of JWT is to have the JWT validation logic reject JWTs containing an "events" claim unless the JWT is intended to be a SET. This approach can be employed for new systems but may not be applicable to existing systems.

For many use cases, the simplest way to prevent substitution is requiring that the SET not include claims that are required for the kind of JWT that might be the target of an attack. For example, for [RFC8055], the "sip_callid" claim could be omitted and for [I-D.ietf-stir-passport], the "orig" claim could be omitted.

In many contexts, simple measures such as these will accomplish the task, should confusion otherwise even be possible. Note that this topic is being explored in a more general fashion in JSON Web Token Best Current Practices [I-D.ietf-oauth-jwt-bcp]. The proposed best practices in that draft may also be applicable for particular SET profiles and use cases.

5. Privacy Considerations

If a SET needs to be retained for audit purposes, the signature can be used to provide verification of its authenticity.

Event Issuers SHOULD attempt to specialize SETs so that their content is targeted to the specific business and protocol needs of the intended Event Recipients.

When sharing personally identifiable information or information that is otherwise considered confidential to affected users, Event Issuers and Recipients MUST have the appropriate legal agreements and user consent and/or terms of service in place.

The propagation of subject identifiers can be perceived as personally identifiable information. Where possible, Event Issuers and Recipients SHOULD devise approaches that prevent propagation -- for example, the passing of a hash value that requires the Event Recipient to know the subject.

In some cases, it may be possible for an Event Recipient to correlate different events and thereby gain information about a subject that the Event Issuer did not intend to share. For example, an Event Recipient might be able to use "iat" values or highly precise "toe" values to determine that two otherwise un-relatable events actually relate to the same real-world event. The union of information from both events could allow an Event Recipient to de-anonymize data or recognize that unrelated identifiers relate to the same individual. Event Issuers SHOULD take steps to minimize the chance of event correlation, when such correlation would constitute a privacy violation. For instance, they could use approximate values for the "toe" claim or arbitrarily delay SET issuance, where such delay can be tolerated.

6. IANA Considerations

6.1. JSON Web Token Claims Registration

This specification registers the "events", "toe", and "txn" claims in the IANA "JSON Web Token Claims" registry [IANA.JWT.Claims] established by [RFC7519].

6.1.1. Registry Contents

- Claim Name: "events"
- Claim Description: Security Event URI
- Change Controller: IESG
- Specification Document(s): Section 2.2 of [[this specification]]

- Claim Name: "toe"
- Claim Description: Time of Event
- Change Controller: IESG
- Specification Document(s): Section 2.2 of [[this specification]]

- Claim Name: "txn"
- Claim Description: Transaction Identifier
- Change Controller: IESG

- Specification Document(s): Section 2.2 of [[this specification]]

6.2. Media Type Registration

6.2.1. Registry Contents

This section registers the "application/secevent+jwt" media type [RFC2046] in the "Media Types" registry [IANA.MediaTypes] in the manner described in [RFC6838], which can be used to indicate that the content is a SET.

- Type name: application
- Subtype name: secevent+jwt
- Required parameters: n/a
- Optional parameters: n/a
- Encoding considerations: 8bit; A SET is a JWT; JWT values are encoded as a series of base64url-encoded values (some of which may be the empty string) separated by period ('.') characters.
- Security considerations: See the Security Considerations section of [[this specification]]
- Interoperability considerations: n/a
- Published specification: Section 2.3 of [[this specification]]
- Applications that use this media type: TBD
- Fragment identifier considerations: n/a
- Additional information:

  Magic number(s): n/a
  File extension(s): n/a
  Macintosh file type code(s): n/a

- Person & email address to contact for further information: Michael B. Jones, mbj@microsoft.com
- Intended usage: COMMON
- Restrictions on usage: none
- Author: Michael B. Jones, mbj@microsoft.com
- Change controller: IESG
- Provisional registration? No

7. References

7.1. Normative References

[IANA.JWT.Claims]
IANA, "JSON Web Token Claims",
<http://www.iana.org/assignments/jwt>.

[IANA.MediaTypes]
IANA, "Media Types",
<http://www.iana.org/assignments/media-types>.

[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>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/info/rfc5246>.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <https://www.rfc-editor.org/info/rfc6125>.

[RFC6749] Hardt, D., Ed., "The OAuth 2.0 Authorization Framework", RFC 6749, DOI 10.17487/RFC6749, October 2012, <https://www.rfc-editor.org/info/rfc6749>.

[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>.

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <https://www.rfc-editor.org/info/rfc7525>.

[RFC7617] Reschke, J., "The 'Basic' HTTP Authentication Scheme", RFC 7617, DOI 10.17487/RFC7617, September 2015, <https://www.rfc-editor.org/info/rfc7617>.

[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>.

7.2. Informative References

[I-D.ietf-oauth-jwt-bcp]
Sheffer, Y., Hardt, D., and M. Jones, "JSON Web Token Best Current Practices", draft-ietf-oauth-jwt-bcp-00 (work in progress), July 2017.

[I-D.ietf-stir-passport]
Wendt, C. and J. Peterson, "Personal Assertion Token (PASSporT)", draft-ietf-stir-passport-11 (work in progress), February 2017.

[OpenID.Core]
Sakimura, N., Bradley, J., Jones, M., de Medeiros, B., and C. Mortimore, "OpenID Connect Core 1.0", November 2014, <http://openid.net/specs/openid-connect-core-1_0.html>.

[RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, DOI 10.17487/RFC2046, November 1996, <https://www.rfc-editor.org/info/rfc2046>.

[RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC 6838, DOI 10.17487/RFC6838, January 2013, <https://www.rfc-editor.org/info/rfc6838>.

[RFC7009] Lodderstedt, T., Ed., Dronia, S., and M. Scurtescu, "OAuth 2.0 Token Revocation", RFC 7009, DOI 10.17487/RFC7009, August 2013, <https://www.rfc-editor.org/info/rfc7009>.

[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>.

[RFC7516] Jones, M. and J. Hildebrand, "JSON Web Encryption (JWE)", RFC 7516, DOI 10.17487/RFC7516, May 2015, <https://www.rfc-editor.org/info/rfc7516>.

[RFC7517] Jones, M., "JSON Web Key (JWK)", RFC 7517, DOI 10.17487/RFC7517, May 2015, <https://www.rfc-editor.org/info/rfc7517>.

[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>.

[RFC8055] Holmberg, C. and Y. Jiang, "Session Initiation Protocol (SIP) Via Header Field Parameter to Indicate Received Realm", RFC 8055, DOI 10.17487/RFC8055, January 2017, <https://www.rfc-editor.org/info/rfc8055>.

[RISC] OpenID Foundation, "OpenID Risk and Incident Sharing and Coordination (RISC) Working Group", <http://openid.net/wg/risc/>.

[saml-core-2.0] Internet2, "Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0", March 2005.

Appendix A. Acknowledgments

The editors would like to thank the members of the IETF SCIM working group, which began discussions of provisioning events starting with draft-hunt-scim-notify-00 in 2015.

The editors would like to thank the participants in the IETF id-event mailing list, the Security Events working group, and related working groups for their contributions to this specification.

Appendix B. Change Log

[[to be removed by the RFC Editor before publication as an RFC]]

From the original draft-hunt-idevent-token:

Draft 01 - PH - Renamed eventUris to events

Draft 00 - PH - First Draft

Draft 01 - PH - Fixed some alignment issues with JWT. Remove event type attribute.

Draft 02 - PH - Renamed to Security Events, removed questions, clarified examples and intro text, and added security and privacy section.

Draft 03 - PH

- General edit corrections from Sarah Squire
- Changed "event" term to "SET"
- Corrected author organization for William Denniss to Google

Changed definition of SET to be 2 parts, an envelope and 1 or more payloads.

Clarified that the intent is to express a single event with optional extensions only.

- mbj - Registered "events" claim, and proof-reading corrections.

Draft 04 - PH -

- Re-added the "sub" claim with clarifications that any SET type may use it.
- Added additional clarification on the use of envelope vs. payload attributes
- Added security consideration for event timing.
- Switched use of "attribute" to "claim" for consistency.
- Revised examples to put "sub" claim back in the top level.
- Added clarification that SETs typically do not use "exp".
- Added security consideration for distinguishing Access Tokens and SETs.

Draft 05 - PH - Fixed find/replace error that resulted in claim being spelled claimc

Draft 06 - PH -

- Corrected typos
- New txn claim
- New security considerations Sequencing and Timing Issues

Draft 07 -

- PH - Moved payload objects to be values of event URI attributes, per discussion.
- mbj - Applied terminology consistency and grammar cleanups.

Draft 08 - PH -

- Added clarification to status of examples

- Changed from primary vs. extension to state that multiple events may be expressed, some of which may or may not be considered extensions of others (which is for the subscriber or profiling specifications to determine).

- Other editorial changes suggested by Yaron

From draft-ietf-secevent-token:

Draft 00 - PH - First WG Draft based on draft-hunt-idevent-token

Draft 01 - PH - Changes as follows:

- Changed terminology away from pub-sub to transmitter/receiver based on WG feedback

- Cleaned up/removed some text about extensions (now only used as example)

- Clarify purpose of spec vs. future profiling specs that define actual events

Draft 02 - Changes are as follows:

- mbj - Added the Requirements for SET Profiles section.

- mbj - Expanded the Security Considerations section to describe how to prevent confusion of SETs with ID Tokens, access tokens, and other kinds of JWTs.

- mbj - Registered the "application/secevent+jwt" media type and defined how to use it for explicit typing of SETs.

- mbj - Clarified the misleading statement that used to say that a SET conveys a single security event.

- mbj - Added a note explicitly acknowledging that some SET profiles may choose to convey event subject information in the event payload.

- PH - Corrected encoded claim example on page 10.

- mbj - Applied grammar corrections.

Draft 03 - Changes are as follows:

- pjh - Corrected old "subscriber" to "Event Receiver". Added clarification in definition that Event Receiver is the same as JWT recipient.

- pjh - Added definition for "toe" (and IANA registration).

- pjh - Removed "nbf" claim.

- pjh - Figure 3, moved "sub" to the events payload next to "iss".

- pjh - Clarified the use of "nonce" in contexts where substitution is possible.

- mbj - Addressed WGLC comments by Nat Sakimura.

- mbj - Addressed WGLC comments by Annabelle Backman.

- mbj - Addressed WGLC comments by Marius Scurtescu.

Draft 04 - mbj - Changes were as follows:

- Clarified that all "events" values must represent aspects of the same state change that occurred to the subject -- not an aggregation of unrelated events about the subject.

- Removed ambiguities about the roles of multiple "events" values and the responsibilities of profiling specifications for defining how and when they are used.

- Corrected places where the term JWT was used when what was actually being discussed was the JWT Claims Set.

- Addressed terminology inconsistencies. In particular, standardized on using the term "issuer" to align with JWT terminology and the "iss" claim. Previously the term "transmitter" was sometimes used and "issuer" was sometimes used. Likewise, standardized on using the term "recipient" instead of "receiver" for the same reasons.

- Added a RISC event example, courtesy of Marius Scurtescu.

- Applied wording clarifications suggested by Annabelle Backman and Yaron Sheffer.

- Applied numerous grammar, syntax, and formatting corrections.

Draft 05 - mbj - Changes were as follows:

- Simplified the definitions of the "iat" and "toe" claims in ways suggested by Annabelle Backman.

- Added privacy considerations text suggested by Annabelle Backman.

- Updated the RISC event example, courtesy of Marius Scurtescu.
- Reordered the claim definitions to place the required claims first.
- Changed to using the RFC 8174 boilerplate instead of the RFC 2119 boilerplate.

Authors' Addresses

Phil Hunt (editor)
Oracle Corporation

Email: phil.hunt@yahoo.com

Michael B. Jones
Microsoft

Email: mbj@microsoft.com
URI: http://self-issued.info/

William Denniss
Google

Email: wdenniss@google.com

Morteza Ansari
Cisco

Email: morteza.ansari@cisco.com